Lecture Notes in Computer Science 3072

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

David Zhang   Anil K. Jain (Eds.)

Biometric Authentication

First International Conference, ICBA 2004
Hong Kong, China, July 15-17, 2004
Proceedings

Springer

Volume Editors

David Zhang
The Hong Kong Polytechnic University, Department of Computing
Hung Hom, Kowloon, Hong Kong
E-mail: csdzhang@comp.polyu.edu.hk

Anil K. Jain
Michigan State University, Department of Computer Science and Engineering
3115 Engineering Building, East Lansing, MI 48824-1226, USA
E-mail: jain@cse.msu.edu

Preface

The past decade has seen a rapid growth in the demand for biometric-based authentication solutions for a number of applications. With significant advances in biometric technology and an increase in the number of applications incorporating biometrics, it is essential that we bring together researchers from academia and industry as well as practitioners to share ideas, problems and solutions for the development and successful deployment of state-of-the-art biometric systems.

The International Conference on Biometric Authentication (ICBA 2004) was the first major gathering in the Asia-Pacific region devoted to facilitating this interaction. We are pleased that this conference attracted a large number of high-quality research papers that will benefit the international biometrics research community. After a careful review of 157 submissions, 101 papers were accepted either as oral (35) or poster (66) presentations. In addition to these technical presentations, this conference also presented the results and summaries of three biometric competitions: Fingerprint Verification Competition (FVC 2004), Face Authentication Competition (FAC 2004), and Signature Verification Competition (SVC 2004). This conference provided a forum for the practitioners to discuss their practical experiences in applying the state-of-the-art biometric technologies which will further stimulate research in biometrics.

We are grateful to Jim L. Wayman, Edwin Rood, Raymond Wong, Jonathon Philips, and Francis Ho for accepting our invitation to give keynote talks at ICBA 2004. In addition, we would like to express our gratitude to all the contributors, reviewers, program committee and organizing committee members who made this a very successful conference. We also wish to acknowledge the Croucher Foundation, the International Association of Pattern Recognition, IEEE Hong Kong Section, the Hong Kong Polytechnic University, the National Natural Science Foundation in China, and Springer-Verlag for sponsoring this conference. Special thanks are due to Ajay Kumar, Vivien Lau, Jane You, George Baciu, Yunhong Wang, Jie Zhou, Michael Wong, Jie Guo, Jingqi Yan, Guangming Lu and Jiannong Cao for their dedication and hard work in various aspects of conference organization.

We hope that the fruitful technical interactions made possible by this conference will benefit your research and development efforts in biometrics.

March 2004   David Zhang
                                                                        Anil K. Jain

General Chairs:

Anil K. Jain (Michigan State University, USA)
Roland Chin (Hong Kong University of Science & Technology, HK)

Program Chairs:

David Zhang (Hong Kong Polytechnic University, HK)
Jim Wayman (San Jose State University, USA)
Joseph P. Campbell (MIT Lincoln Laboratory, USA)

Competition Coordinators:

Dario Maio (University of Bologna, Italy)
James Liu (Hong Kong Polytechnic University, HK)
Tieniu Tan (Chinese Academy of Sciences, PRC)

Exhibition Coordinators:

Helen Shen (Hong Kong University of Science & Technology, HK)
Lin Hong (Identix, USA)

Publication Chair:

Jiannong Cao (Hong Kong Polytechnic University, HK)

Local Arrangements Chairs:

Xiaoou Tang (Chinese University of Hong Kong, HK)

Finance Chair:

Jane You (Hong Kong Polytechnic University, HK)

Publicity Chairs:

Salil Prabhakar (DigitalPersona Inc., USA)
Davide Maltoni (University of Bologna, Italy)
Yunhong Wang (Chinese Academy of Sciences, PRC)

Program Committee:

Simon Baker (Carnegie Mellon University, USA)
Bir Bhanu (University of California Riverside, USA)
Zhaoqi Bian (Tsinghua University, PRC)
Josef Bigun (Halmstad Univ. and Chalmers Univ.of Tech., Sweden)
Wageeh W Boles (Queensland University of Technology, Australia)
Ruud (Rudolf) M. Bolle (IBM T.J. Watson Research Center, USA)
Manfred U.A. Bromba (Bromba GmbH, Germany)
Keith Chan (Hong Kong Polytechnic University, HK)
Ke Chen (UMIST, UK)
RI Damper (University of Southampton, UK)
Kenneth Dong (ID One Inc., USA)
Michael Fairhurst (University of Kent, UK)
Srinivas Gutta (Philips, USA)
Wen Gao (Chinese Academy of Sciences, PRC)
Lin Hong, (Identix Incorporation, USA)
Wen Hsing Hsu, (National Tsing Hua University, TW, ROC)
Behrooz Kamgar-Parsi (Naval Research Lab., USA)
Mohamed Kamel, (University of Waterloo, Canada)
Bernhard Kaemmerer (Siemens, Germany)
Daijin Kim (Pohang University of Science and Technology, Korea)
Jaihie Kim (Yonsei University, Korea)
Josef Kittler (University of Surrey, UK)
Naohisa Komatsu (Waseda University, Japan)
Alex Kot (Nanyang Technological University, Singapore)
Kin Man Lam (Hong Kong Polytechnic University, HK)
Julia Langenbach (Delphi Automotive Systems, Germany)
Seong-Whan Lee (Korea University, Korea)
Lei Li (Hosei University, Japan)
Stan Li (Microsoft Research Asia, PRC)
Xiaobo Li (University of Alberta, Canada)
Ze-Nian Li (Simon Frason University, Canada)
Lee Luan Ling (State University of Campinas, Brazil)
Zhiqiang Liu (City University of Hong Kong, HK)
Javier Ortega-Garcia (Universidad Politecnica de Madrid, Spain)
Edwige Pissaloux (Université de Rouen, France)
Salil Prabhakar (DigitalPersona Inc., USA)
K. Prasad (Ford Motor Co., USA)
James Reisman (Siemens Corporate Research, Inc., USA)
Gerhard Rigoll (Munich University of Technology, Germany)
Arun Ross (West Virginia University, USA)
Pat Sankar (Printrak, USA)
Ulf Cahn Von Seelen (Iridian Technologies, USA)
Dale Setlak (AuhenTec Inc., USA)
Pengfei Shi (Shanghai Jiao Tong University, PRC)

Table of Contents

Fingerprint

Iris

Signature

Speech

Biometric Fusion and Risk Analysis

Other Biometrics

FVC2004: Third Fingerprint Verification Competition

Dario Maio[1], Davide Maltoni[1], Raffaele Cappelli[1],
Jim L. Wayman[2], and Anil K. Jain[3]

[1] Biometric System Lab - DEIS, University of Bologna,
via Sacchi 3, 47023 Cesena - ITALY
{maio,maltoni,cappelli}@csr.unibo.it
[2] Biometric Test Center, College of Engineering, San Jose State University,
San Jose, CA 95192 - USA
JLWayman@aol.com
[3] Pattern Recognition and Image Processing Laboratory, Michigan State University,
East Lansing, MI 48824 - USA
jain@cse.msu.edu

Abstract. A new technology evaluation of fingerprint verification algorithms has been organized following the approach of the previous FVC2000 and FVC2002 evaluations, with the aim of tracking the quickly evolving state-of-the-art of fingerprint recognition systems. Three sensors have been used for data collection, including a solid state sweeping sensor, and two optical sensors of different characteristics. The competition included a new category dedicated to "light" systems, characterized by limited computational and storage resources. This paper summarizes the main activities of the FVC2004 organization and provides a first overview of the evaluation. Results will be further elaborated and officially presented at the International Conference on Biometric Authentication (Hong Kong) on July 2004.

1 Introduction

FVC2004 is the third international Fingerprint Verification Competition, a technology evaluation [1] of fingerprint recognition algorithms open to companies, academic research groups and independent developers. Organization of FVC2004 started in April 2003 and the final evaluations were conducted in January-March 2004 at the University of Bologna, Italy. 67 algorithms were evaluated: each algorithm, provided in the form of a binary executable program compliant with precise input/output specifications, was tested on four new fingerprint databases, previously unseen by the participants. This initiative follows FVC2000 [5], [2] and FVC2002 [6], [3], the first two international Fingerprint Verification Competitions, which were organized by the authors in the years 2000 and 2002 with results presented at the 15th ICPR and the 16th ICPR, respectively. The first two competitions received great attention from both academic and commercial organizations. Several research groups started using FVC2000 and FVC2002 datasets for their experimentations and some companies, which initially did not participate in the competitions, requested the organizers to measure their performance against the FVC2000 and/or the FVC2002 benchmark.

Table 1. The first two Fingerprint Verification Competitions

	FVC2000	FVC2002
Call for participation	November, 1999	October, 2001
Registration deadline	March 1st, 2000	January 10th, 2002
Submission deadline	June 1st, 2000	March 1st, 2002
Evaluation period	July–August, 2000	April–July, 2002
Notes	Anonymous part. not allowed	Anonymous part. allowed
Registered participants	25 (15 withdrew)	48 (19 withdrew)
Algorithms evaluated	11	31
Presentation of the results	15th ICPR Barcelona, September 2000	16th ICPR Quebec City, August 2002
Databases	4 (set A: 100x8, set B: 10x8)	4 (set A: 100x8, set B: 10x8)
DB1	Optical (KeyTronic)	Optical (Identix)
DB2	Capacitive (ST Microelectronics)	Optical (Biometrika)
DB3	Optical (Identicator Technology)	Capacitive (Precise Biometrics)
DB4	Synthetic (SFinGe v2.0)	Synthetic (SFinGe v2.51)
DB availability	DVD accompanying "Handbook of Fingerprint Recognition" [1]	
Website	http://bias.csr.unibo.it/fvc2000 (more than 41,000 accesses)	http://bias.csr.unibo.it/fvc2002 (more than 40,000 accesses)
HW/SW used for running the evaluation	Pentium III (450 MHz) Windows NT FVC Test suite v1.0	Pentium III (933 MHz) Windows 2000 FVC Test suite v1.2

Table 1 compares the first two competitions from a general point of view, highlighting the main differences. Beginning with FVC2002, to increase the number of companies and therefore to provide a more complete panorama of the state-of-the-art, the participants were allowed to participate anonymously. We have continued with this option in FVC2004.

The rest of this paper describes FVC2004: section 2 explains the organization of the event, section 3 the collection of the databases, section 4 the test protocol and the performance indicators measured, and section 5 reports some results; finally section 6 draws some conclusions.

2   FVC2004 Organization

Starting in April 2003 with the creation of the FVC2004 web site [4], we extensively publicized this event. All companies and research groups in the field known to us were invited to participate in the contest. All the participants in the past editions were also informed of the new evaluation. FVC2004 was also announced through mailing lists and biometric-related magazines. Four new databases were collected using three commercially available scanners and the synthetic generator SFinGe [1], [7] (see section 3). A representative subset of each database (sets B, see section 3) was made available to the participants for algorithm tuning to accommodate the image size and the variability of the fingerprints in the databases.

Table 2. Scanners/technologies used for collecting the databases

	Technology	Image	Resolution
DB1	Optical Sensor (CrossMatch V300)	640×480	500 dpi
DB2	Optical Sensor (Digital Persona U.are.U 4000)	328×364	500 dpi
DB3	Thermal Sweeping Sensor (Atmel FingerChip)	300×480	512 dpi
DB4	Synthetic Generator (SFinGe v3.0)	288×384	About 500 dpi

Two different sub-competitions (Open category and "Light" category) were organized using the same databases. Each participant was allowed to submit one algorithm in each category. The "light" category is intended for algorithms characterized by low computing needs, limited memory usage and small template size (see section 4).

By October 15th, 2003 (the deadline for registration), we had received 110 registrations, far more than our expectation. All the registered participants received the training subsets and detailed instructions for algorithm submission. By November 30th, 2003 (the deadline for submission) we had received a total of 69 algorithms from 46 participants: since two algorithms were not admitted due to unrecoverable incompatibility problems with FVC protocol, the final number of algorithms was 67 (41 competing in the Open category, 26 in the Light category). Once all the executables were submitted, feedback was sent to the participants by providing them the results of their algorithms over training subset B (the same data set they had previously been given) to allow them verify that run-time problems were not occurring on our side (In such cases, participants were allowed to submit updated versions of their algorithms).

At the time this paper is being written, the evaluation has been concluded and each participant has been informed of its individual results. Section 5 presents an overview of the results that will be further elaborated and published on the FVC2004 web site by April 15. Results presented here in section 5 are in anonymous form, since participants have until April 13 to decide whether or not to disclose their identities.

3   Database Collection

Four databases constitute the FVC2004 benchmark. Three different scanners and the SFinGE synthetic generator [1], [7] were used to collect fingerprints (see Table 2). Fig. 1. shows an image for each database, at the same scale factor.

A total of ninety students (24 years old on the average) enrolled in the Computer Science degree program at the University of Bologna kindly agreed to act as volunteers for providing fingerprints:

- volunteers were randomly partitioned into three groups of 30 persons; each group was associated to a DB and therefore to a different fingerprint scanner;
- each volunteer was invited to present him/herself at the collection place in three distinct sessions, with at least two weeks time separating each session;
- forefinger and middle finger of both the hands (four fingers total) of each volunteer were acquired by interleaving the acquisition of the different fingers to maximize differences in finger placement;
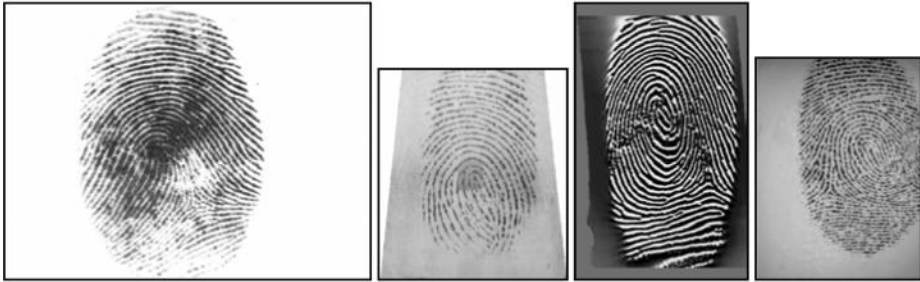
Fig. 1. One fingerprint image from each database, at the same scale factor

- no efforts were made to control image quality and the sensor platens were not systematically cleaned;
- at each session, four impressions were acquired of each of the four fingers of each volunteer;
- during the second session, individuals were requested to exaggerate skin distortion (impressions 1 and 2) and rotation (3 and 4) of the finger;
- during the third session, fingers were dried (impressions 1 and 2) and moistened (3 and 4).

At the end of the data collection, we had gathered for each database a total of 120 fingers and 12 impressions per finger (1440 impressions) using 30 volunteers. As in previous editions, the size of each database to be used in the test was established as 110 fingers, 8 impressions per finger (880 impressions); collecting some additional data gave us a margin in case of collection/labeling errors.

4   Test Protocol and Performance Evaluation

The protocol defining the format for the submitted algorithms was given in FVC2000 and remained unchanged throughout FVC2002 and FVC2004.

Each participant was required to submit two executable programs in the form of "win32 console applications". These executables take the input from command-line arguments and append the output to a text file. The input includes a database-specific configuration file. In fact, participants are allowed to submit a distinct configuration file for each database in order to adjust the algorithm's internal parameters (e.g. according to the different sizes of the images). Configuration files are text files or binary files and their I/O is the responsibility of the participant's code; these files can also contain pre-computed data to save time during enrollment and matching.

In the Open category, for practical testing reasons, the maximum response time of the algorithms was limited to 10 seconds for enrollment and 5 seconds for matching. No other limits were imposed in the Open category.

In the Light category, in order to create a benchmark for algorithms of light architectures, the following limits were imposed:

- maximum time for enrollment: 0.5 seconds;
- maximum time for matching: 0.3 seconds;

- maximum template size: 2 KBytes;
- maximum amount of memory allocated: 4 MBytes.

The evaluation (for both categories) was executed under Windows XP Professional O.S. on AMD Athlon 1600+ (1.41 GHz) PCs.

Each algorithm was tested by performing, for each database, the following matching attempts:

- *genuine recognition attempts*: the template of each impression was matched against the remaining impressions of the same finger, but avoiding symmetric matches (i.e. if the template of impression j was matched against impression k, template k was not matched against impression j);
- *impostor recognition attempts*: the template of the first impression of each finger was matched against the first impressions of the remaining fingers, but avoiding symmetric matches.

Then, for each database:

- a total of 700 enrollments were performed (the enrollment of the last impression of any finger did not need to be performed);
- if all the enrollments were correctly performed (no enrollment failures), the total number of genuine and impostor matching attempts was 2800 and 4950, respectively.

For each database and for each algorithm, the following performance indicators were measured and reported:

- genuine and impostor score histograms,
- FMR and FNMR graph and ROC graph,
- Failure To Enroll Rate and Failure To Match Rate,
- Equal Error Rate (EER), FMR100, FMR1000, ZeroFMR and ZeroFNMR,
- average match time and average enroll time,
- maximum memory allocated for enrollment and for match [*New in FVC2004*],
- average and maximum template size [*New in FVC2004*].

FMR (False Match Rate) and FNMR (False Non-Match Rate) are often referred as FAR (False Acceptance Rate) and FRR (False Rejection Rate) respectively, but the FAR/FRR notation is misleading in some applications. For example, in a welfare benefits system, which uses fingerprint identification to prevent multiple enrollments under false identity, the system "falsely accepts" an applicant if his/her fingerprint is "falsely not matched"; similarly, a "false match" causes a "false rejection". Consequently, in this paper we use the application neutral terminology "False Match" and "False Non-Match" rates. ZeroFMR is given as the lowest FNMR at which no False Matches occur and ZeroFNMR as the lowest FMR at which no False Non-Matches occur. FMR100 and FMR1000, are the values of FNMR for FMR=1/100 and 1/1000, respectively. These measures are useful to characterize the accuracy of fingerprint-based systems, which are often operated far from the EER point using thresholds which reduce FMR at the cost of higher FNMR. FVC2004 introduces indicators measuring the amount of memory required by the algorithms and the template size.
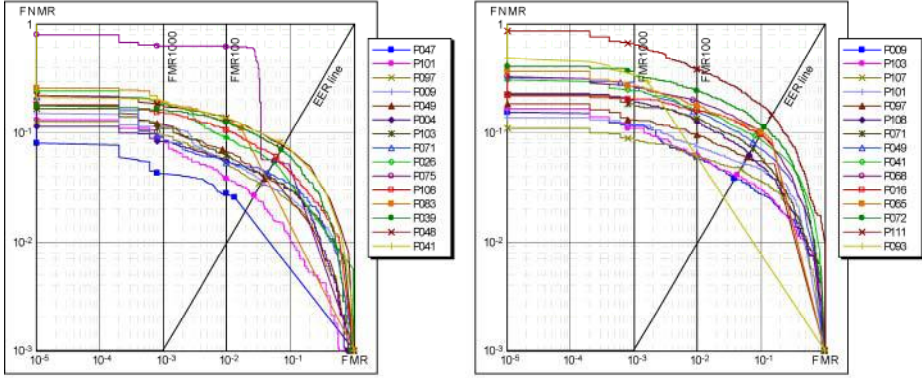
Fig. 2. ROC curves on DB1 (only top 15 algorithms are shown): Open category (on the left), Light category (on the right)

These indicators are reported for both the categories, although they are particularly interesting for the Light category.

5   Results

Due to the lack of space, only a small part of the results are reported here. Fig. 2. shows ROC graphs on the first database for both Open and Light categories. Tables 3 and 4 give the average results over the four databases.

Table 3. Open category: average results over the four databases, sorted by average EER (top ten algorithms)

Algorithm	Avg EER (%)	Avg FMR100 (%)	Avg FMR1000 (%)	Avg ZeroFMR (%)	Avg REJ$_{ENROLL}$ (%)	Avg REJ$_{MATCH}$ (%)	Avg Enroll Time (sec)	Avg Match Time (sec)	Avg Model Size (KB)	Max Model Size (KB)	Max Enroll Mem (KB)	Max Match Mem (KB)
P101	2.07	2.54	4.70	6.21	0.00	0.00	0.08	1.48	24.0	31.5	3204	7752
P047	2.10	2.96	4.61	6.59	0.00	0.00	2.07	2.07	1.3	2.8	5080	5796
P071	2.30	2.73	5.10	10.01	0.00	0.01	0.35	0.67	16.4	31.4	5872	9800
P004	2.45	3.27	5.63	7.34	0.00	0.00	0.69	0.71	2.0	3.8	7012	7032
P039	2.90	4.57	7.44	32.13	0.00	0.00	1.01	1.19	3.1	4.2	4192	4276
P097	3.13	4.49	7.30	11.85	0.04	0.02	0.47	0.51	14.9	28.1	5564	5780
P049	3.24	5.56	9.25	12.62	0.00	0.00	0.34	0.38	0.5	1.2	2472	2496
P009	3.31	4.93	8.32	11.63	0.00	0.00	0.25	0.24	1.3	2.9	2828	2860
P113	3.71	6.07	8.76	11.29	0.00	0.00	0.45	0.48	49.2	93.2	11936	12260
P068	4.03	6.87	11.08	15.68	0.00	0.00	0.43	0.47	7.9	7.9	4468	4456

Table 4. Light category: average results over the four databases, sorted by average EER (top ten algorithms)

Algorithm	Avg EER (%)	Avg FMR100 (%)	Avg FMR1000 (%)	Avg ZeroFMR (%)	Avg REJ$_{ENROLL}$ (%)	Avg REJ$_{MATCH}$ (%)	Avg Enroll Time (sec)	Avg Match Time (sec)	Avg Model Size (KB)	Max Model Size (KB)	Max Enroll Mem (KB)	Max Match Mem (KB)
P009	3.51	5.21	8.71	12.38	0.00	0.00	0.25	0.22	1.2	2.0	2844	2568
P107	3.69	4.68	6.65	8.75	0.00	0.00	0.13	0.13	0.2	0.6	1788	1800
P108	3.96	6.54	10.64	13.12	0.04	0.05	0.23	0.23	1.6	1.6	1952	1976
P101	4.29	6.02	8.91	10.57	0.00	0.00	0.09	0.17	1.1	1.2	2228	3044
P103	4.33	6.66	9.97	13.64	0.00	0.00	0.13	0.14	1.2	2.0	3572	3668
P097	4.86	6.96	10.21	13.12	0.00	0.00	0.19	0.19	2.0	2.0	2100	2108
P071	4.91	8.11	11.44	16.16	0.25	0.18	0.19	0.18	1.2	1.7	2552	2424
P016	5.26	7.68	10.46	13.93	0.00	0.00	0.17	0.19	1.4	2.0	2240	3004
P068	5.29	9.85	15.22	20.18	0.00	0.00	0.16	0.18	2.0	2.0	3448	3428
P049	5.64	10.55	17.13	24.12	0.00	0.00	0.12	0.14	0.5	0.9	1956	1980

6   Conclusions

The third Fingerprint Verification Competition attracted a very high number of participants: 67 algorithms were finally submitted and evaluated. The organization of the event and the evaluation of all the algorithms required more resources and time then expected. Due to time constraints, results presented in this paper are only an initial overview; detailed results and elaborated statistics will be published by April 15 in the FVC2004 web site and discussed at ICBA 2004, where FVC2004 results will be officially presented in a special session together with other technology evaluations on face and signature recognition.

7   References

[1]   D. Maltoni, D. Maio, A.K. Jain and S. Prabhakar, Handbook of Fingerprint Recognition, Springer (New York), 2003
[2]   FVC2000 web site: http://bias.csr.unibo.it/fvc2000
[3]   FVC2002 web site: http://bias.csr.unibo.it/fvc2002
[4]   FVC2004 web site: http://bias.csr.unibo.it/fvc2004
[5]   D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman and A.K. Jain, "FVC2000: Fingerprint Verification Competition", IEEE Transactions on Pattern Analysis Machine Intelligence, vol.24, no.3, pp.402-412, March 2002
[6]   D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman and A.K. Jain, "FVC2002: Second Fingerprint Verification Competition", in proceedings 16th International Conference on Pattern Recognition, Québec City, vol.3, pp.811-814, 2002.
[7]   R. Cappelli, D. Maio and D. Maltoni, "Synthetic Fingerprint-Database Generation", Proc. 16th ICPR, 2002

Face Authentication Competition
on the BANCA Database

Kieron Messer[1], Josef Kittler[1], Mohammad Sadeghi[1],
Miroslav Hamouz[1], Alexey Kostyn[1], Sebastien Marcel[2],
Samy Bengio[2], Fabien Cardinaux[2], Conrad Sanderson[2],
Norman Poh[2], Yann Rodriguez[2], Krzysztof Kryszczuk[2],
Jacek Czyz[3], L. Vandendorpe[3], Johnny Ng[4],
Humphrey Cheung[4], and Billy Tang[4]

[1] University of Surrey
Guildford, Surrey, GU2 7XH, UK
[2] Dalle Molle Institute for Perceptual Artificial Intelligence
CP 592, rue du Simplon 4, 1920 Martigny, Switzerland
[3] Université Catholique de Louvain
Batiment Stevin, Place du Levant 2, 1348 Louvain-la-Neuve, Belgium
[4] Titanium Technology Research Centre
10/F, Tianjin Building, 167 Connaught Road West, Hong Kong

Abstract. This paper details the results of a face verification competition [2] held in conjunction with the First International Conference on Biometric Authentication. The contest was held on the publically available BANCA database [1] according to a defined protocol [6]. Six different verification algorithms from 4 academic and commercial institutions submitted results. Also, a standard set of face recognition software from the internet [3] was used to provide a baseline performance measure.

1 Introduction

Face recognition technology is still developing and many papers on new face verification and recognition algorithms are being published almost daily. However, direct comparison of the reported methods can be difficult because tests are performed on different data with large variations in test and model database sizes, sensors, viewing conditions, illumination and background. Typically, it is unclear which methods are the best and for which scenarios they should be used. The use of common datasets along with evaluation protocols can help alleviate this problem.

The FERET database has defined a protocol for face identification and face verification [18]. However, only a development set of images from the database are released to researchers. The remaining are sequestered by the organisers to allow independent testing of the algorithms. To date three evaluations have taken place, the last one in the year 2000 [17].

More recently, two Face Recognition Vendor Tests [4] have been carried out, the first in 2000 and the second in 2002. The tests are done under supervision

and have time restrictions placed on how quickly the algorithms should compute the results. They are aimed more at independently testing the performance of commercially available systems, however academic institutions are also able to take part. In the more recent test 10 commercial systems were evaluated.

In the year 2000 a competition on the XM2VTS database using the Lausanne protocol [15] was organised [14]. As part of AVBPA 2003 a second competition on exactly the same data and testing protocol was organised [10]. All the data from the Xm2vts database is available from [5]. We believe that this open approach increases, in the long term, the number of algorithms that will be tested on the XM2VTS database. Each research institution is able to assess their algorithmic performance at any time.

In this paper we detail a competition on a new database known as the BANCA database [6]. The database was captured under 3 different realistic and challenging operating scenarios. Several protocols have also been defined which specifies which data should be used for training and testing. Again this database is being made available to the research community through [1].

The rest of this paper is organised as follows. In the next section the competition rules and performance criterion are described. Section 3 gives an overview of each algorithm which entered the competition and in the following section the results are detailed. Finally, some conclusions are made.

2   The Competition

All the experiments were carried according to the Matched Controlled (MC) configuration of the BANCA database. In this configuration a high quality camera was used to capture all the data and the lighting was controlled.

There were two separate parts to the competition.

Part I: Pre-registered: Images were supplied which had already been localised (by hand) and geometrically normalised. The resulting resolution of the images were 55x51 pixels, 8-bit grey-scale and contained just face pixels.

Part II: Automatic: Full video resolution colour images as shown in figure (a) were supplied. All participants had to use an automatic method of localisation for the at least the test phase of the protocol. Manual localisation for the training and evaluation phases was allowed.

Part I of the competition allows us to assess the underlying performance of the core face verification technology as the images had all been localised and geometrically normalised by the same method. Part II of the competition was aimed at testing the complete verification system, including the detection and localisation stage.

To assess the algorithmic performance the *False Rejection Rate* P_{FR} and *False Acceptance Rate* P_{FA} are typically used. These two measures are directly related, i.e. decreasing the false rejection rate will increase the number of false acceptances. The point at which $P_{FR} = P_{FA}$ is known as the EER (Equal Error Rate).

For this competition we requested that the entrants submit their results for 3 specific operating conditions which corresponded to 3 different values of the Cost Ratio $R = C_{FA}/C_{FR}$, namely $R = 0.1, R = 1, R = 10$. Assuming equal *a priori* probabilities of genuine clients and impostor, these situations correspond to 3 quite distinct cases:

$R = 0.1 \rightarrow$ FA is an order of magnitude less
            harmful than FR,
$R = 1 \quad \rightarrow$ FA and FR are equally harmful,
$R = 10 \rightarrow$ FA is an order of magnitude more
            harmful than FR.

The entrants were asked to submit the Weighted Error Rate (WER) for the test data of groups $G1$ and $G2$ at the three different values of R. WER is defined as:

$$WER(R) = \frac{P_{FR} + R \, P_{FA}}{1 + R} \ . \tag{1}$$

For each group and at each operating point there are in total 390 true client claims and 520 impostor attacks.

3   Overview of Algorithms

In this section the algorithms that participated in the contest are summarised. Also, we downloaded a standard set of face recognition software from the internet [3] to provide a baseline performance measure on this database. Due to space limitations we have published the results from these experiments at [2].

3.1   Dalle Molle Institute
for Perceptual Artificial Intelligence (IDIAP)

Pseudo-2D HMM (IDIAP - HMM). The system is comprised of two main parts: an automatic face locator and a local feature probabilistic classifier. To locate faces, a fast cascade of boosted Haar-like features is applied to the integral image to detect potential faces [23], followed by post-processing using a Multi-Layer Perceptron [20] to provide the final localized face. The probabilistic classifier uses DCTmod2 features [22] and models faces using pseudo-2D Hidden Markov Models (HMMs) [7]. In DCTmod2 feature extraction, each given face is analyzed on a block by block basis; from each block a subset of Discrete Cosine Transform (DCT) coefficients is obtained; coefficients which are most affected by illumination direction changes are replaced with their respective horizontal and vertical deltas, computed as differences between coefficients from neighbouring blocks. For the pseudo-2D HMM topology, we use a top-to-bottom main HMM with each state being modeled by a left-to-right HMM. To circumvent the problem of small amount of client training data, parameters for each client model are obtained via Maximum *a Posteriori* (MAP) adaptation of a generic face HMM; the generic face HMM is trained using the Expectation Maximization

algorithm, using world model training data. A score for a given face is found by taking the difference between the log-likelihood of the face belonging to the true client and the log-likelihood of the face belonging to an impostor; a global threshold (i.e. the same for all clients) is used in making the final verification decision. As the generic face HMM is deemed to be a good representation of the general population, it is used as the impostor model.

Fusion (IDIAP - Fusion). The system is composed of an automatic face locator, three classification subsystems and a fusion stage. The face locator has two components: a fast cascade of boosted haar-like features is applied to the integral image to detect potential faces [23], followed by post-processing using a Multi-Layer Perceptron (MLP) [20]. The first two classification subsystems are based on local features and generative models (namely, DCTmod2 features, Gaussian Mixture Models & pseudo-2D Hidden Markov Models [7]), while the third subsystem uses Linear Discriminant Analysis based feature extraction (i.e. holistic features) and a MLP for classification [13]. Finally, the opinions of the three subsystems are fused using an MLP based approach [19].; a global threshold (i.e. the same for all clients) is used in making the final verification decision.

3.2   Université Catholique de Louvain (UCL)

Linear Discriminant Analysis (UCL - LDA). The method is based on classical Linear Discriminant Analysis (LDA) or fisherfaces. The matching score is computed in the LDA subspace using normalised correlation. A large auxiliary dataset is used to compute the LDA basis. Note that instead of using original image $I(x, y)$, we take advantage of face symmetry and use the *symmetrised* image $I_s = (I(x, y) + I(-x, y))/2$ [9].

Fusion (UCL - Fusion). The method is based on a combination of two different face experts. The first expert is based on classical Linear Discriminant Analysis (LDA) or fisherfaces. The matching score is computed in the LDA subspace using normalised correlation. The second expert uses a SVM classifier with linear kernel trained directly in the image space. The two expert scores are conciliated by a fusion module based on a Support Vector Classifier [8].

3.3   Titanium Technology Research Centre

Dynamic Local Feature Analysis (DLFA) is used in this experiment. DLFA is developed from the concept of LFA. Texture and shape information is combined using the Local Feature Analysis (LFA) technique to develop a robust face recognition algorithm. The shape information is obtained by using an adaptive edge detecting method that can reduce the effect of different lighting conditions, while the texture information provides normalized facial features conveying more details about the image.

The approach can be divided into two main steps. The first step is preprocessing. The goal of this step is to reduce noise, transform the input face image into a binary one by dynamic edge detection and then extract the texture of the face. The second step employs the local feature analysis to combine both edge of face shape and the texture [16].

3.4   University of Surrey (UniS)

The input image data is firstly projected into the fisher faces space using the Linear Discriminant Analysis. The Isotropic Gradient Direction metric [21] is then used as the scoring function which measures the degree of similarity between the test image and the claimed identity template. For the first part of the competition only the intensity image was used to comply with the competition rules. For part II of the competition this process was performed in three different colour spaces namely intensity (I), chroma-g (G/I) and opponent chroma-rg ((R-G)/I) spaces [11]. The final score is then calculated by averaging the individual channel scores. The resulting score is finally compared to a pre-set threshold in order to decide whether the claim is genuine or impostor. We have used the XM2VTS database for the LDA training, the histogram equalisation for the photometric normalisation and client specific thresholding method for calculating the thresholds.

4   Results and Discussion

Table 1 shows the results for the pre-registered part of the competition. They show the WER for each group at the three operating points specified in section 2. The last column shows the average WER across all six test conditions. The best performing algorithm was UCL-Fusion which achieved an average WER of 1.95%. Second was IDIAP-Fusion with 2.70%. It is interesting to note that these two best performing algorithms used intra-modal fusion and are approximately 50% better then the two best single expert systems (i.e. UniS 2.99% and IDIAP-HMM 3.53%). This seems consistent with other published results [12]. However, there is a grey area over what constitutes a system using intramodal fusion. For example, a neural network based system uses hundreds of simple experts to arrive at a decision.

Table 2 shows the results using automatic registration for at least the verification stage. This time the best result is IDIAP-HMM with a performance of 3.78%. What is interesting about this result is that there is very little degradation in performance from the manual case (i.e. 3.53%). The fusion systems which performed well in part I of the test degrade in performance by a factor 2. The experts used in these fusion systems have not been so robust to the localisation errors. What is clear from these results is that accurate localisation is critical to verification performance and still needs to improve to match the performance provided by a manual operator.

Table 1. The Weighted Erro Rates on the two groups at the three different operating points using the pre-registered images

	R=0.1 (WER)		R=1(WER)		R=10 (WER)		
	G1	G2	G1	G2	G1	G2	Av
IDIAP-HMM	7.52	4.90	5.45	0.64	2.56	0.12	3.53
IDIAP-Fusion	6.99	2.42	3.85	1.76	0.70	0.47	2.70
UCL-LDA	6.53	1.17	7.05	2.88	1.28	2.10	3.50
UCL-Fusion	3.90	0.26	4.32	1.44	1.28	0.47	1.95
UniS	7.12	0.89	5.58	1.98	1.47	0.92	2.99
Titanium	4.12	3.90	3.04	3.10	1.97	2.12	3.04

Table 2. The Weighted Erro Rates on the two groups at the three different operating points using automatic localisation

	R=0.1 (WER)		R=1 (WER)		R=10 (WER)		
	G1	G2	G1	G2	G1	G2	Av
IDIAP-HMM	7.78	3.76	5.13	2.08	1.17	2.74	3.78
IDIAP-Fusion	6.53	8.68	7.53	6.73	2.10	1.40	5.50
UCL-LDA	8.13	7.11	10.58	9.46	5.89	6.12	7.88
UCL-Fusion	4.28	3.64	9.13	5.12	2.44	1.80	4.40
UniS	5.75	3.00	6.38	4.50	1.95	1.97	3.93
Titanium	5.84	5.12	4.42	3.94	3.01	2.76	4.18

5   Conclusions

This paper presents a comparison of face verification algorithms on a new publically available and challenging face database. It was organised in conjunction with the First International Conference on Biometric Authentication. Six different verification algorithms from a variety of academic and commercial institutions entered the competition.

References

[1] *BANCA; http://www.ee.surrey.ac.uk/banca/.*   8, 9
[2] *BANCA; http://www.ee.surrey.ac.uk/banca/icba2004.*   8, 10
[3] *The CSU Face Identification Evaluation System;*
     http://www.cs.colostate.edu/evalfacerec.   8, 10
[4] *Face Recognition Vendor Tests; http://www.frvt.org.*   8

[5] *The XM2VTSDB; http://www.ee.surrey.ac.uk/ Research/VSSP/xm2vtsdb/.* 9

[6] E. Bailly-Bailliere, S. Bengio, F. Bimbot, M. Hamouz, J. Kittler, J. Mariethoz, J. Matas, K. Messer, V. Popovici, F. Poree, B. Ruiz, and J. P. Thiran. The BANCA database and evaluation protocol. In *Audio- and Video-Based Biometric Person Authentication: Proceedings of the 4th International Conference, AVBPA 2003*, volume 2688 of *Lecture Notes in Computer Science*, pages 625–638, Berlin, Germany, June 2003. Springer-Verlag. 8, 9

[7] F. Cardinaux, C. Sanderson, and S. Bengio. Face verification using adapted generative models. In *Proc. Int. Conf. Automatic Face and Gesture Recognition (AFGR), Seoul, Korea.*, 2004. 10, 11

[8] J Czyz, M Sadeghi, J Kittler, and L Vandendorpe. Decision fusion for face authentication. In *First International Conference on Biometric Authentication*, 2004. 11

[9] Jacek Czyz. *Decision fusion for identity verification using facial images*. PhD thesis, Universite Catholique de Louvain, 2003. 11

[10] Kieron Messer et al. Face verification competition on the xm2vts database. In *4th International Conference on Audio and Video Based Biometric Person Authentication*, pages 964–974, June 2003. 9

[11] J. Kittler and M. Sadeghi. Physics-based decorrelation of image data for decision level fusion in face verification. In *The Fifth Workshop on Multiple Classifier Systems (MCS 2004)*, Cagliari, Italy, June, 2004. 12

[12] Josef Kittler, Marco Ballette, Jacek Czyz, Fabio Roli, and Luc Vandendorpe. Enhancing the performance of personal identity authentication systems by fuion of face verification experts. In *International Conference on Multimedia and Expo*, 2002. 12

[13] S. Marcel. A symmetric transformation for lda-based face verification. In *Proc. Int. Conf. Automatic Face and Gesture Recognition (AFGR), Seoul, Korea*, 2004. 11

[14] J Matas, M Hamouz, K Jonsson, J Kittler, Y P Li, C Kotropoulos, A Tefas, I Pitas, T Tan, H Yan, F Smeraldi, J Bigun, N Capdevielle, W Gerstner, S Ben-Yacoub, Y Abdeljaoued, and E Mayoraz. Comparison and face verification results on the xm2vts database. In A Sanfeliu, J J Villanueva, M Vanrell, R Alquezar, J Crowley, and Y Shirai, editors, *Proceedings of International Conference on Pattern Recognition, Volume 4*, pages 858–863, 2000. 9

[15] K Messer, J Matas, J Kittler, J Luettin, and G Maitre. XM2VTSDB: The Extended M2VTS Database. In *Second International Conference on Audio and Video-based Biometric Person Authentication*, March 1999. 9

[16] Johnny Ng and Humphrey Cheung. Dynamic local feature analysis for face recognition. In *First International Conference on Biometric Authentication*, 2004. 12

[17] P. J. Phillips, H. Moon, P. Rauss, and S. A. Rizvi. The feret evaluation methodology for face-recognition algorithms. volume 22, pages 1090–1104, October 2000. 8

[18] P.J. Phillips, H. Wechsler, J.Huang, and P.J. Rauss. The FERET database and evaluation procedure for face-recognition algorithm. *Image and Vision Computing*, 16:295–306, 1998. 8

[19] N. Poh and S. Bengio. Non-linear variance reduction techniques in biometric authentication. In *Proc. Workshop on Multi-Modal User Authentication (MMUA),Santa Barabara*, pages 123–130, 2003. 11

[20] H.A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Tran. Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998. 10, 11

[21] M. Sadeghi and J. Kittler. Decision making in the lda space: Generalised gradient direction metric. In *The 6th Int. Conf. on Automatic Face and Gesture Recognition*, Seoul, Korea, May, 2004. 12

[22] C. Sanderson and K.K. Paliwal. Fast features for face authentication under illumination direction changes. *Pattern Recognition Letters*, 24(14):2409–2419, 2003. 10

[23] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)Hawai*, volume 1, pages 511–518, 2001. 10, 11

SVC2004: First International Signature Verification Competition

Dit-Yan Yeung[1], Hong Chang[1], Yimin Xiong[1], Susan George[2], Ramanujan Kashi[3], Takashi Matsumoto[4], and Gerhard Rigoll[5]

[1] Hong Kong University of Science and Technology, Hong Kong
[2] University of South Australia, Australia
[3] Avaya Labs Research, USA
[4] Waseda University, Japan
[5] Munich University of Technology, Germany

Abstract. Handwritten signature is the most widely accepted biometric for identity verification. To facilitate objective evaluation and comparison of algorithms in the field of automatic handwritten signature verification, we organized the First International Signature Verification Competition (SVC2004) recently as a step towards establishing common benchmark databases and benchmarking rules. For each of the two tasks of the competition, a signature database involving 100 sets of signature data was created, with 20 genuine signatures and 20 skilled forgeries for each set. Eventually, 13 teams competed for Task 1 and eight teams competed for Task 2. When evaluated on data with skilled forgeries, the best team for Task 1 gives an equal error rate (EER) of 2.84% and that for Task 2 gives an EER of 2.89%. We believe that SVC2004 has successfully achieved its goals and the experience gained from SVC2004 will be very useful to similar activities in the future.

1 Introduction

Handwritten signature verification is the process of confirming the identity of a user based on the handwritten signature of the user as a form of behavioral biometrics [1, 2, 3]. Automatic handwritten signature verification is not a new problem. Many early research attempts were reviewed in the survey papers [4, 5]. The primary advantage that signature verification has over other types of biometric technologies is that handwritten signature is already the most widely accepted biometric for identity verification in daily use. The long history of trust over signature verification means that people are very willing to accept a signature-based biometric authentication system.

However, there has not been any major international effort that aims at comparing different signature verification methods systematically. As common benchmark databases and benchmarking rules are often used by researchers in such areas as information retrieval and natural language processing, researchers in biometrics increasingly see the need for such benchmarks for comparative studies. For example, fingerprint verification competitions (FVC2000 and FVC2002)

have been organized to attract participants from both academia and industry to compare their algorithms objectively. As inspired by these efforts, we organized the First International Signature Verification Competition (SVC2004) recently.

The objective of SVC2004 is to allow researchers and practitioners to compare the performance of different signature verification systems systematically based on common benchmark databases and benchmarking rules. Since on-line handwritten signatures collected via a digitizing tablet or some other pen-based input device can provide very useful dynamic features such as writing speed, pen orientation and pressure in addition to static shape information, only online handwritten signature verification was included in this competition.

We made it clear to all participants from the very beginning that this event should not be considered as an official certification exercise, since the databases used in the competition were only acquired in laboratory rather than real environments. Moreover, the performance of a system can vary significantly with how forgeries are provided. Furthermore, handwritten signature databases are highly language dependent. Nevertheless, it is hoped that through this exercise, researchers and practitioners could identify areas where possible improvements to their algorithms could be made.

2   Participants

The Call for Participation announcement was released on 30 April 2003. By the registration deadline (30 November 2003), 33 teams (27 from academia and six from industry) had registered for the competition showing their intention to participate in either one or both tasks of the competition. Of the 33 teams registered, 16 teams eventually submitted their programs for Task 1 while 13 teams for Task 2 by the submission deadline (31 December 2003). Some teams participated in both tasks. One team submitted a program that requires a licensed software to run it. Eventually this team withdrew. So we ended up having a total of 15 teams for Task 1 and 12 teams for Task 2. All are academic teams from nine different countries (Australia, China, France, Germany, Korea, Singapore, Spain, Turkey, and United States). Table 1 shows all the participating teams, with nine decided to remain anonymous after the results were announced. Team 19 submitted three separate programs for each task based on different algorithms. To distinguish between them when reporting the results, we use 19a, 19b and 19c as their Team IDs.

3   Signature Databases

3.1   Database Design

SVC2004 consists of two separate signature verification tasks using two different signature databases. The signature data for the first task contain coordinate information only, but the signature data for the second task also contain additional information including pen orientation and pressure. The first task is suitable for

Table 1. SVC2004 participating teams

Team ID	Institution	Country	Member(s)	Task(s)
3		Australia	V. Chandran	1 & 2
4	*anonymous*			1 & 2
6	Sabanci University	Turkey	Alisher Kholmatov Berrin Yanikoglu	1 & 2
8	*anonymous*			2
9	*anonymous*			1 & 2
12	*anonymous*			1
14	*anonymous*			1 & 2
15	*anonymous*			1
16	*anonymous*			1
17	*anonymous*			1 & 2
18	*anonymous*			1 & 2
19	Biometrics Research Laboratory, Universidad Politecnica de Madrid	Spain	Julian Fierrez-Aguilar Javier Ortega-Garcia	1 & 2
24	Fraunhofer, Institut Sichere Telekooperation	Germany	Miroslav Skrbek	1
26	State University of New York at Buffalo	USA	Aihua Xu Sargur N. Srihari	1
29	Institut National des Télécommunications	France	Bao Ly Van Sonia Garcia-Salicetti Bernadette Dorizzi	2

on-line signature verification on small pen-based input devices such as personal digital assistants (PDA) and the second task on digitizing tablets.

Each database has 100 sets of signature data. Each set contains 20 genuine signatures from one signature contributor and 20 skilled forgeries from at least four other contributors. Unlike physiological biometrics, the use of skilled forgeries for evaluation is very crucial to behavioral biometrics such as handwritten signature. Of the 100 sets of signature data, only the first 40 sets were released (on 25 October 2003) to participants for developing and evaluating their systems before submission (by 31 December 2003). While the first 40 sets for the two tasks are totally different, the other 60 sets (not released to participants) are the same except that the pen orientation and pressure attributes are missing in the signature data for Task 1. Although both genuine signatures and skilled forgeries were made available to participants, user enrollment during system evaluation accepted only five genuine signatures from each user, although multiple sets of five genuine signatures each were used in multiple runs. Skilled forgeries were not used during the enrollment process. They were only used in the matching process for system performance evaluation. Evaluation of signature verification performance for each user was only started after all users had been enrolled. Therefore, participants could make use of genuine signatures from other users to improve the verification accuracy for a user if they so wished.

3.2 Data Collection

Each data contributor was asked to contribute 20 genuine signatures. For privacy reasons, the contributors were advised not to use their real signatures in daily use. Instead, they were suggested to design a new signature and to practice the writing of it sufficiently so that it remained relatively consistent over different

signature instances, just like real signatures. Contributors were also reminded that consistency should not be limited to spatial consistency in the signature shape but should also include temporal consistency of the dynamic features.

In the first session, each contributor contributed 10 genuine signatures. Contributors were advised to write naturally on the digitizing tablet (WACOM Intuos tablet) as if they were enrolling themselves to a real signature verification system. They were also suggested to practice thoroughly before the actual data collection started. Moreover, contributors were provided the option of not accepting a signature instance if they were not satisfied with it. In the second session, which was normally at least one week after the first one, each contributor came again to contribute another 10 genuine signatures.

The skilled forgeries for each data contributor were provided by at least four other contributors in the following way. Using a software viewer, a contributor could see the genuine signatures that he or she tried to forge. The viewer could replay the writing sequence of the signatures on the computer screen. Contributors were also advised to practice the skilled forgeries for a few times until they were confident to proceed to the actual data collection.

The signatures are mostly in either English or Chinese. Although most of the data contributors are Chinese, many of them actually use English signatures frequently in daily applications.

3.3   Signature Files

Each signature is stored in a separate text file. The naming convention of the files is `UxSy`, where `x` is the user ID and `y` is the signature ID. Genuine signatures correspond to `y` values from 1 to 20 and skilled forgeries from 21 to 40. However, random re-numbering was performed during the evaluation process to avoid the class information from being revealed by the file names.

In each signature file, the signature is represented as a sequence of points. The first line stores a single integer which is the total number of points in the signature. Each of the following lines corresponds to one point characterized by features listed in the following order (the last three features are missing in the signature files for the first task): x-coordinate, y-coordinate, time stamp, button status, azimuth, altitude, and pressure.

4   Performance Evaluation

4.1   Testing Protocol

Both tasks used the same code submission scheme. For each task, each team was required to submit two executable files, one for performing enrollment and the other for matching. Executable files were for the Windows platform and could run in command-line mode without any graphical user interface.

The testing protocol is as follows. Each program was evaluated on two signature databases. The first database, which was released to the participants,

consists of genuine signatures and skilled forgeries for 40 users. The second database consists of similar signature data for 60 users. This set was not released to the participants. For each user from either database, 10 trials were run based on 10 different random subsets of five genuine signatures each from files S1-S10 for enrollment. After each enrollment trial, the program was evaluated on 10 genuine signatures (S11-S20), 20 skilled forgeries (S21-S40), and 20 random forgeries selected randomly from genuine signatures of 20 other users. Whenever randomness was involved, the same random sets were used for all teams.

For each signature tested, a program is expected to report a similarity score, between 0 and 1, which indicates the similarity between the signature and the corresponding template. The larger the value is, the more likely the signature tested will be accepted as a genuine signature. Based on these similarity scores, we computed false rejection rates (FRR) and false acceptance rates (FAR) for different threshold values. Equal error rates (ERR) and Receiver Operating Characteristics (ROC) curves were then obtained separately for skilled forgeries and random forgeries.

4.2   Results

The programs of some teams encountered problems during the evaluation process. In particular, they failed to report similarity scores for some input signatures. For fairness of comparison, EER statistics and ROC curves are not reported for these programs. Besides reporting the average EER over all users and all 10 trials for each team, we also report the standard deviation (SD) and maximum EER values.

Tables 2 and 3 show the EER results for both tasks evaluated on signature data from 60 users not released to participants. Figures 1 and 2 show the corresponding ROC curves for the evaluation with skilled forgeries. The results of some teams (Teams 3 and 9 for Task 1 and Teams 3, 9 and 29 for Task 2) are not included in the tables since their programs failed to report similarity scores for some signatures. For both tasks, Team 6 from the Sabanci University of Turkey gives the lowest average EER values when tested with skilled forgeries. Due to page limit, some results are not included in this paper. Readers are referred to http://www.cs.ust.hk/svc2004/results.html for more details.

5   Discussions

We have noticed that the EER values tend to have relatively large variations as can be seen from the SD values. While behavioral biometrics generally have larger intra-class variations than physiological biometrics, we speculate that this is at least partially attributed to the way in which the signature databases were created for SVC2004. Specifically, the signatures are not the real signatures of the data contributors. Although they were asked to practice thoroughly before signature collection, larger variations than expected were still expected.

Table 2. EER statistics for Task 1 (60 users)

Team ID	10 genuine signatures + 20 skilled forgeries			10 genuine signatures + 20 random forgeries		
	Average	SD	Maximum	Average	SD	Maximum
6	**2.84%**	**5.64%**	**30.00%**	2.79%	5.89%	50.00%
24	4.37%	6.52%	25.00%	1.85%	2.97%	15.00%
26	5.79%	10.30%	52.63%	5.11%	9.06%	50.00%
19b	5.88%	9.21%	50.00%	2.12%	3.29%	15.00%
19c	6.05%	9.39%	50.00%	2.13%	3.29%	15.00%
15	6.22%	9.38%	50.00%	2.04%	3.16%	15.00%
19a	6.88%	9.54%	50.00%	2.18%	3.54%	22.50%
14	8.77%	12.24%	57.14%	2.93%	5.91%	40.00%
18	11.81%	12.90%	50.00%	4.39%	6.08%	40.00%
17	11.85%	12.07%	70.00%	3.83%	5.66%	40.00%
16	13.53%	12.99%	70.00%	3.47%	6.90%	52.63%
4	16.22%	13.49%	66.67%	6.89%	9.20%	48.57%
12	28.89%	15.95%	80.00%	12.47%	10.29%	55.00%

Table 3. EER statistics for Task 2 (60 users)

Team ID	10 genuine signatures + 20 skilled forgeries			10 genuine signatures + 20 random forgeries		
	Average	SD	Maximum	Average	SD	Maximum
6	**2.89%**	**5.69%**	**30.00%**	2.51%	5.66%	50.00%
19b	5.01%	9.06%	50.00%	1.77%	2.92%	10.00%
19c	5.13%	8.98%	51.00%	1.79%	2.93%	10.00%
19a	5.91%	9.42%	50.00%	1.70%	2.86%	10.00%
14	8.02%	10.87%	54.05%	5.19%	8.57%	52.63%
18	11.54%	12.21%	50.00%	4.89%	6.65%	45.00%
17	12.51%	13.01%	70.00%	3.47%	5.53%	30.00%
4	16.34%	14.00%	61.90%	6.17%	9.24%	50.00%

We have also noticed that the results for Task 1 are generally slightly better than those of Task 2. This seems to imply that additional dynamic information including pen orientation and pressure is not useful and can lead to impaired performance. While conflicting results have been seen in the literature, we believe this is again due to the way of collecting our signature data, as discussed above. The invariance of pen orientation and pressure is likely to be less than that of other dynamic information used for Task 1.

From these findings, we are further convinced that establishing benchmark databases that faithfully reflect the nature of signature data found in real-world applications is of great importance to the research community. We hope SVC2004 can facilitate collaborative efforts in establishing such databases before long.

More performance criteria may be considered in the future. While this competition considers only accuracy as measured by EER, it would be useful, particularly from the application perspective, to include other criteria such as running time. Moreover, we may also allow a program to reject a signature during the enrollment and/or testing phase.
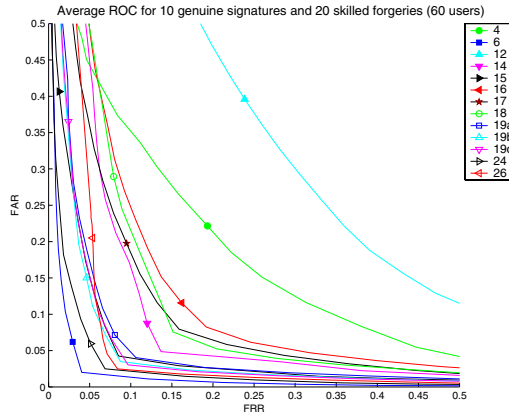
Fig. 1. ROC curves for Task 1 (60 users)
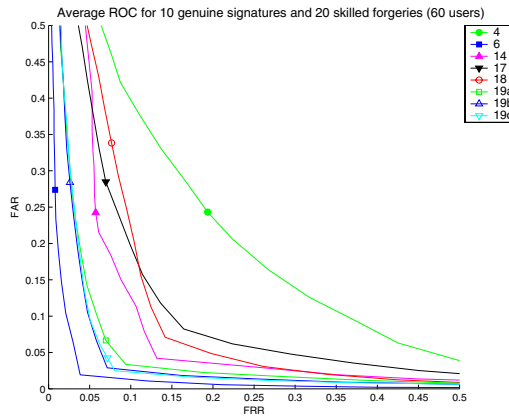
Fig. 2. ROC curves for Task 2 (60 users)

References

[1] V.S. Nalwa. Automatic on-line signature verification. *Proceedings of the IEEE*, 85(2):215–239, 1997. 16

[2] A. Jain, R. Bolle, and S. Pankanti. *Biometrics: Personal Identification in Networked Society*. Kluwer Academic Publishers, Boston, MA, USA, 1999. 16

[3] A.K. Jain, F.D. Griess, and S.D. Connell. On-line signature verification. *Pattern Recognition*, 35(12):2963–2972, 2002. 16

[4] R. Plamondon and G. Lorette. Automatic signature verification and writer identification – the state of the art. *Pattern Recognition*, 22(2):107–131, 1989. 16

[5] F. Leclerc and R. Plamondon. Automatic signature verification: the state of the art – 1989–1993. *International Journal of Pattern Recognition and Artificial Intelligence*, 8(3):643–660, 1994. 16

Watch List Face Surveillance
Using Transductive Inference

Fayin Li and Harry Wechsler

Department of Computer Science
George Mason University, Fairfax, VA, 22030, USA
{fli,Wechsler}@cs.gmu.edu

Abstract. The *open set* recognition task, most challenging among the biometric tasks, operates under the assumption that not all the probes have mates in the gallery. It requires the availability of a *reject* option. For face recognition open set corresponds to the *watch list face surveillance* task, where the face recognition engine must *detect* or *reject* the probe. The above challenges are addressed successfully in this paper using *transduction*, which is a novel form of inductive learning. Towards that end, we introduce the *Open Set TCM-kNN* algorithm, which is based upon algorithmic randomness and transductive inference. It is successfully validated on the (small) watch list task (80% or more of the probes lack mates) using FERET datasets. In particular, Open Set TCM-kNN provides on the average 96% correct detection / rejection and identification using the PCA and/or Fisherfaces components for face representation.

1   Introduction

The *open set* recognition task, most challenging among the biometric tasks, operates under the assumption that not all the probes have mates in the gallery. The above task, which has to detect the presence or absence of some biometric within the gallery while correctly finding the identity for the same biometric, requires the availability of a *reject* option. For face recognition open set corresponds to the *watch list face surveillance* task, where the face recognition engine must first find out if an individual probe is, or is not, on the watch list. If the probe is on the watch list, the face recognition engine must then *identify / recognize* the individual.

The performance index for correctly detecting and identifying an individual on the watch list is called the *detection and identification/ recognition rate*. "Typically, the watch list task is more difficult than the identification [recognition] or verification task alone. For the best system [FRVT2002 face recognition engine] using a watch list of 25 people, the detection and identification rate is 77%. Systems achieve better performance for a smaller watch list. If the impetus of the watch list application is to detect and identify [recognize] the "most wanted" individuals, the watch list should be kept as small as possible" (Phillips et al., 2003). The watch list face surveillance task is addressed successfully in this paper using *transduction*, which is a novel form of inductive learning (Vapnik, 1998). Towards that end, we introduce a novel approach, the *Open Set TCM-kNN* (Transduction Confidence Machine – *k* Nearest Neighbor), which is based upon algorithmic randomness and transductive inference. Open Set

TCM-kNN is successfully validated on the (small) watch list task (80% or more of the probes lack mates) using FERET datasets characterized by challenging (varying illumination and changing facial expression) face images. In particular, Open Set TCM-kNN provides on the average 96% correct detection / rejection and identification / recognition using the PCA and/or Fisherfaces components for face representation.

2   Transduction Confidence Machine (TCM) – kNN

Confidence measures can be based upon universal tests for randomness, or their approximation. Universal tests for randomness are not computable and hence one has to approximate the p-values using non-universal tests. We use the p-value construction in Gammerman et al. (1998) and Proedrou et al. (2001) to define information quality. Given a sequence of proximities (distances) between the given training (gallery) set and a probe i, the *strangeness* of i with putative label y in relation to the rest of the training set exemplars is defined as:

$$\alpha_i = \left(\sum_{j=1}^{k} D_{ij}^y \right) \left(\sum_{j=1}^{k} D_{ij}^{-y} \right)^{-1} \tag{1}$$

The strangeness measure is the ratio of the sum of the k nearest distances D from the same class (y) to the sum of the k nearest distances from all other classes ($-y$). Based on the *strangeness*, a valid randomness test (Nouretdinov et al., 2001) defines the p-value measure of a probe with a possible classification assigned to it as

$$p = \frac{f(\alpha_1) + f(\alpha_2) + \cdots + f(\alpha_m) + f(\alpha_{new})}{(m+1)f(\alpha_{new})} \tag{2}$$

where f is some monotonic non-decreasing function with $f(0) = 0$, e.g., $f(\alpha) = \alpha$, m is the number of training examples, and α_{new} is the strangeness measure of a new test probe exemplar c_{new} with a possible classification. An alternative definition available for the p-value is $p\ (c_{new}) = \#\{i:\ \alpha_i \geq \alpha_{new}\}/(m+1)$.

Based on strangeness and p-value definitions above, Proedrou et al. (2001) have proposed that TCM-kNN (Transduction Confidence Machine-k Nearest Neighbor) serves as a formal transduction inference algorithm. If there are n classes in the training data, there are n p-values for each probe exemplar. Using the p-value one can now predict the class membership as the one that yields the largest p-value. This is defined as the *credibility* of the assignment made. The associated *confidence* measure, which is either the 1st largest p-value or one minus the 2nd largest p-value, indicates how close the first two assignments are. One can compare the top ranked assignments, rather than only the first two assignments, and define additional confidence criteria. Both the credibility and confidence measures allow the face recognition engine to adapt to existing conditions and act accordingly.

We have used several well-known similarity measures to evaluate their effect on different face representation (PCA and Fisherfaces). In those similarity measures, L_1 defines the city-block distance, L_2 defines the Euclidean distance. Cosine, Dice, Overlap and Jaccard measure the relative overlay between two vectors. L_1, L_2 and cosine can also be weighted by the covariance matrix of training data, which leads to the Mahalanobis related distances. Our findings indicate that Mahalanobis related

similarity distances are superior to others when expressive features (driven by PCA) are used; while overlay related similarity distances are superior when discriminating features are used.   *Open Set TCM-kNN*, which is an instantiation of TCM-kNN appropriate for the watch list face surveillance problem, is described in the next section.

3   Open Set TCM-kNN

The *watch list face surveillance* problem operates under the *open set* assumption that not all the probes have mates in the gallery and it requires a *reject* option. The standard *Open Set* PCA and Fisherfaces classifiers learn the operational threshold from the intra- and inter-distance (similarity) distribution of each training exemplar. The statistics of intra-distance distribution set the lower bound of the threshold and the statistics of inter-distance distribution set the upper bound. As the minimum distance of the new exemplar to the prototypes of each subject becomes closer to or larger than the upper bound, the more likely the new testing exemplar will be rejected. Our experiments have shown that face recognition performance varies according to the threshold chosen. It is, however, not easy to determine the threshold needed for rejection even if the lower and upper bound are known.   In addition, one needs to make a strong assumption that the distribution of the similarity distances is similar across different training and test sets. Those problems are overcome by the *Open Set TCM-kNN* classifier and the rationale is explained next.

Given a new exemplar, the *p*-value vector output from Open Set TCM-kNN gives the likelihoods that the new exemplar belongs to each subject in the training data. If some *p*-value significantly outscores the others, the new exemplar can be mated to the corresponding subject ID; otherwise, the probe is equally likely to be mated to each class and it should be rejected due to its ambiguity. If the top ranked (highest *p*-values) choices are very close to each other and outscore the other choices, the top choice should be accepted, i.e., should not be rejected, but its recognition be questionable due to the ambiguity involved. The ambiguity is measured by the *PSR* (peak-side-ratio) value and it determines if the test exemplar should be rejected or not. The Open Set TCM-kNN algorithm is:

```
Open Set TCM-kNN Algorithm

Calculate the alpha values for all training exemplars;
for i = 1 to c do
 for every training exemplar t classified as i do;
  for j = 1 to c and j!=i do
   Assume t is classified as j,which should be rejected
   Recalculate the alpha values for all the training
    exemplars classified as non-j;
   Calculate alpha value for t if it is classified as j
   Calculate p-value for t if it is classified as j
  end for
  Calculate the Pmax, Pmean and Pstdev(standard deviation)of
   exemplar t's p-values
  Calculate the PSR value of t: PSR=(Pmax- Pmean)/Pstdev
 end for
end for
```

```
Calculate the mean, stdev (standard deviation) for all the
PSR values
Calculate the rejection threshold = mean + 3*stdev
Calculate the distance of the probe s from all training
exemplars
for i = 1 to c do
 Calculate alpha value for s if it is classified as i
 Calculate p-value for s if it is classified as i
end for
Calculate the largest p-value max for s
Calculate the mean and stdev of probe p-values without max
Calculate the PSR value for the probe exemplar:
  PSR = (max-mean)/stdev
Reject the probe s if its PSR is less than or equal to the
threshold
Otherwise predict the class with the largest p-value
Output as prediction confidence 1st largest p-value or one
minus the 2nd largest p-value
Output as prediction credibility the largest p-value
```

In the algorithm, the threshold for rejection is learned from the training data set based on transductive inference. For each training exemplar X its mates are removed from the training data set, X is now treated as a probe, and Open Set TCM-kNN outputs the p-values for X. The corresponding PSR value for X is PSR = $(P_{max} - P_{mean}) / P_{stdev}$. Very low PSR values result for X since it should be rejected because it lacks any mates. The training exemplars' PSR distribution provides thus a robust method for deriving the operating threshold.

4   Data Collection and Experimental Results

Fig. 1. Face Images

Our data set is drawn from the FERET database, which has become a de facto standard for evaluating face recognition technologies (Phillips et al., 1998). The data set consists of 600 FERET frontal face images corresponding to 200 subjects, which were acquired under variable illumination and facial expressions. Each subject has three normalized (zero mean and unit variance) images of size 128 x 128 with 256 gray scale levels. Each column in Fig. 1. corresponds to one subject. The normalized face images are processed to yield 400 PCA coefficients and 200 Fisherfaces using FLD (Fisher Linear Discriminant) on the reduced 200 dimensional PCA space.

TCM-kNN yields similar performance with kNN on the closed set face recognition problem except that its output includes confidence and credibility for each decision made. The best similarity distances for PCA and Fisherfaces representations were found to be {Mahalanobis + (L_1, L_2 or cosine)} and {cosine, Dice, Jaccard, (Mahalonobis + cos)}, respectively. Only Mahalanobis + L_1 and cosine distances, which yield slightly better performance than the other distances, are used by Open Set TCM-kNN for the watch list face surveillance experiments reported next.
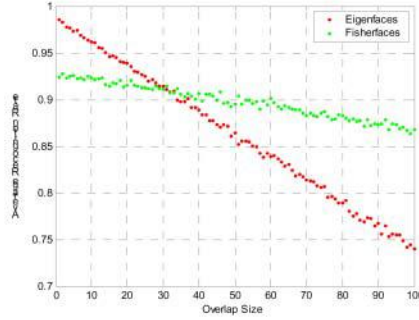
Fig. 2. Mean Detection and Identification Rates vs. Overlap Size

Fig. 2 shows the mean detection and recognition rates for PCA and Fisherfaces components for different overlap percentages (between the gallery [watch list] and the probe sets) using Mahalanobis + L_1 and cosine distance, respectively. There are 200 subjects (see Sect. 4), two images are used for training while the third image is available for testing, the size for both the watch list [gallery] and the probe sets is 100 subjects, and the overlap between the watch list and the probe set varies from 0 to 100 subjects. The detection and recognition rate is the percentage of subjects whose probe is correctly detected (or rejected) and identified. We report the average results obtained over 100 randomized runs. The performance goes down, almost linearly, as the overlap between the watch list and the probe sets increases. One gets overall better performance when using Fisherfaces components compared to PCA components; for small overlap, i.e., the small watch list size, which is most relevant, PCA yields slightly better performance than Fisherfaces. The explanation for the observed performance is that as both the size of watch list increases it becomes more difficult to detect and identify individuals on the watch list. This confirms the FRVT2002 findings (Phillips et al., 2003).

The next experiment uses the Open Set and Open Set TCM-kNN classifiers on small watch lists, whose size varies from 10 to 40 subjects, and reports the mean performance (detection and identification) rates obtained over 100 randomized runs. Let the watch list size be k subjects, each of them having 2 (two) images in the gallery. Then there are $600 - 2k$ face images in the probe set, k stands for the number of subjects on the watch list and 3 × (200 - k) stands for the number of face images that come from subjects that are not on the watch list.

Table 1 and 2 shows the mean performance of Open Set and Open Set TCM-kNN algorithms for different watch list sizes. For watch list size k, the accuracy [detection and identification rate] is (*average correct rejection + average correct recognition*) /

(*600 - 2k*). The numerical results when the number of subjects on the watch list is *k* should be interpreted as follows. The average results are better the closer the *correct rejection* number is to *3 × (200 – k)*, the closer the *correct recognition* number is to the watch list size and the higher the *accuracy* is.

Table 1. Mean Performance of Open Set Algorithm

Watch List Size	Eigenfaces (Mah+L_1 distance)			Fisherfaces (Cosine distance)		
	Average Correct Reject	Average Correct Recognition	Accuracy	Average Correct Reject	Average Correct Recognition	Accuracy
10	546.38	5.67	95.18%	550.00	7.98	96.20%
20	494.08	12.29	90.42%	502.36	15.93	92.55%
30	448.11	18.98	86.50%	457.83	24.41	89.30%
40	407.78	22.94	82.83%	416.05	32.24	86.21%

Table 2. Mean Performance of Open Set TCM-kNN

Watch List Size	Eigenfaces (Mah+L_1 distance)			Fisherfaces (Cosine distance)		
	Average Correct Reject	Average Correct Recognition	Accuracy	Average Correct Reject	Average Correct Recognition	Accuracy
10	536.62	8.67	94.02%	544.64	8.99	95.45%
20	522.38	16.97	96.31%	522.35	17.88	96.47%
30	495.65	25.16	96.44%	493.33	26.76	96.31%
40	467.39	33.42	96.31%	464.56	35.39	96.14%

The lower and upper bounds for the reject threshold for the Open Set algorithm are computed from the training set. The average correct rejection and identification numbers are recorded by varying the threshold from the lower bound to the upper bound. Since the watch list size is much smaller than the number of the subjects that should be rejected, the accuracy will be very high even if all the probes are rejected. Therefore, the average correct reject and recognition numbers as well as the detection and identification accuracy are needed to evaluate the performance of the algorithm. As an example, Table 1 shows that only about 60% (PCA) and 80% (Fisherfaces) of the subjects from the watch list (*average correct recognition /watch list size*) are recognized correctly even when the accuracy is high (about 96% for *k* = 10). Table 1 shows the results with the best threshold considering those three factors. As the watch list size increases, both the average rejection number, further away from *3 × (200 – k)*, and the accuracy, drop dramatically.

    Table 2 shows the average performance of Open Set TCM-kNN algorithm for different watch list sizes. PCA is very close to Fisherfaces in overall performance. If the data in the table is examined carefully, PCA is a little better than Fisherfaces with the reject option while Fisherfaces is a little better than PCA when the decision of

identification is made. Open Set TCM-kNN is much better than Open Set algorithm, when the correct rejection and correct recognition number as well as the accuracy are taken into account, especially when the watch list size is large.   The overall performance for Open Set TCM-kNN, which keeps almost constant as the watch list size increases, is thus much more stable than the overall performance displayed by the Open Set algorithm.

The difference in performance shown by Fig. 2 and Table 2 indicates that the gallery size is also an important factor affecting algorithm performance. In Fig. 2 the gallery [watch list] size is always 100 subjects and only the overlap size between the gallery and probe sets changes, while in Table 2 the gallery size [watch list] varies according to k.

5   Conclusions

This paper introduced the *Open Set TCM-kNN* (Transduction Confidence Machine – k Nearest Neighbor) for the *watch list face surveillance* task. Open Set TCM-kNN, which is based upon algorithmic randomness and transductive inference, has been successfully validated on the (small) watch list task (80% or more of the probes lack mates) using FERET datasets characterized by challenging (varying illumination and changing facial expression) face images. In particular, Open Set TCM-kNN provides on the average 96% correct detection / rejection and identification accuracy using the PCA and/or Fisherfaces   components for face representation.

References

[1]   Gammerman, V. Vovk, and V. Vapnik (1998), Learning by Transduction. In *Uncertainty in Artificial Intelligence*, 148 – 155.
[2]   Nouretdinov, T. Melluish, and V. Vovk (2001), Ridge Regression Confidence Machine, *Proc. 17th Int. Conf. on Machine Learning.*
[3]   S. Pankanti, R. M. Bolle and A. Jain (2000), Biometrics-the-Future of Identification, *Computer*, Vol. **33**, No. 2, 46 – 49.
[4]   P. J. Phillips, H. Wechsler, J. Huang, and P. Rauss (1998), The FERET Database and Evaluation Procedure for Face Recognition Algorithms, *Image and Vision Computing* , Vol. 16, No. 5, 295-306.
[5]   P. J. Phillips, P. Grother, R. J. Micheals, D. M. Blackburn, E. Tabassi and M. Bone (2003), Face Recognition Vendor Test 2002 – Overview and Summary.
[6]   K. Proedrou, I. Nouretdinov, V. Vovk and A. Gammerman (2001), Transductive Confidence Machines for Pattern Recognition, TR CLRC-TR-01-02, Royal Holloway University of London.
[7]   V. Vapnik (1998)*, Statistical Learning Theory*, Wiley.

Matching 2.5D Scans for Face Recognition

Xiaoguang Lu, Dirk Colbry, and Anil K. Jain

Department of Computer Science & Engineering,
Michigan State University, East Lansing, MI 48824
{lvxiaogu,colbrydi,jain}@cse.msu.edu

Abstract. The performance of face recognition systems that use two-dimensional images is dependent on consistent conditions such as lighting, pose, and facial appearance. We are developing a face recognition system that uses three-dimensional depth information to make the system more robust to these arbitrary conditions. We have developed a face matching system that automatically correlates points in three dimensions between two 2.5D range images of different views. A hybrid Iterative Closest Point (ICP) scheme is proposed to integrate two classical ICP algorithms for fine registration of the two scans. A robust similarity metric is defined for matching purpose. Results are provided on a preliminary database of 10 subjects (one training image per subject) containing frontal face images of neutral expression with a testing database of 63 scans that varied in pose, expression and lighting.

1   Introduction

Automatic human face recognition has gained a lot of attention during the last decade [1]. A few approaches utilized depth information provided by 2.5D range images [2-5], but most efforts have been devoted to face recognition from only two-dimensional (2D) images [1]. Current 2D face recognition systems can achieve good performance in constrained environments, however, they still encounter difficulties in handling large variations in pose and illumination [6]. Utilizing 3D information can improve face recognition performance [6, 7]. Range images captured by 3D sensor [8, 9] explicitly represent face surface shape information as well as providing registered color texture images. Face recognition based on range images has been addressed in different ways [2-4, 10, 11].
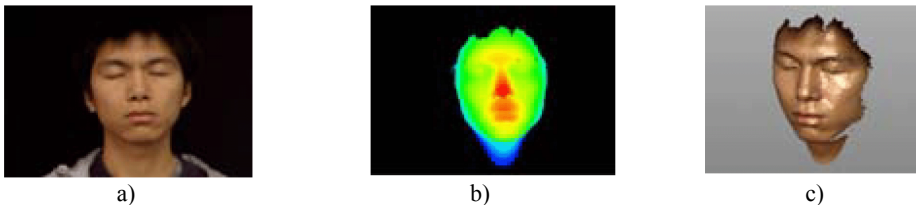
Fig. 1. An example for Minolta Vivid 910 facial scan. (a) Texture image; (b) range image, the more red a point, the closer it is to the sensor; (c) 3D rendering of the face scan
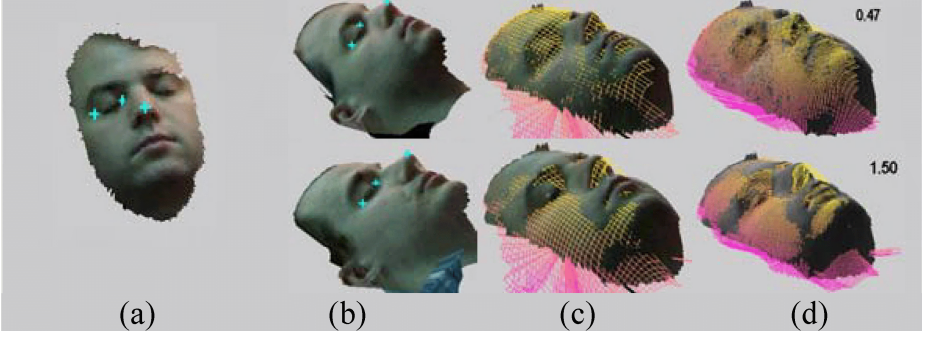
Fig. 2. Matching a given 2.5D face scan to 2.5D face scans in a database. a) Automatic feature point detection; b) the genuine scan and an imposter scan in the database with feature points labeled. c) Coarse alignment between the test scan and the gallery scan. d) Fine iterative registration to find the closest match

In this research 2.5D range images are used for face recognition. A 2.5D image is a simplified three-dimensional (x, y, z) surface representation that contains at most one depth (z) value for every point in the (x, y) plane (see Figure 1). A full three-dimensional model could account for a more robust recognition system, however, there is an added cost to construct and store the complete 3D face model.

2   Face Alignment and Matching

Our face recognition system is illustrated in Fig. 2.

2.1 Automatic Feature Point Localization and Coarse Alignment

A minimum of three corresponding points is needed in order to calculate the rigid transformation between two 2.5D scans. Once the three corresponding points are known, the transformation can be made using a combination of rigid transformation matrices following the guidelines described by Weinstein [12].

Using the procedure described by Dorai and Jain [13], we determine the local shape information at each point within the 2.5D image. This shape index at point p is calculated using the maximum (κ_1) and minimum (κ_2) local curvature (see Equation 1). This calculation produces a shape scale between zero and one. A value near one represents a dome shape and a value close to zero represents a cup or dent shape. A value in the middle (0.5) represents a saddle point.

$$S(p) = \frac{1}{2} - \frac{1}{\pi} \tan^{-1} \frac{\kappa_1(p) + \kappa_2(p)}{\kappa_1(p) - \kappa_2(p)} \tag{1}$$

This shape calculation is independent of the coordinate system and, therefore, is a potentially useful metric for finding similar points between scans with different poses. The faces shown in Figure 3(b) are examples of 2.5D face images with the grayscale representing the shape index at a particular location, a dark point has a shape index close to zero and a lighter point has a shape index close to one.
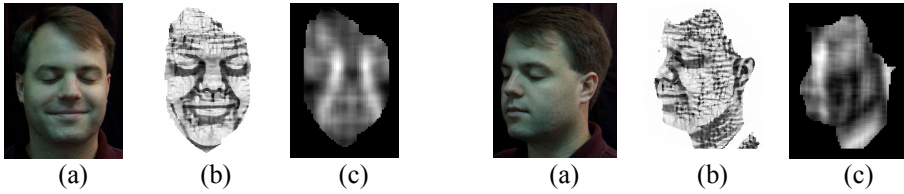
Fig. 3. (a) Texture Image; (b) shape index; (c) shape index after applying 30x10 averaging mask (compared with (b), the shape index value is inversed for better illustration)

For simplicity, we pick a combination of the inside of an eye, the outside of the eye and the nose tip, which vary, based on the pose of the range image. Of these, the easiest point to identify is the inside edge of an eye right next to the bridge of the nose, because this point (and the area around this point) has a shape index value that is very close to zero. A simple averaging mask of size 30 x 10 can be used to identify this area on the shape space (See Fig. 3.(c)). We use the inside edge of the eye as an anchor in order to identify other feature points within the face image. Some failures in our system occur when these points are misclassified.

Once the inside eyes are found, other feature points within the face can be located using a simple model of the relative locations between the points. The outside of the eye is detected by following the rut (defined by the shape space) that consistently runs along the bottom of the eye. The mouth tips are found by looking for a valley in the shape space below the eyes. The mouth tips are not robust features because of the movement of the mouth. Despite this, the inside eye, outside eye and a mouth point form a plane that can be used to reliably locate the nose tip which is the farthest point from the plane formed by these three points.

2.2 Fine Alignment with Hybrid ICP

After coarse alignment the scans are not registered well due to the localization errors of the feature points. The fine alignment process can be formulated as follows: Given two partially overlapping sets of 3D points, P and Q, roughly aligned, find the best (rigid) transformation Ψ to align them so that the the error function $E = Dist(\Psi(P), Q)$ is minimized, where the $Dist(.)$ is the distance metric. Our fine registration process follows the Iterative Closest Point (ICP) framework [14-16]. With an initial estimate of the rigid transformation, the algorithm iteratively refines the transform by alternately choosing corresponding (control) points in two scans and finding the best rigid transformation that minimizes an error function based on the distance between them. In Besl and McKay [14], the point-to-point distance is applied and a fast close-form solution is provided when calculating the transformation matrix during each iteration. The point-to-plane distance in Chen and Medioni [15] makes the ICP algorithm less susceptible to local minima than the point-to-point metric [14, 17]. This algorithm shows that if the two meshes are close to each other, the point-to-plane distance is the best approximation for the true distance between the two surfaces [18]. But calculating the point-to-plane distance takes longer than the point-to-point distance. We integrate these two classical ICP algorithms [14, 15] in a zigzag running style, which is called hybrid ICP algorithm. Besl's scheme is used to compute an

estimation of the alignment, followed by Chen's scheme for a refinement. Since the distance metrics are utilized interactively, it is possible to achieve a better registration result than each individual. The control points are selected only within the overlapping area to avoid outliers. A scheme similar to RANSAC [19, 20] is applied. Each time a number of control points are chosen randomly and ICP is applied based on the points selected. This procedure of control point selection and applying ICP is repeated several times. We pick the alignment with the minimum registration error as the final result.

2.3 Matching

In our experiments, the intensity (color), shape index, along with the 3D Cartesian coordinates are used for facial scan matching. One way to define the similarity metric is to combine these attributes into a single feature vector. Each attribute has relatively independent interpretation so they are treated separately in this matching scheme. The N control points used in the fine registration stage are represented by p_i, i=1,2…N.

- **Surface matching.** Surface matching in fine registration stage provides the registration error, which can be used as a matching score between two facial scans.

$$E_D = \sqrt{\frac{1}{N}\sum_{i=1}^{N} D(\Psi(p_i), S_i)} \tag{2}$$

  where Ψ is the transformation, $D(.)$ represent the point-to-plane distance between transformed point $\Psi(p_i)$ and the tangent plane S_i at the position of its counterpart in the other scan surface [15].

- **Texture matching.** The 2.5D surface registration from two scans also registers the corresponding color texture maps. Let p_i (i=1,2…N) denote the control points in scan P, and q_i be the closest counterpart in scan Q after the transformation Ψ is applied to P. Let $I(p_i)$ be the texture intensity at p_i. Then the texture vectors of P and Q are represented as $T_P = \{I(p_i), i=1,2…N\}$ and $T_Q = \{I(q_i), i=1,2…N\}$, respectively. The matching score between two registered texture vectors is calculated as follows, $MS_T = <T_P, T_Q > /(\|T_P\| \cdot \|T_Q\|)$, where T_P and T_Q are the two texture vectors, $< \cdot >$ denotes the dot-product, $\|\cdot\|$ represents the norm of the texture vector. The MS_T is normalized to a range from 0 to 1.

- **Shape Index Matching.** Each point in the facial scan is associated with a shape index calculated by Eq. 1. Similar to the texture matching, the matching score based on the shape index attribute can be defined as, $MS_T = < S_P, S_Q > /(\|S_P\| \cdot \|S_Q\|)$, where S_P and S_Q are the two shape index vectors of the control point sets. The MS_S is also normalized to range from 0 to 1.

The surface matching distance, the registered texture matching score, and the registered shape index matching score can be integrated in a cascade mode or in a combination mode. The range information is more robust than the texture information under varying pose and lighting conditions. The shape index attribute is subject to estimation errors due to the discretization of the range data. Therefore, the surface matching distance is considered as the primary dissimilarity metric. In the cascade

mode, only the test scan, whose surface matching distance is low enough, will proceed to the texture and shape index matching stages. In the combination mode, the texture and shape index matching scores are used to weight the surface matching distance. The final matching distance is computed as:

$$MD = E_D \cdot (1 - MS_T) \cdot (1 - MS_S) \tag{3}$$

The smaller the *MD*, the better the matching. The combined matching distance can be considered as a combination of range and texture evidence by the product rule [21].

3   Experiments

All range images were collected using a Minolta Vivid 910 scanner [9], which uses structured laser light to record the face scans in less than a second. Each point in a scan has a texture color (r, b, g) as well as a location in 3-space (x, y, z). This data is stored in individual (*p×q*) matrices. Some of the data points are not valid (e.g., points outside the range of the scanner) so there is also a (*p×q*) binary flag matrix that is true for all the valid points in the scan. The range image is of size 480×640 in the fast scan mode of the Minolta sensor (with a reported depth accuracy of approximately 0.1 mm) and downsampled to 240×320 in our experiment to reduce the computation cost. The registered color texture image has the same size as the range image. There are 10 subjects in our collected database. The frontal scan with neutral expression for each subject was used to construct the training database. The test database consists of 63 scans of the same 10 people, which varied in pose (up to left and right 60 degrees from the frontal view), facial expression and lighting.

In order to properly evaluate the performance of the system, it is tested in two experimental modes; automatic and manual. In automatic mode, the algorithm selects the feature points used in coarse registration, where as in the manual mode, the feature points are labeled by hand. The error between the manually selected feature points and the automatically selected feature points is 18.12mm with a standard deviation of 37.58mm. The error is calculated by taking the average distance between the manual labeled points and the automatically extracted points.

The fine alignment and face recognition system has been tested on both manually selected feature points and automatically selected feature points. The recognition accuracy is given in Table 1. Besides the hybrid ICP, two classical ICP algorithms are presented for comparison. Table 1 provides a detailed accuracy for the hybrid ICP, which achieves the best performance among three ICP schemes.

One error out of 15 frontal test scans was achieved when comparing frontal 2.5D face images. This is noteworthy considering that many of the test images had people with different expressions than the training images (i.e., some were smiling). When the testing database was extended to include semi-profile images, the error went up but still maintained a reasonable recognition rate of 5 errors out 63 (92% accuracy). These results assume that the coarse alignment procedure is correctly done. The combined total error of the entire automatic system is 10 out of 63 (84% accuracy). The hybrid ICP scheme is more robust to the localization errors of feature points, which outperforms the other two classical ICP algorithms on the surface matching (range image only). The combination of surface, texture and shape index matching achieves better performance than surface matching based on the range image alone.

Table 1. Number of rank-one failed tests due to recognition. The combination of range, texture and shape index matching is computed as Eq. 3. The number of iterations (10,10) represents a total of 10 iterations for Besl's ICP scheme and 10 iterations for Chen's scheme

Algorithm	Number of Iterations	Feature Point Extraction	Range image only	Range + Texture + Shape Index
ICP_hybrid	(10, 10)	Manual	6	5
ICP_hybrid	(10, 10)	Auto	13	10
ICP_Chen [15]	20	Manual	10	6
ICP_Chen [15]	20	Auto	16	14
ICP_Besl [14]	20	Manual	19	5
ICP_Besl [14]	20	Auto	33	15

4   Conclusions and Future Work

We have presented a face recognition system that is more robust to variations in pose and lighting. This system can match 2.5D scans of arbitrary poses with lighting and expression variations to a database of neutral expression frontal 2.5D scans. Unlike the appearance-based method [22], the proposed matching scheme does not need an extensive training process based on a collected large-scale database.

A Hybrid ICP scheme is proposed to integrate two classical ICP algorithms to achieve better registration performance. Besides the range image, other attributes, such as the registered texture and shape index information, are also used to design a combined metric for the facial scan matching.

This research is an encouraging first step in designing a system that is capable of recognizing faces with arbitrary pose and illumination. There are a number of directions that we are planning to consider in our future work. The first will be to improve the automatic coarse recognition system by incorporating a more complete model of the human face. We also plan on increasing the training database to include profile images. This should make a more robust template set and increase the systems matching capabilities. With a better model we will also consider methods for matching arbitrary two-dimensional training data.

Acknowledgement

Reference

[1]   Zhao, W., R. Chellappa, A. Rosenfeld and P. J. Phillips. "Face Recognition: A Literature Survey," *CVL Technical Report,* University of Maryland, Oct. 2000.
[2]   Gordon, G. "Face Recognition Based on Depth and Curvature Features," *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp. 108-110, 1992.

[3]   Lee, J. and E. Millios, "Matching Range Images of Human Faces," *Proc. ICCV*, pp. 722-726, 1990.

[4]   Tanaka, H., M. Ikeda and H. Chiaki. "Curvature-based face surface recognition using spherical correlation," *Proc. Third Int. Conf. on FG*, pp. 372-377, 1998.

[5]   Chua, C., F. Han and Y. Ho. "3D Human Face Recognition Using Point Signature," *Proc. Fourth Int. Conf. on FG*, pp. 233-238, Grenoble, March 2000.

[6]   "Face Recognition Vendor Test (FRVT)," http://www.frvt.org/.

[7]   Blanz, V. and T. Vetter, "Face Recognition Based on Fitting a 3D Morphable Model," *IEEE Trans. PAMI*, vol. 25, no. 9, pp. 1063-1074, 2003.

[8]   "Cyberware Inc.," http://www.cyberware.com/.

[9]   "Minolta Vivid 910 non-contact 3D laser scanner," http://www.minoltausa.com/vivid/.

[10]   Beumier, C. and M. Acheroy, "Automatic 3D face Authentication," *Image and Vision Computing*, vol. 18, no. 4, pp. 315-321, 2000.

[11]   Pan, G., Z. Wu and Y. Pan. "Automatic 3D Face Verification From Range Data,"" *Proc. IEEE ICASSP*, pp. 193-196, 2003.

[12]   Weinstein, D. M. "The Analytic 3-D Transform for the Least-Squared Fit of Three Pairs of Corresponding Points," *School of Computing Technical Report,* No. UUCS-98-005*, University of Utah, Salt Lake City, UT, March 1998.

[13]   Dorai, C. and A. K. Jain, "COSMOS - A Representation Scheme for 3D Free-Form Objects," *IEEE Trans. on PAMI*, vol. 19, no. 10, pp. 1115-1130, 1997.

[14]   Besl, P. and N. McKay, "A Method for Registration of 3-D Shapes," *IEEE Trans. PAMI*, vol. 14, no. 2, pp. 239-256, 1992.

[15]   Chen, Y. and G. Medioni, "Object Modeling by Registration of Multiple Range Images," *Image and Vision Computing*, vol. 10, no. 3, pp. 145-155, 1992.

[16]   Zhang, Z., "Iterative point matching for registration of free-form curves and surfaces," *International Journal of Computer Vision*, vol. 13, no. 1, pp. 119-152, 1994.

[17]   Gelfand, N., L. Ikemoto, S. Rusinkiewicz and M. Levoy. "Geometrically Stable Sampling for the ICP Algorithm," *Proc. International Conference on 3D Digital Imaging and Modeling*, pp. 260-267, Banff, October 2003.

[18]   Pottman, H. and M. Hofer. "Geometry of the squared distance function to curves and surfaces," *Vienna Institute of Technology Technical Report,* No. 90, January 2002.

[19]   Fischler, M. and R. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381-395, 1981.

[20]   Chen, C. S., Y. P. Hung and J. B. Cheung, "RANSAC-based darces: a new approach to fast automatic registration of partially overlapping range images," *IEEE Trans. PAMI*, vol. 21, no. 11, pp. 1229-1234, 1999.

[21]   Kittler, J., M. Hatef, R. Duin and J. Matas, "On Combining Classifiers," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226-239, 1998.

[22]   Chang, K. I., K. W. Bowyer and P. J. Flynn. "Multi-Modal 2D and 3D Biometrics for Face Recognition," *Proc. IEEE Workshop on AMFG*, France, October 2003.

Constructing SVM Multiple Tree
for Face Membership Authentication

Shaoning Pang

Knowledge Engineering & Discover Research Institute
Auckland University of Technology
Private Bag 92006, Auckland 1020, New Zealand
`snpang@aut.ac.nz`

Abstract. In membership authentication problem that has a complicated and mixed data distribution, the authentication accuracy obtained from using one classifier is not sufficient despite its powerful classification ability. To overcome this limitation, an support vector machine (SVM) multiple tree is developed in this paper according to a "divide and conquer" strategy. It is demonstrated that the proposed method shows a good membership authentication performance, as well as the strong robustness to the variations of group membership, as compared with the SVM ensemble method [1]. Specifically, the proposed method shows a better improvement in authentication performance as the group size increases larger.

1   Introduction

Membership authentication problem can be depicted as follows, consider a certain human group G with N members, which is the universal set. If there exists an arbitrary subgroup M such that $M \subset G$ and $|M| < N/2$, then it is a membership group, and the remaining people $\overline{M} = G - M$ is a non-membership group. Thus, membership authentication problem is to distinguish the membership class M from the non-membership class $\overline{M}$ in the human group.

Unlike most works on face recognition that tried to identify the identity of the given face image, I proposed a concept of dynamic membership authentication [1] that attempts to authenticate an individual's membership without revealing the individual's identity and without restricting the group size and/or the members of the group, implemented the concept using an SVM ensemble method.

One problem of SVM ensemble method is that the correct classification rate for the membership is not good as to be expected, as the size of membership group is bigger than 45 (16.6% percentage of total group). This is due to a complicated mixed data distribution among the membership and non-membership face images, as it is very difficult to discriminate such complicated mixed data in terms of only one classifier even its classification performance is powerful.

To solve this problem, a new membership authentication method of SVM multiple tree is developed in terms of a "divide and conquer" strategy, where

the whole data is divided into several subgroups by an iterative membership-based data splitting and authenticate the membership in each subspace by SVM classification. Consequently, a considerable improvement is obtained in robustness to the size changes in the membership group.

2   SVM Multiple Tree

The SVM multiple tree is constructed by a divide and conquer approach. Mathematically, the proposed membership authentication method can be formulated as follows.

$$F(\mathbf{x}) = \begin{cases} 1 & if\ f_{g_i}(x) = 1, \mathbf{x} \in g_i\ i = 1..L \\ -1 & if\ f_{g_i}(x) = -1, \mathbf{x} \in \bar{g}_i \end{cases} \tag{1}$$

where the total group (G) is divided into subgroups $\{g_1, g_2, \cdots, g_L\}$, and each subgroup g_i belongs to membership class or nonmembership class. f_{g_i} is a subgroup membership classifier working on g_i. Hence, the whole membership authentication system $F(\mathbf{x})$ consists of numbers of subgroup membership classifiers, each input face $\mathbf{x}$ is first judged to be the member of a certain subgroup, then the class of $\mathbf{x}$ is determined by the label of this subgroup.

2.1   Tree Growing

Similar to decision tree, the glowing of SVM tree is determined by a succession of splits that partition the training data into disjoint subsets. Starting from the root node that contains all training data, a membership-based PCA data splitting is performed to partition the data into numbers of disjoint subsets. These subsets are represented by the same number of child nodes originating from the root nodes, and the same splitting method is applied to each child node. This recursive procedure terminates when the subdivided clusters have only either membership or nonmembership training data.

Basically, two procedures are performed at each node in the above tree generation. First, the membership-based data splitting performs a 'soft' separation of membership because it splits the data into disjoint subsets. Next, a multiclass SVM classifier is trained by the result of 'soft' classification and can be thought of as performing 'hard' decision of membership. Consequently, with the continuing of data splitting, a subspace hierarchy is built up, and a SVM multiple tree is constructed thereafter, where each terminal node of the tree is associated with a label of membership or nonmembership. Algorithm 1 shows the procedure of constructing the SVM multiple tree that includes the membership-based clustering.

Therefore, for an input face, its membership can be predicted in the above SVM decision tree. First, a new data $\mathbf{x}$ is feed to root node SVM tester $T_1(\mathbf{x})$ in the SVM tree. Depending the result of a decision made by an internal node SVM test $T_i(\mathbf{x})$, the tree will branch to one of the node's children. This procedure is repeated until a terminal node is reached and a membership label is then assigned to the given input face. This procedure is depicted as Algorithm 2.

Algorithm 1: SVM multiple tree construction.
Function SVMTree_Build (A training set X) {
  if (X contains the same membership) {
    Mark the end of a branch in SVM tree T;
    return;}
  Multiple_Membership-based_Splitting(X;$X_1, ..., X_q$);
   /*Divide X into disjoint subsets $X_1, ..., X_q$ */
  Append_node(T_i);
  Train_SVM_Classifier(X);
  for($j = 1; j < q; j + +$) {SVMTree_Build(X_j);}
}

Algorithm 2: Authentication by SVM multiple tree.
Function SVMTree_Test (SVM multiple tree T, Input face $\mathbf{x}$) {
  current=1;   /* set current node is root of T */
  while ($T_{current}$ is an inside node) {
     SVM_Test_$T_{current}$($\mathbf{x}$);
     next = Search_next_node($T_{current}$);
     current = next;
   /* set next node as current node*/
    }
  return $T_{current}$($\mathbf{x}$);
}

2.2 Data Splitting

In PCA, membership can be represented as a set of eigenfaces together characterized the variation of between all the group member face images. So are the non-group member faces. Thus the training set can be partitioned by a data splitting procedure with the membership eigenfaces and nonmembership eigenfaces as reference centers.

The computation of the eigenface is based on principal component analysis, which finds the optimal basis for representation of the training data space in the mean squared error sense[2]. Let the training set $I = \{\mathbf{x}_1, x_2, x_3, ..., x_M\}$ of N-dimensional face feature vectors. The mean, μ, and the covariance matrix,$\boldsymbol{\Sigma}$, of the data are given by:

$$\mu = \frac{1}{M} \sum_{m=1}^{M} \mathbf{x}_m \tag{2}$$

$$\boldsymbol{\Sigma} = \frac{1}{M} \sum_{m=1}^{M} [\mathbf{x}_m - \mu][\mathbf{x}_m - \mu]^T = \mathbf{U}\mathbf{U}^T. \tag{3}$$

Where $\boldsymbol{\Sigma}$ is an $N \times N$ symmetric matrix. This matrix characterizes the scatter of the data set. A none-zero vector $\mathbf{u}_k$ for which

$$\boldsymbol{\Sigma}\mathbf{u}_k = \lambda_k \mathbf{u}_k \tag{4}$$

is a principal component of the face feature space from I. This is often referred as eigenface[2]. Its weight is λ_k, which represents the contribution of this component in the original space. If $\lambda_1, \lambda_2, ..., \lambda_K$ are the K largest weight, then $\mathbf{U} = [\mathbf{u}_1\mathbf{u}_2...\mathbf{u}_k]$ are the representative eigenfaces of data set I.

In membership authentication, the group members can be divided into membership and non-membership group members as $G = M \cup \bar{M}$. Applying the above eigenface technique Eq. (2) - Eq. (4) to M and $\bar{M}$ respectively, two representative eigenface sets are obtained as the membership eignfaces $\mathbf{U}_M = [\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_K]$ and the non-membership eigenfaces $\mathbf{U}_{\bar{M}} = [\bar{\mathbf{u}}_1, \bar{\mathbf{u}}_2, \cdots, \bar{\mathbf{u}}_L]$. They characterize the "membership-face" and "nonmembership-face" respectively.

For splitting the data following above steps, lets identify the two partitioned groups as an $c \times n$ binary matrix $\mathbf{V}$, where the element v_{ij} is 1 if the jth data point $\mathbf{x}_j$ belongs to group i, otherwise 0. Once cluster centers $\mathbf{U}_M$ and $\mathbf{U}_{\bar{M}}$ are fixed, then the data splitting based on membership can be performed as follows:

$$v_{ij} = \begin{cases} 1 \; if \; Min_{i=1}^{K}\mathbf{x} \cdot \mathbf{u}_i \leq Min_{j=1}^{L}\mathbf{x} \cdot \bar{\mathbf{u}}_j \\ 0 \; otherwise \end{cases} \tag{5}$$

Where $\mathbf{x} \cdot \mathbf{u}_i$ is the distance projected onto the membership eigenfaces ($\mathbf{U}_M$), and $\mathbf{x} \cdot \bar{\mathbf{u}}_j$ is the distance projected onto the non-membership eigenfaces ($\mathbf{U}_{\bar{M}}$). Thus, with the growing of the splitting, members and non-members are always driven into a separated subspace. Until arrive the terminal node, member and non-member can be clearly distinguished and clustered in many subgroups.

2.3   Multiclass SVM Classification

Support vector machine is a new and promising classification and regression technique proposed by Vapnik and his group at AT&T Bell laboratories[3]. For the multi-class classification, SVM can be extended in the following two ways. One method is called the "one-against-all" method, where we have as many SVMs as the number of classes. The ith SVM is trained from the training samples, where some examples contained in the ith class have "+1" labels, and other examples contained in the other classes have "-1" labels. Then, the decision function is

$$f(\boldsymbol{x}) = sign(\underset{j\in 1,2,...,C}{Max}((\boldsymbol{w}^j)^T \cdot \varphi(\boldsymbol{x}_j) + b_j)), \tag{6}$$

where C is the number of the classes. Another method is called the one-against-one method[15]. When the number of classes is C, this method constructs $\frac{C(C-1)}{2}$ SVM classifiers. The ijth SVM is trained from the training samples where some examples contained in the ith class have "+1" labels and other examples contained in the jth class have "-1" labels. The class decision in this type of multi-class classifier can be performed in the following two ways. The first decision is based on the "Max Wins" voting strategy, in which $\frac{C(C-1)}{2}$ binary SVM classifiers will vote for each class, and the winner class is the class with the maximum number of votes. The second method uses the tournament match, which reduces the classification time to a log scale.
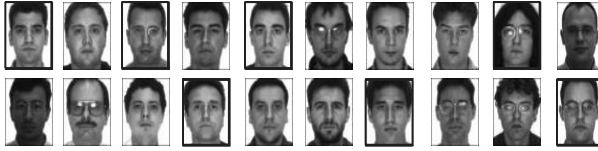
Fig. 1. Examples of face dataset for membership authentication

3   Experiments and Discussions

3.1   SVM Multiple Tree Construction

The proposed membership authentication technique has been implemented on the same face dataset as in [1]. For the construction of membership face group which are divided into a training and test set, a certain number of persons equal to the membership group are selected among 271 persons as group members, and the remaining persons are non-members. Particularly, to ensure the membership group meaningful, the size of membership group size is set to be less than the size of non-membership group, and the percentage of the group member over total persons to be within 40%. Fig. 1 are some examples of our face dataset for membership authentication, where persons in frame are assigned as member, and the remaining persons are nonmembers.

Algorithm 1 is applied to construct the SVM multiple tree for membership authentication. Fig. 2 illustrates a SVM-classification multiple tree for 10 member membership authentication that was implemented in the experiment. Each internal node of the tree identifies a multi-class SVM classifier, which is represented as as an ellipse with a label of a_b, where a is level of the node in the tree, and b is the number of brother node in the same level. The terminal node is represented as a circle or a filled circle, which denotes the class, membership or non-membership, respectively.

3.2   Comparison Experiments

The proposed method has been compared with the face recognition method and the SVM ensemble method on the membership authentication of two big group-size membership datasets, whose group sizes are 50 and 60 respectively, which are greater than the limitation of SVM ensemble method. The used face-identification method is embedded HMM with the 2nd-order block-specfic eigenvectors[4], which was reported, had the best recognition performance on the above face database. The SVM ensemble method is the previous work on dynamic membership authentication[1]. Table 1 shows the contrast statistical results, where SVM ensemble method is denoted as Proposed method I, and the proposed multiple SVM classification tree method as Proposed method II. Note that in Table 1, the proposed SVM classification tree provides a good ability of membership recognition, as well as a good system stability on these two
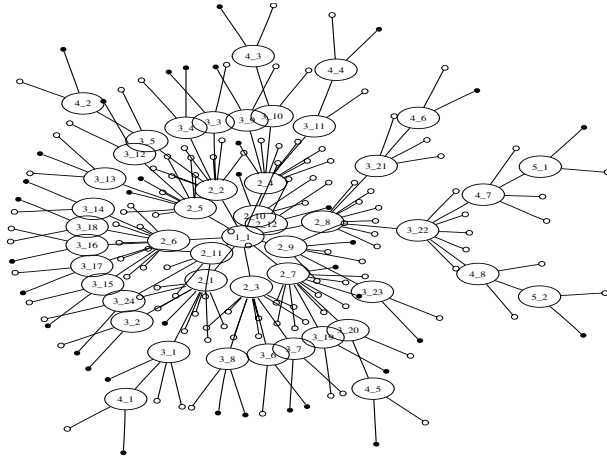
Fig. 2. An SVM multiple tree for 10 member membership authentication

Table 1. The comparison result of authentication error between two different authentication methods

	Proposed method I		Proposed method II		Identification-based
Group Size	50	60	50	60	method
Ex1(que:1,reg:2,3,4,5)	4.1%	5.8%	3.4%	3.7%	0.6%
Ex2(que:2,reg:1,3,4,5)	2.8%	11.5%	4.1%	4.8%	2.9%
Ex3(que:3,reg:1,2,4,5)	9.2%	3.7%	3.7%	4.1%	0.6%
Ex4(que:4,reg:1,2,3,5)	3.1%	13.3%	4.1%	4.8%	5.0%
Ex5(que:5,reg:1,2,3,4)	10.5%	23.7%	3.7%	4.1%	18.0%
Average	5.9%	11.6%	3.8%	4.3%	5.4%
Variance	130.0e-005	610.0e-005	0.9e-005	2.4e-005	530.0e-005

big group-size membership datasets. while the performance of SVM ensemble come to deteriorate. The correct rate becomes junior to face identification-based method, and the variance also jumps to the same grade as face identification method.

3.3   Stability Test

In the stability test of the proposed authentication method under the condition of different membership group sizes. The proposed authentication method is compared with the SVM ensemble method with different membership group sizes ranging from 5 to 95 persons with a step size of 5 persons. Fig. 3 shows the comparison result of the number of mis-authentications between two authentication methods, where two pieces of curve illustrate the stability of the
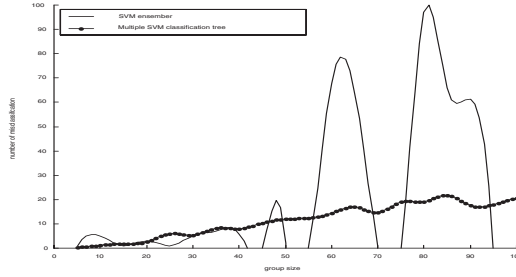
Fig. 3. Stability test under different group size

two methods under different membership group size, respectively. As can be seen, the proposed authentication method shows a good stability because it still shows a small number of mis-authentication as the membership group size grows greater than 45, and its curve fluctuates very slightly and smoothly. While the SVM ensemble-based method loses such stability, some mutation appears in the latter half part of its curve, which implies that the performance of system is out of control. It is evident that the proposed membership authentication method has a much better salability under the change of membership group size than the SVM ensemble method.

4   Conclusions

This paper presents a new membership authentication method by face classification using SVM multiple tree, in which the size of membership group and the members in the membership group can be changed dynamically. The experimental results show that the proposed SVM classification tree-based method not only keeps the good properties that the SVM ensemble method has, such as a good authentication accuracy and the robustness to the change of members, but also has a considerable improvement on the stability under the change of membership group size.

References

[1] S. N. Pang, D. Kim, and S. Y. Bang, "Membership authentication in the dynamic group by face classification using SVM ensemble," Pattern Recognition Letters, Vol. 24, pp. 215-225, 2003. 37, 41
[2] Pentland, A., Turk,M. (1999) Eigenfaces for Recognition Journal of Cognitive Neuroscience, 3(1):71-86. 39, 40
[3] V. Vapnik. Estimation of dependences based on empirical data. Springer-Verlag, 1982. 40
[4] Min-Sub Kim, Daijin Kim, Sung Yang Bang, Sang-Youn Lee, Young-Sik Choi(2002) Face Recognition Descriptor Using the Embedded HMM with the 2nd-order Block-specific Eigenvectors ISO/IEC JTC1/SC21/WG11/M7997, Jeju, March 2002. 41

A Combination of Shape and Texture Classifiers for a Face Verification System

Sanun Srisuk[1], Maria Petrou[2], Rerkchai Fooprateep[1], Khamron Sunat[1], Werasak Kurutach[1], and Pichet Chopaka[1]

[1] Mahanakorn University of Technology, Bangkok, Thailand
[2] University of Surrey, Guildford, United Kingdom, GU2 7XH
{sanun,rerkchai,khamron,werasak,cpichet}@mut.ac.th, m.petrou@surrey.ac.uk

Abstract. In this paper, we present a general framework for a combination of shape (shape Trace transform-STT) and texture (masked Trace transform-MTT) classifiers based on the features derived from the Trace transform. The MTT offers "texture" representation which is used to reduce the within-class variance, while STT provides "shape" characteristics which helps us maximize the between-class variance. In addition, weighted Trace transform (WTT) identifies the tracing lines of the MTT which produce similar values irrespective of intraclass variations. Shape and texture are integrated by a classifier combination algorithm. Our system is evaluated with experiments on the XM2VTS database using 2,360 face images.

1   Introduction

We propose a face authentication system using the Trace transform [1]. The original contributions of the proposed system are: 1) It offers the capability of constructing robust features from the Trace transform, which describe the characteristics of face images and do not necessarily have physical or geometric meaning. 2) It provides two important characteristics for face representation: shape and texture features. 3) It proposes a robust shape matching measure which is based on both spatial and structural information. The organization of this paper is as follows. Section 2 gives a brief overview of the Trace transform. We present shape and texture classifiers in section 3 and 4, respectively. We describe a reinforcement learning in section 5 with the training algorithm in section 6. We then propose a classifier combination algorithm in section 7, experimental results in section 8, and conclusion in section 9.

2   The Trace Transform

The Trace transform [1] is a new tool for image processing which can be used for recognizing objects under transformations. To produce the Trace transform one computes a functional T along tracing lines of an image. Each line is characterized by two parameters ϕ and p, and parameter t is defined along the line.

Table 1. Example of Trace functionals T

No.	Trace Functionals	Details
1	$T(f(t)) = \int_0^\infty f(t)dt$	
2	$T(f(t)) = \int_0^\infty \left\| \frac{d}{dt} f(t) \right\| dt$	differentiation means taking the difference of successive samples
3	$T(f(t)) = \int_c^\infty (t-c)^2 f(t)dt$	$c = \frac{1}{S}\int_0^\infty t\|f(t)\|dt,\ S = \int_0^\infty \|f(t)\|dt$
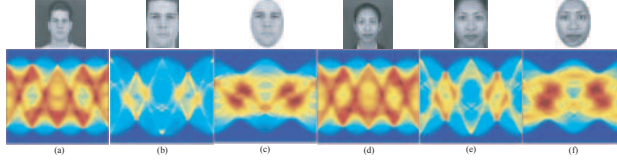
Fig. 1. Examples of the Trace transform with different shapes. (a) and (d) Full images; (b) and (e) Rectangular shapes, and (c) and (f) Elliptical shapes

With the Trace transform the image is transformed to another "image", which is a 2-D function $g(\phi, p) = T(F(\phi, p, t))$. In our current implementation we have 22 different trace functionals, 3 of which are shown in table 1.

2.1 Texture Representations of Face Images

The Trace transform is known to be able to pick up shape as well as texture characteristics of the object it is used to describe. We represent each extracted face inside an ellipse [5] as a masked face. We show the result of the Trace transform of the full image, the rectangular face and elliptical shape in Fig. 1. We call this face representation the masked Trace transform (MTT). The values of the MTT may be regarded as some sort of "texture" characteristics of a face image.

2.2 Shape Representations of Face Images

The Trace transform can be represented as shape characteristics by thresholding it with the appropriate values:

$$B(\phi, p) = \begin{cases} 1, & \text{if } g(\phi, p) \geq v, \\ 0, & \text{Otherwise,} \end{cases} \tag{1}$$

where v is some threshold. The goal of segmenting the shape from the MTT is to extract useful information, in the sense that each face must have its own shape in MTT which must be different from person to person.

3   Shape Classification

In this paper we use the shape matching measure as described in our previous work [6]. It can be briefly summarized as follows. The Hausdorff distance is a distance measure between two point sets A and B.

$$H(A, B) = \max(h(A, B), h(B, A)), \tag{2}$$

where $h(A, B)$ can be replaced by the Hausdorff Context

$$h_{HC}(A, B) = \sum_{a \in A} \underbrace{w(a, b')}_{\substack{\text{spatial} \\ \text{information}}} \min_{b \in B} \overbrace{\mathcal{C}(a, N(b))}^{\substack{\text{structural} \\ \text{information}}}, \tag{3}$$

where $w(a, b') = \frac{\mathcal{D}(a,b')}{\sum_{a \in A} \mathcal{D}(a,b')}$ with $\sum w(a, b') = 1$, $b' = \arg\min_{b \in B} \mathcal{C}(a, N(b))$, and $\mathcal{C}()$ is the χ^2 test statistic as defined in [4]. The result of matching between two points a and b, $\mathcal{C}(a, N(b))$, is weighted by their distance, $\mathcal{D}(a, b')$. The shape similarity measure in (3) with the maximum Hausdorff matching in (2) is defined to be a confidence level of matching for the STT:

$$r_1(A, B) = 1 - H(A, B). \tag{4}$$

4   Texture Classification

By selecting the features in the Trace transform which persist for an individual, even when their expression changes, we are identifying those scanning lines that are most important in the definition of the person. We refer to this method as the weighted Trace transform (WTT). Suppose that we have 3 training images which were transformed to the Trace transform space as described in section 2. We first compute the differences between the MTTs of the 3 images, $D_1 = |g_1 - g_2|$, $D_2 = |g_1 - g_3|$ and $D_3 = |g_2 - g_3|$, where g_i is the MTT for the ith training image. These image differences can be used to indicate the characteristics of the variations in appearance of images of the same face. We define the weight matrix as follows

$$W(\phi, p) = \begin{cases} 1, & \text{if } \sum_l \rho(D_l(\phi, p)) = 0, \\ 0, & \text{Otherwise.} \end{cases} \tag{5}$$

where $\rho(x) = 0$ if $x \leq \kappa$, otherwise $\rho(x) = 1$, and κ is some threshold. The values of these lines will constitute the "texture" features from the Trace transform. We therefore use the weight matrix to measure the similarities between two images as

$$r_2(T_r, T_t) = \frac{1}{\exp\left[\frac{1}{n_\kappa} \sum_{\phi, p} |T_r(\phi, p) - T_t(\phi, p)| \cdot W(\phi, p)\right]}, \tag{6}$$

where T_r is the MTT of one of the training images, all of which are used as reference images, T_t the MTT of the test image, and n_κ the total number of flagged lines in the WTT. This measure is defined to be the confidence level of matching for WTT.

5   Reinforcement Learning

We need to find the optimal parameters of υ and κ by using reinforcement learning (RL) [2]. In RL, the probability of y_i produces a 1 given W is

$$\Pr\{y_i = 1|W\} = p_i = f(s_i) = \frac{1}{1 + e^{-s_i}}. \tag{7}$$

Having generated output $\mathbf{y}(t)$ and received reinforcement $r(t)$, increment w_{ij} by

$$\Delta w_{ij}(t) = \alpha[(r(t) - \overline{r}(t-1))(y_i(t) - \overline{y}_i(t-1))]x_j - \delta w_{ij}(t), \tag{8}$$

Let us suppose that parameters υ and κ are represented by n Bernoulli units with m significant bits. The m most significant bits are forced to change by

$$y_i = \begin{cases} 1, & \text{if } p_i > 0.5, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

6   The Training Algorithm

Let us suppose that we have three training images. We pick one image for each person from the training set and thus create the reference set. The remaining two images for each person are defined as the tuning set. For STT, we use the default segmentation parameter in the reference set. For WTT, we compute the differences of the MTTs of the training set, in order to construct the weight matrix $W(\phi, p)$. Our approach is described by Algorithm 1, where N is the maximum number of iterations, τ_r a user-defined threshold, and τ_b the 'biased' RL threshold. The parameter values were chosen as follows: $\alpha = 0.9$, $\delta = 0.01$, $\gamma = 0.9$, $\tau_b = 0.85$, and $\tau_r = 0.92$, respectively.

Algorithm 1: The Closed-Loop Object Recognition Training Algorithm for STT and WTT

1. Initialise randomly weights w_{ij} ($w_{ij} \in [-1, 1]$)
2. Initialise matching confidence r^k to 0 (i.e. $r^k = 0$, $\forall k$)
3. For each image k in the tuning set do
   (a) Input tuning image x^k to the RL algorithm
   (b) Update parameters υ and κ for STT and WTT
      − *For shape Trace transform*
         i. Segment image x^k with current segmentation parameter υ, i.e. perform the following test:
         if $r^k \geq \tau_b$, obtain parameter υ for the most significant bits m by (9); update the outputs of the remaining $n - m$ units using the Bernoulli probability distribution in (7)
         Otherwise ($r^k < \tau_b$), acquire parameter υ by (7)

        ii. Obtain STT S_k by extracting the edge pixels of the segmented image k

      – *For weighted Trace transform*

        i. Compute weight matrix with current parameter κ, i.e. perform the following test:

           if $r^k \geq \tau_{\mathsf{b}}$, obtain weight matrix parameter κ for the most significant bits m by (9); update the outputs of the remaining $n - m$ units using the Bernoulli probability distribution in (7)

           Otherwise ($r^k < \tau_{\mathsf{b}}$), acquire weight matrix parameter κ by (7)

  (c) Perform matching measures for STT and WTT

      – *For shape Trace transform*

        i. Calculate shape matching using Hausdorff context for S_k against reference set by (4)

      – *For weighted Trace transform*

        i. Compute template matching using distance measure for M_k against reference set by (6)

  (d) Update each weight w_{ij} using r^k as reinforcement for the RL algorithm by (8)

4. Find the minimum matching confidence $\zeta = \min_k r^k$
5. Repeat step 3 until the number of iterations is equal to N or $\zeta \geq \tau_{\mathsf{r}}$

7   Classifier Combination

The results from two classifiers (STT and WTT) are combined by our nonuniform weighting function:

$$\mathcal{S}_{com}^{(k)}(x; \alpha^{(k)}) = \alpha^{(k)^T} \mathcal{S}^{(k)}(x), \tag{10}$$

where $\alpha^{(k)} = \{\alpha_1^{(k)}, \dots, \alpha_L^{(k)}\}^T \in \mathcal{R}^L$ is the nonuniform weighting factor for class ω_k and $\mathcal{S}^{(k)}(x) = \{\mathcal{S}_1^{(k)}(x), \dots, \mathcal{S}_L^{(k)}(x)\}^T \in \mathcal{R}^L$ is the set of soft decision labels produced by the L classifiers for class ω_k. T denotes the transposition operator. The nonuniform weighting factors $\alpha^{(k)}$ arise from the fact that different classifiers may be constructed from different assumptions and therefore each classifier may work best for a particular class. When the matching scores of STT and WTT have been calculated completely, they are combined by (10) where STT stands for classifier 1 and WTT for classifier 2, respectively.

8   Experiments and Results

In order to compare our algorithm with other approaches, the XM2VTS database was used here. We follow the standard evaluation protocol and performance measure as described in [3]. Figs. 2 and 3 show intermediate STTs and WTTs, respectively, obtained during training, when the parameters of the algorithm were being learned. Figs. 2(a) and 3(a) show the matching confidence values for STT and WTT respectively, obtained during training. Confidence is expected to increase as iterations progress. Fig. 2(b) shows the reference STT which was obtained with the default value of parameter υ, while Fig. 2(c) shows the MTT obtained from one of the images of the tuning set, which was used to search for
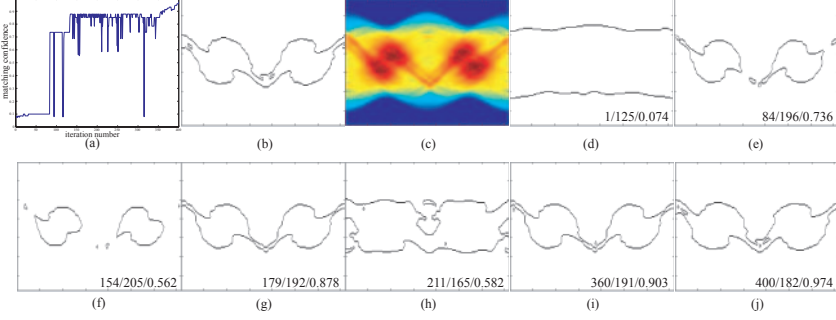
Fig. 2. Results of searching for the optimal segmentation parameter υ using RL. (a) Matching confidence (b) Reference STT where $\upsilon = 179$ (user defined parameter) (c) MTT produced from the tuning set. The results of thresholded versions of it are shown in order from (d) to (j) where labels in the bottom right corner indicate iteration, υ and matching confidence
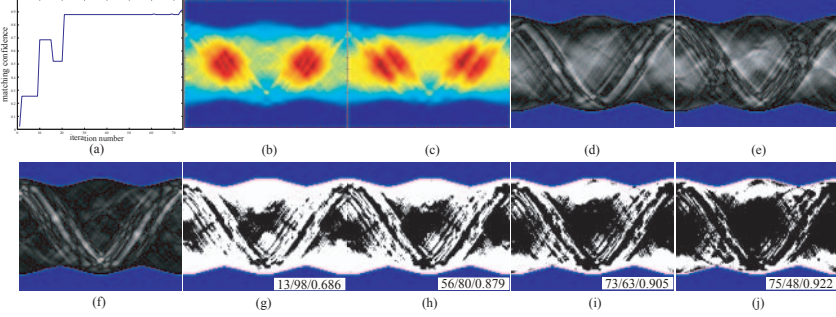
Fig. 3. Results of searching for the optimal thresholding parameter κ using RL. (a) Matching confidence (b) Reference MTT (c) MTT computed from one of the tuning images. The differences of the MTT for D_1, D_2 and D_3 are shown in (d), (e) and (f), respectively. WTTs obtained at each iteration step are shown in order from (g) to (j) where labels in the bottom right corner indicate iteration, κ and matching confidence

the optimal value of υ. Figs. 2(d) to (j) show itermediate STTs produced from MTT for the value of υ at the corresponding iteration step. The image in Fig. 2(j) is the result produced at the final iteration step. We can see how the similarity with the reference SST improves as the iterations progress. Figs. 3(b) and (c) show the MTTs produced from the reference and one of the tuning images. The 3 differences of the MTTs are shown in order from (d) to (f). Figs. 3(g) to (j) show the intermediate WTTs obtained during the training procedure when the thresholding parameter κ was being learned.

The error rates corresponding to the evaluation and test sets are summarized in Table 2. It should be noted that the results of STT and WTT are linearly combined with the optimal weighting factors $\alpha^{(1)} = \{0.2, 0.8\}$ and
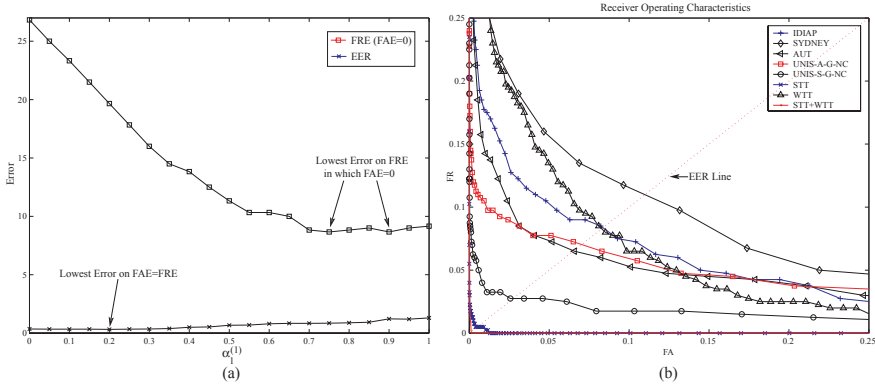
Fig. 4. Error rates obtained from the evaluation and test sets. (a) Combinations of STT and WTT on the evaluation set in which the lowest error rates on FAE=FRE and FRE with FAE=0 are obtained. (b) ROC curve computed on the test set

Table 2. Error rates according to protocol configuration I [3] (the results with * are from [3]). The weighting factors $\alpha^{(1)} = \{0.2, 0.8\}$ and $\alpha^{(2)} = \{0.85, 0.15\}$ were used in the calculation for the combinations of STT and WTT

Experiment	Evaluation set			Test set								
	FAE =FRE	FAE(FRE =0)	FRE(FAE= =0)	FAE=FRE		FRE=0		FAE=0		Total Error Rate (TER)		
				FA	FR	FA	FR	FA	FR	FAE=FRE	FRE=0	FAE=0
AUT*	8.1	48.4	19.0	8.2	6.0	46.6	0.8	0.5	20.0	*14.2*	*47.4*	*20.5*
IDIAP*	8.0	54.9	16.0	8.1	8.5	54.5	0.5	0.5	20.5	*16.6*	*55*	*21*
SYDNEY*	12.9	94.4	70.5	13.6	12.3	94.0	0.0	0.0	81.3	*25.9*	*94*	*81.3*
UniS-A-G-NC*	5.7	96.4	26.7	7.6	6.8	96.5	0.3	0.0	27.5	*14.4*	*96.8*	*27.5*
UniS-S-G-NC*	3.5	81.1	16.2	3.5	2.8	81.2	0.0	0.0	14.5	*6.3*	*81.2*	*14.5*
STT	1.12	2.62	9.16	0.97	0.5	3.3	0.0	0.0	6.5	1.47	3.3	6.5
WTT	9.55	70.59	88.83	6.5	10.25	72.39	0.0	0.0	87.5	16.75	72.39	87.5
STT+WTT	0.319	0.41	18.83	0.18	0.0	0.25	0.0	0.0	18.75	0.18	0.25	18.75

$\alpha^{(2)} = \{0.85, 0.15\}$, which is the case of lowest EER (Fig. 4 (a)). In this case, we achieved a $TER_{FAE=FRE}$ of 1.47% and 16.75% for STT and WTT, respectively. Further improvements, i.e. a $TER_{FAE=FRE}$ equal to 0.18%, were obtained when we combine the STT and WTT by a method of classifier combination with weighting factors $\alpha^{(k)}$ obtained from the evaluation set. From the inspection of the table, it can be seen that the proposed method, STT+WTT, is ranked as the first method with respect to TER. Fig. 4(b) shows a comparison between the ROC curves on the test set for the methods reported in [3] (score files have been made available in [7]) and the proposed method. It is seen that the area under the ROC curve for the proposed method is much smaller than the other ones.

9   Conclusions

We have introduced novel face feature representations using shape and texture characteristics derived from the MTT. Texture representation provides classifica-

tion ability for reducing the within-class variance, while shape representation offers discriminating features for increasing the between-class variances. A method for reducing the within-class variance by using the RL was proposed. A very low TER of 0.18% was obtained when we combine the STT together with WTT by means of classifier combination. From the experimental result, it is shown that the combined STT and WTT provide the best results followed in order by STT and WTT.

References

[1] A. Kadyrov and M. Petrou, *The Trace Transform and Its Applications*, IEEE Trans. PAMI, Vol. 23, No. 8, pp. 811-828, Aug. 2001.

[2] B. Bhanu and J. Peng, *Adaptive Integrated Image Segmentation and Object Recognition*, IEEE Trans. SMCC, Vol. 30, No. 4, pp. 427-441, Nov. 2000.

[3] J. Matas, M. Hamouz, K. Jonsson, J. Kittler, Y. Li, C. Kotropoulos, A. Tefas, I. Pitas, T. Tan, H. Yan, F. Smeraldi, J. Bigun, N. Capdevielle, W. Gerstner, S.B. Yacoub, Y. Abdeljaoued and E. Mayoraz, *Comparison of Face Verification Results on the XM2VTS Database*, In Proc. ICPR, pp. 858-863, 2000.

[4] S. Belongie, J. Malik and J. Puzicha, *Shape Matching and Object Recognition using Shape Context*, IEEE Trans. PAMI, Vol. 24, No. 24, pp. 509-522, Apr. 2002.

[5] S. Srisuk and W. Kurutach, *A New Robust Face Detection in Color Images*, in Proc. IEEE AFGR, pp. 306-311, May 2002.

[6] S. Srisuk, M. Petrou, W. Kurutach and A. Kadyrov, *Face Authentication using the Trace Transform*, in Proc. IEEE CVPR, pp. 305-312, June 2003.

[7] `http://www.ee.surrey.ac.uk/CVSSP/xm2vtsdb/results/face/verifica tion_LP/`, Surrey Univ. XM2VTS Face Authentication Contest, 2000.

Enhance the Alignment Accuracy of Active Shape Models Using Elastic Graph Matching

Sanqiang Zhao[1, 2], Wen Gao[2], Shiguang Shan[2], and Baocai Yin[1]

[1]Multimedia and Intelligent Software Technology Beijing Municipal Key Laboratory
Beijing University of Technology, Beijing, China, 100022
[2]ICT-ISVISION JDL for Face Recognition, Institute of Computing Technology
CAS, Beijing, China, 100080
{sqzhao,wgao,sgshan}@jdl.ac.cn, ybc@bjut.edu.cn

Abstract. Active Shape Model (ASM) is one of the most popular methods for image alignment. To improve its matching accuracy, in this paper, ASM searching method is combined with a simplified Elastic Bunch Graph Matching (EBGM) algorithm. Considering that EBGM is too time-consuming, landmarks are grouped into contour points and inner points, and inner points are further separated into several groups according to the distribution around salient features. For contour points, the original local derivative profile matching is exploited. While for every group of inner points, two pre-defined control points are searched by EBGM, and then used to adjust other points in the same group by using an affine transformation. Experimental results have shown that the proposed method greatly improves the alignment accuracy of ASM with only a little increase of time requirement since EBGM is only applied to a few control points.

1   Introduction

The accurate localization and alignment of facial feature points is of great importance for face recognition, animation and tracking, etc. Over the last two decades, various methods have been proposed to deal with this task. After Kass et al proposed Active Contour Models (ACMs) [1], or snakes in 1987, Cootes and Taylor's Active Shape Models (ASMs) [2] and later Active Appearance Models (AAMs) [3] have proven to be among the most powerful tools in this field.

   ASM and AAM are both based on statistical models. In ASM, the derivative profile, or local texture along the normal of the shape boundary, is exploited to model the local feature of each landmark point, and used to search each landmark position. Then the global shape models are applied to adjust the local search result based on the statistical shape model. Yet, as further research shows [4, 5, 7], due to its ambiguous local-texture searching strategy, ASM performs most successfully only when the object or structure class is fairly consistent in shape and intensity appearance. On the other hand, AAM [3] combines global statistical shape models with texture constraints to build appearance models and uses a linear prediction to obtain the appearance parameters for optimization; thus the shape can be calculated by minimizing the texture reconstruction error. To some extent, AAM may give a quite good match to image texture, but when the target image and background vary

significantly, it is still unable to locate feature landmarks accurately. Meanwhile, both the training process and the searching procedure of AAM are quite complex and slow.

Therefore, much improvement was put forward to advance ASM for face alignment. Ginneken et al [4] presented a non-linear gray-level appearance instead of the original derivative profile to model the local texture, and got a better matching result. Rogers et al [5] suggested a robust parameter estimation method using M-estimator and random sampling approaches to evaluate the shape parameters more reliably. However, ASM still depends heavily on the initialization and may easily be stuck in local minima.

Among other methods, Wiskott et al [6] constructed a stack-like structure, called Face Bunch Graph, and used it to search the whole image to find the pre-defined feature points. Through the iterating distortion of the graph, this Elastic Bunch Graph Matching (EBGM) algorithm can tolerate a certain degree of pose and expression changes and demonstrate a successful result. However, since this method is Gabor feature based, its time-consuming nodes searching process in the entire image region to large extent confines its further progress.

On account of all above, in our work, the landmarks are sorted into contour points and inner points, and the inner points are further separated into several groups according to the distribution around salient features in a face region. For the contour points, the original normal-based searching strategy is operated; for every group of inner points, two control points are chosen, and a simplified EBGM algorithm is utilized to adjust them. Fully understanding that the EBGM algorithm is quite a time-consuming process, we select the control points very carefully and representatively. When the new positions of the control points are identified, an affine transformation is applied to adjust other points in every group. This process is seamlessly combined with the ASM iterating searching algorithm and thus the accuracy can be greatly improved. The experimental results show that this method performs significantly better, with a speed not much slower than standard ASM algorithm.

The remaining part of this paper is organized as follows. In Section 2, the fundamentals of standard ASM are described. The simplified EBGM algorithm is presented in Section 3. After the detailed demonstration of our algorithm in Section 4, the experimental results are listed in Section 5, and the last section concludes the paper.

2   Standard ASM

ASM is definitely one of the best-suited approaches in the shape detecting task. The advantage of ASM is that it allows for considerable variability but still specific to the class of objects or structure they intend to represent. The following is a brief description of the standard ASM technique.

Given a set of annotated images, the manually labeled n landmark points in each image can be represented as a vector

$$\mathbf{x} = [x_0, y_0, x_1, y_1, \cdots, x_{n-1}, y_{n-1}]^T . \tag{1}$$

After aligning these vectors into a common coordinate, principal component analysis is applied to get a set of orthogonal basis $\mathbf{P}$. Every aligned shape can then be approximately represented as

$$\mathbf{x} \approx \overline{\mathbf{x}} + \mathbf{Pb} \ , \tag{2}$$

where $\mathbf{b}$ is the shape parameters vector. Meanwhile, in the standard ASM, Cootes et al [2] used a normalized derivative profile to build the local texture models for each landmark. The similarity between a normalized derivative search profile $\mathbf{g}_s$ and the model profile $\overline{\mathbf{g}}$ can be measured by its Mahalanobis distance, which is a weighted square error function

$$f(\mathbf{g}_s) = (\mathbf{g}_s - \overline{\mathbf{g}})^T \mathbf{C}_g^{-1}(\mathbf{g}_s - \overline{\mathbf{g}}), \tag{3}$$

where $\mathbf{C}_g$ is the covariance matrix of the normalized derivative profile.

Based on the Point Distribution Model (PDM) and the gray level model, the search progress can then be operated. After initialization, each landmark in the model is optimized by selecting the point with a minimum distance mentioned above in the direction perpendicular to the contour within a certain range. As the result of new shape is possibly implausible, PDM is used to adjust the shape parameters. Such procedure is repeated until no considerable change is observed.

3   A Simplified EBGM Algorithm

As [6] presented, the standard EBGM algorithm uses a coarse-to-fine procedure to find the landmarks and thus to extract from the image a graph that can maximize the similarity with the Face Bunch Graph (FBG), which is a stack-like structure that serves as a representation of both jets and edges information. However, this procedure is quite a slow process, because the final optimal image graph is obtained by distorting FBG in various sizes and positions over the entire image region, and that actually needs every pre-defined feature point to search in a large range.

Fortunately, in most practical applications, ASM can provide a quite close approximation to the initial landmark positions, which makes the rough searching stage in the EBGM unnecessary. Besides, the PDM in ASM can successfully confine the landmark locations to a class of acceptable geometrical structures, so edge constraints in EBGM is not needed to participate in the similarity measure for our algorithm. In this sense, based on [6], a simplified EBGM algorithm is presented as follows:

1) Initialize the rough position of the landmarks by ASM.

2) For a landmark $\vec{x}$ that needs to be accurately positioned, calculate its jet J.

3) From the bunch corresponding to that landmark, select the ith jet as a model jet J_{mi}, and calculate an estimated displacement $\vec{d}_i$ between jets J and J_{mi}, with all five frequency levels used.

4) Compute the similarity $S_{\phi i}$ between J and J_{mi} with the displacement $\vec{d}_i$ obtained from the former step. And then find another jet in the bunch and go to step 3) to repeat the above procedures until all the jets in the bunch have been operated.

5) Find the highest $S_{\phi i}$ value as the best one: $i^* = \arg\max_i \{S_{\phi i}\}$. And the corresponding $\vec{d}_{i^*}$ is used as the optimal displacement for the landmark $\vec{x}$.

6) Change to the next landmark to be adjusted and go to step 2) for another searching process.

One point to be noticed is that, in step 4), we do not take a jet J' from position $\vec{x}' = \vec{x} + \vec{d}_i$; instead, we approximate the similarity between J' and J_{mi} by computing the similarity between J and J_{mi} with the displacement $\vec{d}_i$. This is because we can estimate the similarity as if J' was extracted from a displacement $\vec{d}_i$ from its current location [8]. As a whole, jet computation for a landmark searching process is only once in this simplified EBGM algorithm, which will reduce computational effort greatly.

4   ASM Combined with EBGM Searching Strategy

The proper initialization of the mean shape is critical to ASM searching progress in the sense that a good initialization would be less possible to lead to incorrect local minima. The standard ASM partially solves this problem by multi-resolution strategy at the cost of more expenditure spent during training in every resolution level.

In most current face detection systems, two irises, thanks to their natural distinct feature, can be located easily and quickly, though sometimes not very accurately. And it has been proved that two irises can provide enough information to initialize the translation, the scale and the rotation angle for the mean shape model [7].

Experiments show irises' slight inaccuracy will induce ASM's significant initialization error. Therefore, we use the simplified EBGM algorithm in Section 3 to refine the face detector's result. Our refinement results also reveal the accurate searching performance of this algorithm.

4.1 Feature-Based Landmark Grouping and Alignment

Unfortunately, though our EBGM algorithm is simplified, it is still quite slower than ASM's normal-based searching method; moreover, experiments show that EBGM algorithm does not always perform better than the normal-based searching method in some outlying points along the image border [9].

Under this condition, we sort the face landmarks (Fig. 1a) into contour points and inner points; the inner points are further separated into several groups based on the distribution around salient features in a face region. Fig. 1(b) shows an example of six groups of inner points corresponding to two brow areas, two eye areas, one nose area and one mouth area. For the contour points, the original normal-based searching strategy is operated; as to every group of inner points, we select two control points, which are used to control all the other inner points in the same group. Since these two control points are very important, the simplified EBGM algorithm is utilized to refine

them. To speed the searching performance, we define only 11 representative control points (Fig. 1b) totally in all six groups. As an exception, one of the control points in the nose area is omitted, for it can be estimated from other control points in the two eye areas.

After the new positions of control points are identified by the simplified EBGM algorithm, we apply an affine transformation to adjust other points in every corresponding inner group. Actually this is a problem of alignment between two sets of feature points. We define this alignment as the rotation, scaling and translation which minimizes the sum of squared distances between pairs of corresponding points. Since we have two corresponding sets of points (two control points before and after refinement), we can compute the corresponding affine transformation parameters, and then the remaining inner points' positions in the same group will be easily calculated. Fig. 2. gives an instance of refinement to landmarks in the mouth area using our algorithm.

(a)   **(b)**
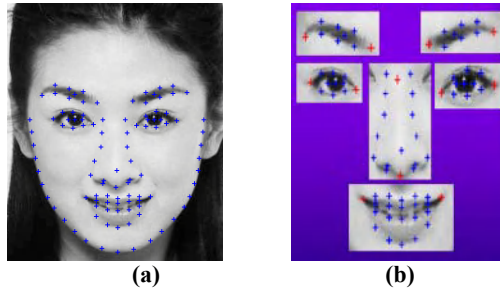
Fig. 1. Face landmarks grouping. (a) A manually labeled face image with 103 landmarks. (b) 78 inner points separated into six groups; red crosses representing control points

(a)   **(b)**
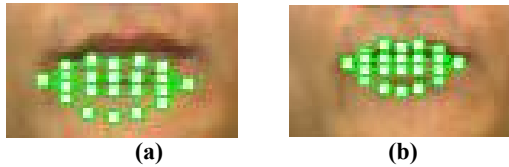
Fig. 2. Landmark alignment. (a) Initial landmarks in the mouth area. (b) Result of EBGM adjustment to (a)

4.2 ASM Combined with EBGM Searching Strategy

In standard ASM matching process, every point along the normal of the shape contour is searched to find a best one, and the accuracy of every searching step is very important to the final result. Based on the above landmark grouping and alignment, our algorithm reaches a compromise between computational effort and searching accuracy.

The entire searching procedure of our algorithm is listed as follows:

1) Initialize the starting shape by the model shape and two iris locations, which are identified by a face detector and refined by the simplified EBGM algorithm.
2) For the contour points, use the standard normal-based searching method to find the required movements.
3) For every group of inner points, the simplified EBGM algorithm is operated for searching the two control points. After that, the method in Section 4.2 is used to adjust other points in the same group. We then get a new shape.
4) Calculate additional pose and shape parameter changes required to move the model shape as close as possible to the new shape.
5) Update the shape and pose parameters by the above changes, and act on the model shape. Till now, we finish an iteration step. Go to step 2) for another matching circle until no considerable change is observed.

5   Experimental Results

Our experiment is based on a 500 manually labeled faces database, most of which are near frontal. We have normalized 350 of them for training the PDM and others for testing; we also selected 70 representative face images for training the EBGM bunches. To evaluate the performance of our algorithm, the average Euclidean distance error is calculated by the following equation:

$$E = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{1}{n} \sum_{j=1}^{n} \sqrt{(x_{ij} - x'_{ij})^2 + (y_{ij} - y'_{ij})^2} \right),$$
(4)

where N is the total number of test images, and n is the number of the landmarks in one face image. (x_{ij}, y_{ij}) is the jth manually labeled landmark location of the ith test image; and (x'_{ij}, y'_{ij}) is the corresponding j^{th} landmark location we calculated.

We also calculate the overall improvement percentage of our algorithm to the standard ASM by:

$$I = \frac{E_{ASM} - E_{ASM-EBGM}}{E_{ASM}} \times 100 \% \cdot$$
(5)

Table 1 lists our final experimental results based on 150 test images. The average error for standard ASM is about 3.08 pixels. After we apply EBGM to refine two iris locations, the error is reduced to 2.54 pixels. The overall improvement of our algorithm is about 39.9% compared with the standard ASM. Through comparison, we can see our algorithm improves the searching accuracy significantly, especially that of inner points.

Table 1. Performance of our algorithm

Method	Average Error	Improvement
ASM	3.08	
ASM with iris-refinement	2.54	17.5%
ASM combined with EBGM	1.85	39.9%

6   Conclusions and Acknowledgements

In this research, we combine the standard ASM algorithm with a simplified EBGM to improve the alignment accuracy. Because EBGM has the advantage of Gabor wavelet representation, the landmarks can be more accurately localized compared with using gray level representation. In order to reach a compromise between the searching efficiency and the alignment accuracy, we classify all the landmarks into contour points and inner points, and further separate the inner points into several groups according to their distribution around salient features in a face region. We apply the simplified EBGM algorithm to accurately localize the two control points of every group, and then, an affine transformation is used to adjust other points in the corresponding inner group. Experimental results show that our algorithm performs much better than the standard ASM.

References

[1]   M. Kass, A. Witkin and D. Terzopoulos, "Active Contour Models", 1st International Conference on Computer Vision, London, June, 1987, pp. 259-268.

[2]   T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham, "Active Shape Models - Their Training and Application", Computer Vision and Image Understanding, 61(1), 1995, pp. 38-59.

[3]   T.F. Cootes, G.J. Edwards, and C.J. Taylor, "Active Appearance Models," Proceeding of the 5th European Conference on Computer Vision, vol. 2, 1998, pp. 484-498.

[4]   B.V. Ginneken, A.F. Frangi et al, "A Non-linear Gray-level Appearance Model Improves Active Shape Model Segmentation", IEEE Workshop on Mathematical Models in Biomedical Image Analysis, MMBIA 2001, pp. 205-212.

[5]   M. Rogers and J. Graham, "Robust Active Shape Model Search", Proceedings of the European Conference on Computer Vision. May, 2002.

[6]   L. Wiskott, J.M. Fellous, N. Kruger et al, "Face Recognition by Elastic Graph Matching", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, July, 1997.

[7]   W. Wang, S. Shan, W. Gao and B. Cao. "An Improved Active Shape Model For Face Alignment", The 4th International Conference on Multi-modal Interface, IEEE ICMI 2002, Pittsburgh, USA, pp. 523-528, Oct., 2002.

[8]   D.S. Bolme, "Elastic Bunch Graph Matching", Masters Thesis, CSU Computer Science Department, June, 2003.

[9]   B. Zhang, W. Gao, S. Shan and W. Wang. "Constraint Shape Model Using Edge Constraint And Gabor Wavelet Based Search", 4th International Conference on Audio and Video Based Biometric Person Authentication, AVBPA 2003.

Reconstruction of High-Resolution Facial Image Using Recursive Error Back-Projection

Jeong-Seon Park[1,2] and Seong-Whan Lee[2]

[1] WatchVision, Inc.
Dongseon-dong 3ga, Seongbuk-gu, Seoul 136-053, Korea
jspark@watchvision.com
[2] Center for Artificial Vision Research, Korea University
Anam-dong, Seongbuk-ku, Seoul 136-701, Korea
{jspark, swlee}@image.korea.ac.kr

Abstract. This paper proposes a new method of reconstructing high-resolution facial image from a low-resolution facial image using a recursive error back-projection of example-based learning. A face is represented by a linear combination of prototypes of shape and texture. With the shape and texture information about the pixels in a given low-resolution facial image, we can estimate optimal coefficients for a linear combination of prototypes of shape and those of texture by solving least square minimization. Then high-resolution facial image can be reconstructed by using the optimal coefficients for linear combination of the high-resolution prototypes. Moreover recursive error back-projection is applied to improve the accuracy of high-resolution reconstruction. An error back-projection is composed of estimation, simulation, and error compensation.

The encouraging results of the proposed method show that our method can be used to improve the performance of the face recognition by applying our method to enhance the low-resolution facial images captured at visual surveillance systems.

1 Introduction

Handling low-resolution images is one of the most difficult and commonly occurring problems in various image processing applications such as analysis of scientific, medical, astronomical, and weather images, archiving, retrieval and transmission of those images as well as video surveillance or monitoring[1]. Numerous methods have been reported in the area of estimating or reconstructing high-resolution images from a series of low-resolution images or single low-resolution image. Super-resolution is a typical example of techniques reconstructing a high-resolution image from a series of low-resolution images[2]-[4], whereas interpolation produces a large image from only one low-resolution image.

We are concerned with building a high-resolution facial image from a low-resolution facial image for visual surveillance. Our task is distinguished from previous works that built high-resolution images mainly from scientific images

or image sequence of video data. The example based approach to interpreting images of variable objects are now attracting considerable interest among many researchers[5][6], due to its potential power of deriving high-level knowledge from a set of prototypical components.

This paper proposes a new method of reconstructing high-resolution facial image from a low-resolution facial image using a recursive error back-projection of example based learning. The extended 2D morphable face model[6] is used in example based learning, and a mathematical procedure for solving least square minimization(LSM) is applied to the model. More over, a recursive error back-projection procedure is applied to improve the performance of the reconstruction of the high-resolution under the extended 2D morphable face model. A error back-projection is composed of estimation, simulation, and error compensation.

2   Proposed Recursive Error Back-Projection Procedure

In order to improve the accuracy of the initial estimation, the basic concept of error back-projection has been used to various applications such as super-resolution[4]. In this section, we explained the procedure of the our recursive error back-projection for reconstructing high-resolution facial images.

Before explaining our proposed recursive error back-projection, we first define some notations as following Table 1.

The flowchart of the procedure we have designed for recursive error back-projection is outlined in Figure 1. An error back-projection is composed of three works: estimation of initial high-resolution data from an input low-resolution data, simulation of low-resolution data from the estimated high-resolution data, and error compensation which adjust the estimated high-resolution according to the reconstruction error.

First, as an estimation procedure, we estimate the initial high-resolution data(H_0^R) from input low-resolution one(L^I) by using our solution of least square minimization described in our previous works[6].

Second, as a simulation procedure, in order to verify the accuracy of our method, we simulate the low-resolution data(L_1^R) from the estimated high-

Table 1. Notations for recursive error back-projection

Notation	Definition
L^I	Input low-resolution data
t	Iteration index, $t = 1, 2, \cdots, T$
H_t^R	Reconstructed high-resolution data at iteration t
L_t^R	Low-resolution data simulated by down-sampling reconstructed one at iteration t
D_t^L	Reconstruction error measured by Euclidean distance between input and simulated low-resolution data at iteration t
T_1	Threshold value to determine whether the reconstruction is accurate or not
T_2	Threshold value to determine whether the iteration is convergent or not
L_t^E	Evaluated low-resolution error data by pixel-wise difference between input and simulated low-resolution data at iteration t
H_t^E	Reconstructed high-resolution error of low-resolution error at iteration t
ω_t	Weight for error compensation at iteration t
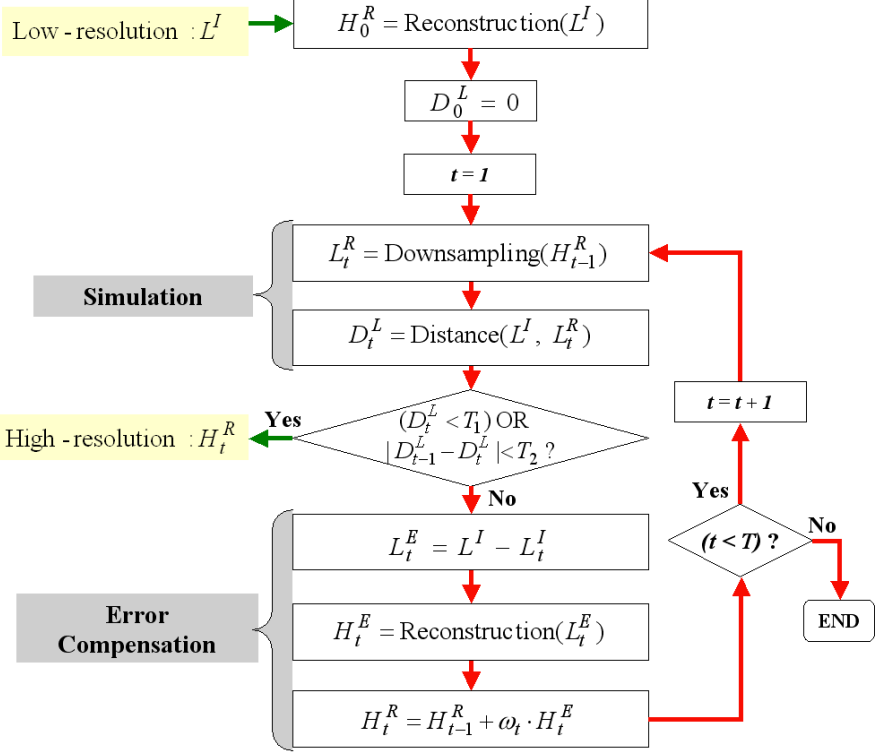
Fig. 1. Flowchart of the recursive error back-projection procedure

resolution one by down-sampling it, then measure the reconstruction error(D_1^L) between input low-resolution data and simulated one by simple Euclidean measure. We assume that if reconstruction is successful, the reconstruction error(or distance) will be very small. From this assumption, we determine whether the reconstruction is accurate or not by comparing current reconstruction error and one threshold(T_1) and whether the iteration is convergent or not by comparing amount of previous and current distance and another threshold value(T_2). If one or two of both comparisons are satisfied, then current result of reconstruction is considered as output high-resolution data, otherwise error back-projection procedure is applied.

Third, as an error compensation procedure, we create low-resolution error data between input low-resolution and simulated one by simple difference operation, estimate high-resolution error data by our reconstruction of low-resolution error data, then compensate initially estimated high-resolution data by adding currently estimated error to it, in order to improve it. In this procedure, we use weight value(ω_t) which is smaller than 1 in order to prevent divergence of iterations. The weight can be varied according the the current reconstruction distance(error), that is, the larger distance the larger weight.

We iteratively perform the same procedure until an accurate estimation is achieved, iterations are convergent, or maximum number of iteration(T) is performed.

3   Reconstruction Procedure of Example Based Learning

Suppose that sufficiently large amount of facial images are available for off-line training, then we can represent any input face by a linear combination of a number of facial prototypes. Moreover, if we have a pair of low-resolution facial image and its corresponding high-resolution image for the same person, we can obtain an approximation to the deformation required for the given low-resolution facial image by using the coefficients of examples. Then we can obtain high-resolution facial image by applying the estimated coefficients to the corresponding high-resolution example faces. Our goal is to find an optimal parameter set $\boldsymbol{\alpha}$ which best estimates the high-resolution image from a given low-resolution image, and to enhance the estimated result by applying recursive error back-projection explained previous section.

The proposed method is based on the morphable face model introduced by Poggio et al. and developed further by Vetter et al.[7][8]. Assuming that the pixelwise correspondence between facial images has already been established, the 2D shape of a face is coded as the displacement field from a reference image. So the shape of a facial image is represented by a vector $S = (d_1^x, d_1^y, \cdots, d_N^x, d_N^y)^T \epsilon \Re^{2N}$, where N is the number of pixels in image, (d_k^x, d_k^y) the x, y displacement of a point that corresponds to a point x_k in the reference face and can be denoted by $S(x_k)$. The texture is coded as the intensity map of the image which results from mapping the face onto the reference face. Thus, the shape normalized texture is represented as a vector $T = (i_1, \cdots, i_N)^T \epsilon \Re^N$, where i_k is the intensity or color of a point that corresponds to a point x_k among N pixels in the reference face and can be denoted by $T(x_k)$.

Next, we transform the orthogonal coordinate system by principal component analysis(PCA) into a system defined by eigenvectors s_p and t_p of the covariance matrices C_S and C_T on the set of M faces. Where $\bar{S}$ and $\bar{T}$ represent the mean of shape and that of texture, respectively. Then, a facial image can be represented by

$$S = \bar{S} + \sum_{p=1}^{M-1} \alpha_p s_p, \quad T = \bar{T} + \sum_{p=1}^{M-1} \beta_p t_p \tag{1}$$

where $\boldsymbol{\alpha}, \boldsymbol{\beta} \epsilon \Re^{M-1}$.

Before explaining the proposed synthesis procedure, we define two types of warping processes, forward and backward warping. Forward warping warps a texture expressed in reference shape onto each input face by using its shape information. This process results in an original facial image. Backward warping warps an input face onto the reference face by using its shape information. This process results in a texture information expressed in reference shape.

Table 2. Procedure of the proposed reconstruction

Step 1.	Obtain the texture of a low-resolution facial image by backward warping.
Step 2.	(a) Estimate a high-resolution shape from the given low-resolution shape.
	(b) Improve the high-resolution shape by recursive error back-projection.
Step 3.	(a) Estimate a high-resolution texture from the obtained low-resolution texture obtained at Step 1.
	(b) Improve the high-resolution texture by recursive error back-projection.
Step 4.	Synthesize a high-resolution facial image by forward warping the estimated texture with the estimated shape.

The resolution enhancement procedure consists of 4 steps, starting from a low-resolution facial image to a high-resolution face. Here the displacement of the pixels in an input low-resolution face which correspond to those in the reference face is known.

Step 1 and Step 4 are explained from the previous studies of morphable face models in many studies[5]. Step 2(a) and Step 3(b) are carried out by similar mathematical procedure except that the shape about a pixel is 2D vector and the texture is 1D(or 3D for RGB color image) vector. The detailed mathematical procedure of Step 2(a) of estimating high-resolution shape information from low-resolution one is available at the report of our previous works[6].

4   Experimental Results and Analysis

4.1   Face Database

For testing the proposed method, we used about 200 images of Caucasian faces that were rendered from a database of 3D head models recorded by a laser scanner[7]. The original images were color image set size 256×256 pixels. The pixel-wise correspondence between a reference facial image and every image in the database is given by the previous works of Vetter's group[7].

The original images were converted to 8-bit gray level and resized to 16×16 and 32×32 for low-resolution facial images by Bicubic interpolation technique. PCA was applied to a random subset of 100 facial images for constructing bases of the defined face model. The other 100 images were used for testing our algorithm.

4.2   Results of High-Resolution Facial Reconstruction

As mentioned before, 2D-shape and texture of facial images are treated separately. Therefore, a facial image is synthesized by combining both of the estimated shape and the estimated texture.

Figure 2 shows the examples of the high-resolution facial image synthesized from two kinds of low-resolution images of 16×16 and 32×32. Figure 2(a) shows the input low-resolution images, Figure 2 (b) to (e) the reconstructed high-resolution images using Bilinear interpolation, Bicubic interpolation, previous method(using only example based reconstruction) and proposed

(a) Input

(b) Bilinear

(c) Bicubic

(d) Previous

(e) Proposed

(f) Original

Reconstruction from 16×16 *Reconstruction from* 32×32
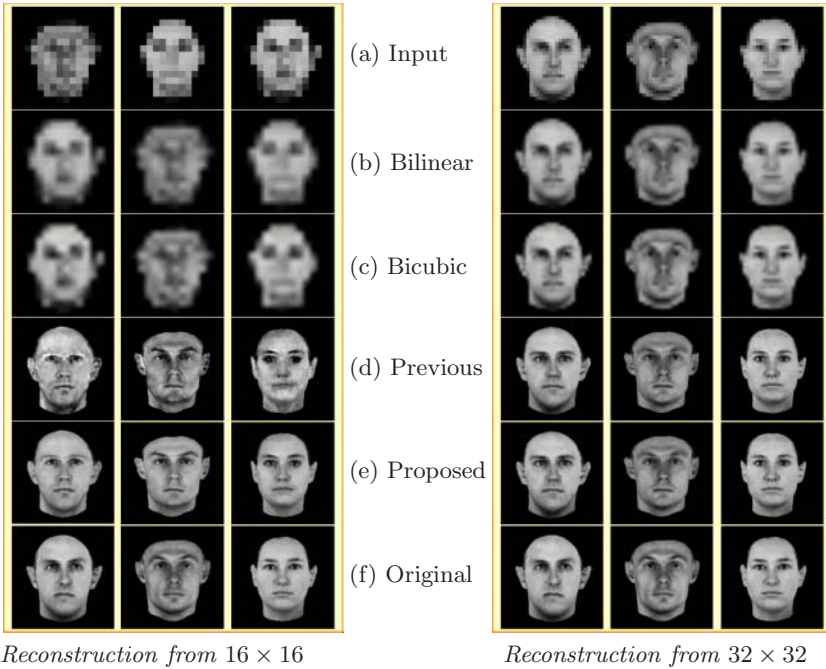
Fig. 2. Examples of high-resolution reconstructed from low-resolution facial image

method(enhanced (d) by our recursive error-back projection), respectively. Figure 2(f) is the original high-resolution facial images.

As shown in Figure 2, the reconstructed images by the proposed example-based method are more similar to the original ones and clearer than others. Better effect of reconstructing high-resolution data by the proposed method can be found in the reconstruction results of 16×16 facial image.

Figure 3 shows the mean reconstruction errors in shape, texture and facial image from the original high-resolution image. Horizontal axes of Figure 3 (a) and (b) represent the input low-resolution, two interpolation methods, the our previous method and proposed recursive error back-projection method. Vertical axes of them represent the mean displacement error per pixel about shape vectors and the mean intensity error per pixel(for an image using 256 gray level) about texture and image vector, respectively. Err_S_x and Err_S_y in Figure 3(a) are the x-directional mean displacement errors along the $x-$ and $y-$ axes for shape, respectively. And Err_T and Err_I in Figure 3(b) implies the mean intensity error for texture and for image, respectively.

5   Conclusions

In this paper, we proposed an efficient method of reconstructing high-resolution facial image using an recursive error back-projection of example-based learning
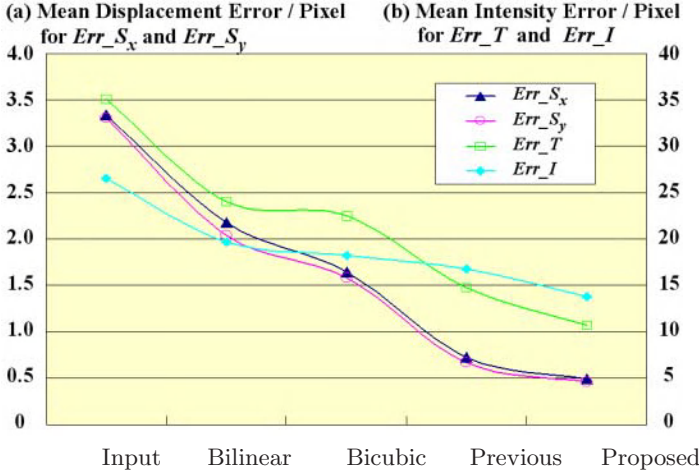
Fig. 3. Mean reconstruction errors

under extended 2D morphable face model. The proposed method consists of the following steps : estimating the initial high-resolution data from an input low-resolution data, simulating the low-resolution data from the initially estimated high-resolution data, and compensating error data to according to the initial reconstruction error, respectively.

The encouraging results of the proposed method show that our method can be used to improve the performance of the face recognition by applying our method to enhance the low-resolution facial images captured at visual surveillance systems.

Acknowledgements

References

[1] B. Tom and A. K. Katsaggelos, "Resolution Enhancement of Monochrome and Color Video Using Motion Comensation," *IEEE Trans. on Image Processing*, Vol. 10, No. 2, pp.278-287, Feb. 2001.   59
[2] S. Baker and T. Kanade, "Limit on Super-Resolution and How to Break Them," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 24 No. 9, pp.1167-1183, Sep. 2002.   59
[3] M. G. Kang and S. Chaudhuri, "Super-Resolution Image Reconstruction," *IEEE Signal Processing Magazine*, Vol. 23 No. 3, pp.19-36, May 2003.

[4] F. Dekeyser, P. Perez and P. Bouthemy, "Restoration of Noisy, Blurred, Under-sampled Image Sequences Using Parametric Motion Model," *Proc. of the ISIVC 2000*, Rabat, Morocco, April 2000. 59, 60

[5] M. J. Jones, P. Sinha, T. Vetter and T. Poggio, "Top-down learning of low-level vision tasks[brief communication]," *Current Biology*, Vol. 7 pp.991-994, 1997. 60, 63

[6] J.-S. Park and S.-W. Lee, "Resolution Enhancement of Facial Image Based on Top-down Learning," *Proc. of ACM SIGMM 2003 Workshop on Video Surveillance*, pp.59-64, Nov. 2003. 60, 63

[7] T. Vetter and N. E. Troje, "Separation of texture and shape in images of faces for image coding and synthesis," *Journal of the Optical Society of America A*. Vol. 14, No. 9, pp.2152-2161, 1997. 62, 63

[8] V. Blanz, S. Romdhani and T. Vetter, "Face Identification across Different Poses and Illuminations with a 3D Morphable Model," *Proc. of the 5th Int'l Conf. on Automatic Face and Gesture Recognition*, Washington, D. C. pp. 202-207, 2002. 62

Biometrics of Asymmetrical Face

Leonid Kompanets

Czestochowa University of Technology
Institute of Computer and Information Sciences
Center for Intelligent Multimedia Techniques
Dabrowski str. 73, 42-200 Czestochowa, Poland
`leonid.kompanets@icis.pcz.pl`

Abstract. Traditionally, in face authentication/identification methods the presumption concerning face/head symmetry is used. For novel applications that concern creating techniques by means of which it is possible to reproduce the extraordinary complexity of skin, muscle, eye and hair movements conveying emotion, gesture, psychological state or psycho-sociological traits, we begin to create new research direction called *Biometrics of Asymmetrical Face.* In this paper, a novel type of 2D precise normalized model of a face – called *Czestochowa-face* model - for modern and prospect authentication/identification techniques creation is presented. Also, the results of creating and pioneer researching the *ophthalmogeometrical technique* and the *facial asymmetry technique*, which are based on the Czestochowa-face model, are given. Some attention has been drawn to interdisciplinary research context. Beside the first-hand usage, the novel type of face model and the techniques may be employed in the areas of human-computer interaction, identification of cognition-psyche type of personality for personnel management, and some other future applications.

1 Background and Problem Statement

In the paper, the result of development of a novel model of a human face taking into consideration the ophthalmogeometry, facial asymmetry, and brain asymmetry phenomena are presented. The model and based on it the ophthalmogeometry and facial asymmetry techniques development are concerned with the algorithmic processing and interpretation of face and eyes part of facial image features as resources of traditional biometrics and/or specific cognition-psyche information.

This study is based on two theses: 1) A value of eye cornea diameter (the only known constant of human body after 4-5 years old) equals 10±0,56 mm [1]. So for purpose of a face image normalization it was introduced a unit called *Muld* (1*Muld* = (10±0,56 × 10±0,56) mm^2); for current scale of the image, 1 *Muld* evaluates as *?N* pixels. 2) Main source of facial asymmetry effect is a brain asymmetry phenomenon [2]. These complex information features of a special normalized facial images can be used as a live biometrics and/or as a specific cognition-psyche information [5-7].

Author's estimation of the effect from the asymmetric face model implementation indicates that 1) the error of absolute values of primary face features measurement can

be reduced by 5-7%, 2) "precise and subtle" primary facial information can be used in advanced applications (psychological testing, emotional or gesture simulation).

The idea of face asymmetry phenomenon usage came to author's mind while getting familiarized with [3] in New York in March 2002. It's first presentation with the slide session was realize by the author during the oral presentation [6]. The materials [4, 2, 1] were found in the Internet later on.

During the creation of the model and the techniques, and their approval the basis of images of about 150 people, the following problems were successfully solved:

1. Finding (discovering) a 'truthful" vertical axis Y of a face.
2. Creation of the *Muld*-normalization procedure of a facial image.
3. Constructing a combine (two polar and one Cartesian) facial coordinate system Y-$O1$-$X1$-$O2$-$X2$ based on the "truthful" axis Y, and a special virtual coordinate system Y^*-O^*-X^* for an eyes area.
4. Visualization in Y^*-O^*-X^* of person's ophthalmogeometrical pattern.
5. Synthesis of special facial images (left-left (LC) and right-right (RC) composites – Fig.20, 21) for face asymmetry features evaluation.
6. The Czestochowa-face creation (Fig.15, 16).
7. Creation of the ophthalmogeometrical authentication technique.
8. Creation of the face asymmetry authentication technique (an algorithm for precise mapping and evaluating (pseudo-information) similarity of a pair of compared facial component or given component set in holistic manner).
9. Formulation of some theses concerning the cognition-psyche interpretation of gained ophthalmogeometrical, face asymmetry and other features.

The lack of space allows only to focus on the graphical an algorithmic means of contents transmission of the enumerated problems.

2   On the Ophthalmogeometry and Brain Asymmetry Phenomena

According to Ernst Muldashev [1], person's psychological and other states are described with 22 parameters of an eyes part of a face. We use information that is included in specific planar ophthalmogeometrical figures, which are produced with tangents of four corners, appropriate eyes silhouettes, wrinkles over eyes, and some other facial components (Fig.1, 22-24). Face asymmetry phenomenon has been known for long and researched by painters, psychologist [3]. However, it is not widely used. At the moment, the original *instrumental* interpretation of face asymmetry is based on the technical and psychological formalization of the brain hemispheres asymmetry phenomenon, being dealt with Avtandil Anuashvili method [2].

The primary information of the enumerated phenomena is presented by means of a 2D single frontal facial image. Biometrics and/or cognition-psyche features may be drawn out of this information in a non-intrusive way. The main aim of the research is to verify the theses that the ophthalmogeometry and face asymmetry phenomena may be used 1) for construction of new valuable biometrics especially in case of their fusion, 2) for interpretation of "precise and subtle" primary facial information in cognition-psyche terms in advanced and future applications.
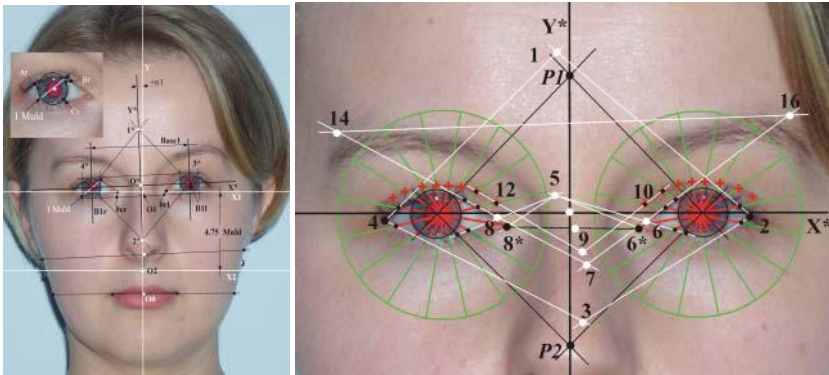
Fig.1 and **2.** Illustration of the preparatory and content-related stages of an ophthalmogeometrical pattern visualization

5 years old   10   25

(2 years old)   9   24
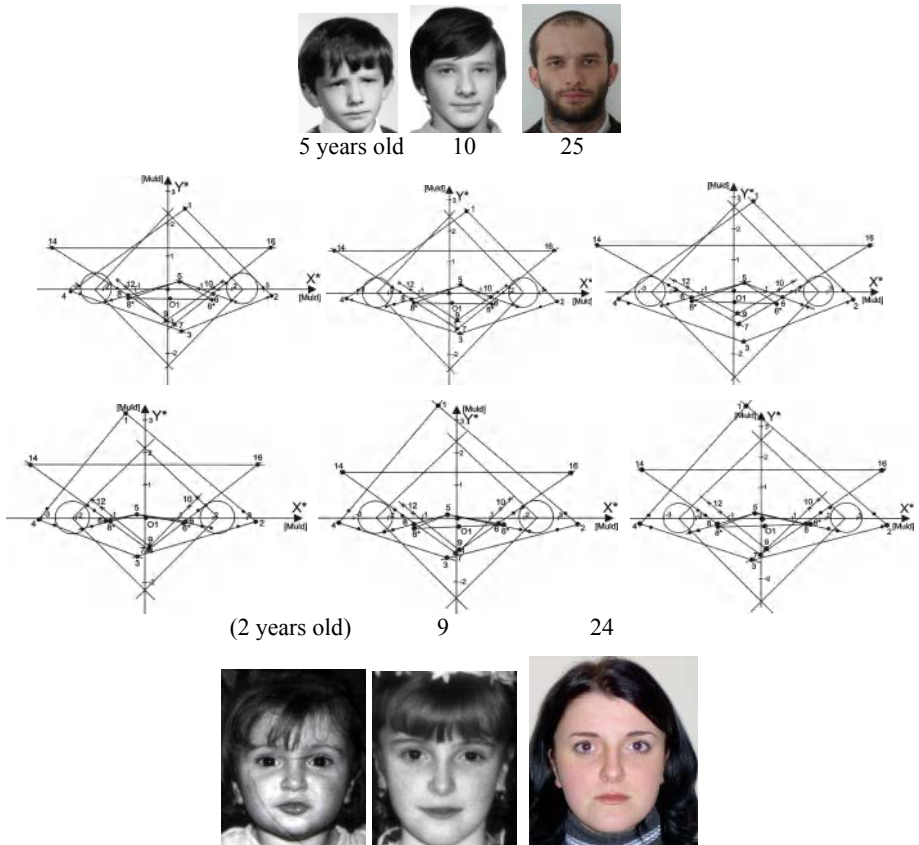
Fig.3-8 and **9-14.** Accordingly, the photos and visualized ophthalmogeometrical patterns of person 1 in different age (5, 10, 25 years old) and person 2 (2, 9, 24)

3   "Truthful" Vertical Axis, *Muld*-Normalization, Two Special Coordinate Systems of a Face

Human face is an example of a new class of information object – non-rigid, dynamic changing, intelligent one, mathematical and algorithmic description of which is an art and challenge for programmer. The idea of finding a "truthful" vertical axis Y of a face and constructing the special coordinate systems Y-$O1$-$X1$-$O2$-$X2$ and Y^*-O^*-X^* are illustrated in Fig.1 and 2.

   In the course of experiment for finding the axis Y, there were chosen anthropometrical points $O1$ (the center of a line linking interior eye corners Ocr, Ocl in Fig.1 (points 8^* and $6*$ in Fig.2) and $O0$ (the center of a line linking mouth corners in Fig.1). (In the case of the assignation of point $O0$ being impossible this way, there has been constructed reserve procedure.) For strict face components mapping and comparison (silhouettes, eyes, brown, mouth, nose, wrinkles, unique traits), the necessary quantity of rays coming from points $O1$ and/or $O2$ whole facial system and necessary quantity of parallels to Y and/or $X1$ lines are used (Fig.3, 4). All measurements here and below are done in *Muld*s.

4   Ophthalmogeometrical Pattern Visualization and Facial Composite Synthesis

As proven in [1, 5-7], person's pattern has a unique appearance from birth till human death (Fig.3-8 and 9-14). But it is not ordinary visible. Examples of the patterns are given in Fig.2, 3-17.

   To visualize the pattern, the sequence of calculating steps is as the following.
   INPUT: 2D single frontal view facial image.
   OUTPUT: Person's ophthalmogeometric pattern (Fig.1).

1. Find the placement of two circles described around the eyes cornea (iris) and measure $?N$ of pixels that constitutes 1 *Muld* (Fig.1).
2. Build the system Y^*-O^*-X^* and the figure including points P1*, P2* (Fig.2). If these points are placed on the line Y^*, then
3. Build the external figures 1,2,3,4 and internal – 5,6,7,8, the triangle 5,8,8* and 5,6,6*, 7,14,15,16, the angle 10,9,12 (Fig.2).
4. Find point $O1$. Construct the system Y-$O1$-$X1$-$O2$-$X2$.
5. Adjust the system Y^*-O^*-X^* and system Y-$O1$-$X1$-$O2$-$X2$ as in Fig.1.
6. Measure the ophthalmogeometrical pattern parameters (see Tab.2)
7. $[\pm XO^*, \pm YO^*, \pm\alpha\,[°]$, $Base1$; $X3, Y3, X1, Y1, Y15, X5, Y5, X7, Y7, X9, Y9]$.

   In Fig.18-21 and 22-24, the examples of facial composite synthesis are given. A composite is an input information for the facial asymmetry technique.

5   Constructing the Czestochowa-Faces

The main Czestochowa-face features are: 1) the "truthful" vertical axis Y of an asymmetrical face, 2) the special combined system Y-$O1$-$X1$-$O2$-$X2$ and adjusted to it the special virtual coordinate system Y^*-O^*-$X*$ for eyes area of a face, 3) the *Muld*-normalization procedure and construction of "Face Box", (Fig.15-16), 4) the ophthalmogeometrical pattern of a person, face asymmetry features, and brain asymmetry phenomenon background, 5) the procedure of evaluating of (pseudo-information) similarity for any pair of facial components or set of components, 6) algorithmic realization of the model, 7) measured features interpretation in biometrics, brain asymmetry, and cognition-psyche terms, 8) possibility of an exact facial image mapping in the uniformed absolute values (in *Muld*s).

Example of the *Czestochowa-faces*
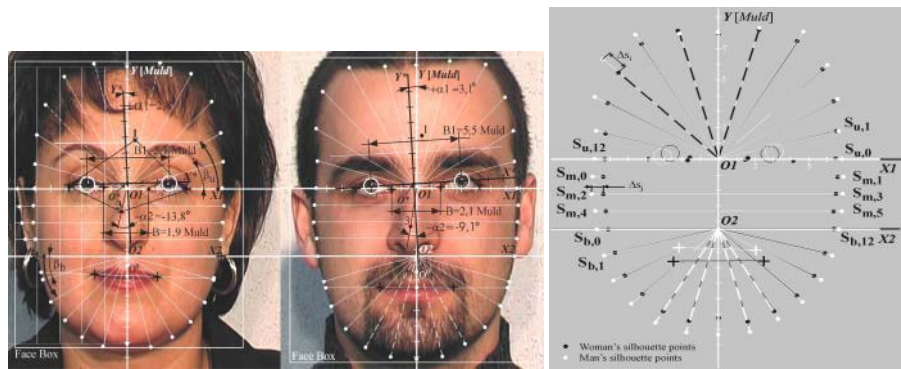
Fig. 15-17. Illustration of the *Muld*-normalization and mapping procedures. Woman's and man's facial images are mapped for *F-, L*-vector extraction and pseudo-information similarity evaluation of facial silhouettes (see Eq.(1)). *JeK* (*F, L*) = - 0,0927

Left-sided		Original	Right-sided
0,00143886423	Mouth		-0,00117342786
0,00138366621	Left Eye		-0,00012540007
0,00231125818	Right Eye		-0,00183951326
0,00531095134	Left Brow		-0,00411619702
0,00088645774	Right Brow		-0,00010896059
0,01133119770	TOTAL		-0,00736349880

Fig. 18-21 and **22-24.** Results of the pseudo-information similarity of the face pairs evaluation. (The values must be multiplied by 100, see Eq.(1))

6   Face Asymmetry and Ophthalmogeometry Techniques Experimenting

It was proved by the author [5] that the pseudo-information measure (1) is an effective tool for a holistic similarity evaluation and its advantages were presented.

The functional *JeK*[.] for the similarity measure is the following [5]:

$$\pm JeK^{(\Sigma)}\,[PDF(\sum_{z=1}^{Z}F_i^{(Z)});PDF(\sum_{z=1}^{Z}L_i^{(Z)})] = \frac{\sum\limits_{i,z}[f_i^{(z)}-l_i^{(z)}]\log_2[f_i^{(z)}/l_i^{(z)}]}{-\sum\limits_{i,z}f_i^{(z)}\log_2 f_i^{(z)}-\sum\limits_{i,z}l_i^{(z)}\log_2 l_i^{(z)}}100 \tag{1}$$

where: $F_i = f\{f_1,...,f_I\}$, $L_i = \{l_1,...,l_I\}$ − basic (*F*) and comparing (*L*) functions (for example, sample sequences, images, correlation or other functions); *PDF*s − *F* and *L* functions that answer requirements of *pdf* (a probability density function); *I, i=1,...,I* − grid, on which values of *F, L* are formed; *Z, z=1,...,Z* – a number of comparing objects components (for example, 5 components in Fig.22-24); ±*JeK* – the sign that identifies a situation: when a default basic function is *F* – (+), if on the contrary - (-).

In Fig.17, 18-21, 22-24 and Tab.1, the examples of similarity values for faces/composites are given. The facial asymmetry technique is sensitive tool.

Table 1. Estimators of the pseudo-information similarity for facial silhouettes

±*JeK(.)*[%]	*NI*-MI	*NI*-LC	*NI*-RC	LC-RC
Woman face (Fig.18-21)	0,0394	-0,0170	0,1525	0,2070
Man face (Fig.16)	0,0334	0,3578	0,1486	-0,1193

Table 2. Result of the ophthalmogeometric authentication technique experiment

Two different persons (accordingly, images 2 and 3)						
Input of image *2* →		???				Input of image *3* ←↓
Feature	**1**	**2**	**3**	**4**	**5↓**	**← 6**	**7**
XO* [*Muld*]	0	ε	*1*	+	0.1021	0.0933	0.1022
YO* [*Muld*]	0.3125	+	*0*	+	0.3502	0.3200	0.0377
±*α* [°]	0.9624	+	*0*	+	1.5231	1.3917	-0.5607
Base1 [*Muld*]	5.5313		*0*		5.5313	5.0540	0
X3 -"-	0.1941	+	*1*	–	-1.4995	-1.3701	*1.6936*
Y3 -"-	-1.7028	–	*0*	–	-1.1000	-1.0060	*-0.6019*
X1 -"-	0.4479	+	*0*	+	1.5597	1.4251	*-1.1118*
Y1 -"-	1.8303	+	*0*	+	-1.1417	-1.0432	*2.9720*
Y15 -"-	2.1659	+	*0*	+	2.0384	1.8625	*0.1275*
X5 -"-	-0.1628	–	*0*	–	-0.8037	-0.7343	*0.6409*
Y5 -"-	0.0454	ε	*1*	–	-0.1634	-0.1493	*0.2088*
X7 -"-	0.0599	+	*1*	–	-0.3044	-0.2781	*0.3643*
Y7 -"-	-1.3442	–	*0*	–	-1.2512	-1.1432	*-0.0930*
X9 -"-	-0.4701	ε	*0*	ε	0.0873	0.0797	*-0.5573*
Y9 -"-	-2.0035	–	*0*	–	-1.4767	-1.3489	*-0.5272*
					σ₁₂ = →	*1.029*

Legend: 1 and ← 6 – the ophthalmogeometrical pattern parameter vectors of a basic and comparing images; 5↓ - parameter vector of normalized pattern ←6; 2 and 4

– sign masks of the vectors 1 and 5↓; 3 – Hamming distance vector for the vectors 2 and 4.

The idea of using of the pattern for authentication, and also the result of experiment in working mode are presented in Tab.2. At the first stage, a sign vectors 2 and 4 produced by the pattern parameter of basic (1) and comparing (←6) images are compared. If necessary, the mean square distance σ_{12} of parameters $x3 - y9$ are calculated. If Hamming distance 3 is non-zero, further comparing is not need. It can be assume that the precision ε of parameter measuring is near to $\sigma_{12} = 0,16$ *Muld*.

The experiment result confirms high efficiency of the technique: 1) the pattern can be used as the novel live biometric, 2) achieved precision of the method is high ($\sigma_{12} \approx 0,16$ *Muld*), 3) expenditure of computational resources is minimal, 4) some pattern's points (for example, point 3, may be interpreted in the cognition-psyche terms.

7   Conclusion

C1. The novel type of 2D precise model of a face – the Czestochowa-face model - for authentication/identification techniques creation has been constructed.

C2. The ophthalmogeometrical technique and the facial asymmetry one, which are based on the Czestochowa-face model, have been created and researched.

C3. It has also been verified that these biometrics contain some person's cognitive and psychological information; it can make them be interesting in future applications.

References

[1]   Muldashev, E.: Whom did we descend from?, OLMA-PRESS, Moscow (2002) (In Russian)

[2]   Anuashvili, A.: Fundamentals of Psychology. Scientific, Philosophic and Spiritual Fundamentals of Psychology. The Institute for Control Problems Press, Moscow (2001) (In Russian).

[3]   Carter, R.: Mapping the Mind. California (1998)

[4]   Liu, Y., et. al.: Facial Asymmetry: A New Biometric. (2001). Available at `http://www.ri.cmu.edu/projects/project_437.html`

[5]   Kompanets, L., Valchuk, T.: Identification/Authentication of Person Cognitive Characteristics. The IEEE AutoID'02 Proc. 3rd Workshop on Automatic Identification Advanced Technologies, Tarrytown, New York, USA, 14-15 March (2002) 12-16

[6]   Kompanets, L., et. al.: Mental Characteristics of Person as Basic Biometrics. In: Tistarelli, M., Bigun, J., Jain, Anil K. (eds): Biometric Authentication. Lecture Notes in Computer Science, Vol. 2359. Springer-Verlag, Berlin Heidelberg New York (2002) 78-90

[7]   Kompanets, L., et. al.: Based on Pseudo-Information Evaluation, Facial Asymmetry and Ophthalmologic Geometry Techniques for Human-Computer Interaction, Person Psyche Type Identification, and Other Applications. Proc. 7th Intern. Multi-Conf. on Systemics, Cybernetics and Informatics: SCI'03, July 27-30, Orlando, Fl., USA, Vol. XII (2003) 235-240

Face Authentication from Cell Phone Camera Images with Illumination and Temporal Variations

Saim Qidwai, Krithika Venkataramani, and B. V. K. Vijaya Kumar

Department of Electrical and Computer Engineering, CyberSecurity Lab
Carnegie Mellon University, Pittsburgh, PA, 15213 USA
`saq@andrew.cmu.edu`
`{krithika,kumar}@ece.cmu.edu`
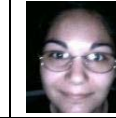
Abstract. We investigated the performance of 3 face verification algorithms (Correlation Filters, Individual PCA, FisherFaces) on an image database that was collected by a cell phone camera. Cell phone camera images tend to be of poorer quality and because of their portability, algorithms must deal with scale changes and dynamic outdoor illumination changes. While Individual PCA and FisherFaces focus on the image domain, Correlation Filters work in the frequency domain and offer advantages such as shift-invariance, ability to accommodate in-class image variability and closed form expressions. Results suggest that correlation filters offer better verification performance with this database.

1   Introduction

Many face recognition algorithms perform well on databases that had been collected with high-resolution cameras and in highly controlled situations. However, they may not retain good performance in real life situations where there is a lot of variation in illumination, scale, pose, etc. In applications such as face authentication using cameras in cell phones and PDAs, the cameras may introduce image distortions (e.g., because of fish-eye lens) and may be used in a wide range of illumination conditions, as well as variation in scale and pose. An important question is which of the face authentication algorithms will work well with face images produced by cell phone cameras? To address this issue, we collected a face database at Carnegie Mellon University using a cell phone camera. In this paper, we evaluate and compare the performance of correlation filters for face authentication with Individual PCA [1] and FisherFaces [2] under various lighting conditions. Correlation filters are attractive for a variety of reasons such as shift in-variance, ability to accommodate in-class image variability, ability to trade-off between discrimination and distortion tolerance, and the fact that they provide closed-form expressions [3-5].

The rest of the paper is organized as follows. Section 2 provides some background on correlation filters. Section 3 gives details on the database collection process using a cell phone camera and the pre-processing done on these images. Section 4 provides an evaluation of correlation filters using this database along with a comparison with Individual PCA and FisherFaces. Finally, conclusions are provided in Section 5.

Table 1. Sample images of the same person under different illuminations in Rounds 1 and 2

	Indoor	**Both lights**	**Left light**	**Right light**	**No light**	**Outdoor**
1						
2						

2   Background

2.1   Correlation Filters

The major difference between correlation filters and other methods is that the correlation filters are used in the spatial frequency domain, whereas other methods such as the Individual PCA and the FisherFaces work primarily in the spatial domain. While more details are available in [6], correlation filters are designed in the frequency domain using the Fourier Transforms (FTs) of the training images. The multiplication of the FT of the test image with the filter and the inverse FT of the resulting product gives the correlation output. Typically, for a well-designed filter, a sharp correlation output peak implies an authentic, whereas lack of such a distinct peak implies an impostor. Correlation filters have the advantage of built-in shift invariance, in that if the test image is shifted with respect to the training images, then the correlation peak will shift by the same amount from the origin. Typically the peak to side lobe ratio (PSR) is used as a measure of the sharpness of the peak of the correlation output [6] and is typically high for authentics and low for impostors.

Previously, correlation filters were mostly used for automatic target recognition (ATR) applications [3], and have been applied to biometrics [6,7] only recently. While there are many correlation filter designs [3-5], the Unconstrained Optimal Tradeoff Synthetic Discriminant Function (UOTSDF) filters [4,5] is used here in this paper because of its ability to provide high discrimination while providing noise tolerance, its simplicity in computation and especially for its ability to be easily updated incrementally [6]. The UOTSDF filter is given by the following equation.

$$\mathbf{h} = (\alpha \mathbf{C} + \beta \mathbf{D})^{-1} \mathbf{m} \tag{1}$$

where $\mathbf{C}$ is a dxd diagonal matrix containing the elements of the input noise power spectral density along its diagonal, $\mathbf{D}$ is a dxd diagonal matrix containing the average power spectrum of the training images placed along its diagonal, $\mathbf{m}$ is the mean FT of the training images, lexicographically reordered into a column vector and α and β are relative weights for noise tolerance and peak sharpness respectively.

By updating the mean FT $\mathbf{m}$ and the average power spectrum $\mathbf{D}$ of the training images, the UOTSDF filter can be updated easily [6]. This reduces the need to store all training images. This method can also be used to selectively choose the training set

such that the filter is updated only with images that are not already sufficiently well represented in the filter. The new training image is correlated with the current filter, and if its PSR is below a threshold τ_h (say 100), it is used to update the current filter. We can also go one step further to build multiple filters from the training set such that each filter represents a different distinct set of training images. If a new training image is sufficiently different from the current filter(s), i.e., the PSR of the correlation output of the new training image with current filter(s) is below a threshold τ_l (say 40), it is used to make a new filter [8].

2.2   Individual PCA

The Eigenface approach to face recognition, also known as Universal PCA, was introduced by Turk and Pentland [1]. Training images from all the classes are used to construct an eigenspace. Within this universal eigenspace, the face images of each of the classes form a cluster around a certain region. It was shown in [9] that face authentication performance can be improved by constructing eigenspaces from the face images of one person since it is tuned to that person. This method is referred to as the Individual PCA.

2.3   FisherFaces

The Fisher Linear Discriminant Analysis (LDA) [10] tries to find an optimal projection direction **w** so as to find the best discrimination between classes in a reduced dimensional space. In the case of face authentication, Fisher LDA projects the face images into a one-dimensional (1-D) subspace such that the between-class distance in the projected space is increased while the within-class scatter is reduced. The optimal projection direction **w** is given by

$$\mathbf{w} = \mathbf{S}_W^{-1}(\mathbf{m}_1 - \mathbf{m}_2) \tag{2}$$

where $\mathbf{S}_W$ is the within class scatter matrix and $\mathbf{m}_i$ is the ith class mean lexicographically ordered into a column vector. Usually, the within class scatter matrix S_W is singular, since it is of size dxd, d being the number of pixels in an image, and its rank is at most N-2, N being the total number of training images. Since the number of pixels d is much larger than the number of training images N, the training images are first projected onto a lower dimensional space (typically (N-2)), using Universal PCA, and then the discriminant analysis is done on the lower dimensional space [2].

3   Cell Phone Camera Face Database

3.1   Data Collection

The image database comprised of 20 subjects whose images were taken with a cell phone camera. The cell camera used was NTT DoCoMo's mova D25li part of the SH251i product line. It is equipped with a 170,000-pixel charge coupled device (CCD) camera. The images are stored in a "Memory Stick Duo". All the images were

collected in the camera's "Burst Mode", where 20 frames of size 120x120 pixels are captured over a period of time. The vertical and horizontal resolutions are 96 dots per inch and the bit depth of the images was 8.

Two rounds of images were collected with a period of approximately one month between the rounds to evaluate the performance of the methods over time as people's appearances change with time. For evaluating the performance of various algorithms under different illumination conditions, images were collected at six different illumination settings. The first setting was the normal ambient background indoor lighting setting. Additional illumination using a lamp to the left, right and both sides of the subject were the next three settings. The fifth setting mimicked total darkness with all artificial lights (including the ambient lights) being switched off and using only the camera's built-in compact light. The last setting is outdoors, where the illumination is very dynamic and undergoes a lot of variation depending on the time of the day and the location. The outdoor images were taken at different locations between the 2 rounds, and it was attempted to take the images at different times of the day as well so as to incorporate the fluctuations in outdoor lighting. In each round there were 2 sets, Set 1 and Set 2 respectively. Set 1 images were used primarily in the training of the algorithms whereas Set 2 images were used primarily in the testing. There are 20 images in each set, thus totaling 80*6=480 images per person for all illuminations.

Table 1 shows one image from each different variation and the different rounds. It is important to note that no strict constraints were put on the orientation and scale of the images in order to mimic real life situations. The fish-eye lens of the camera resulted in distorted images at different distances and orientations from the camera thus providing scale, tilt and pose variations in the images.

Table 2. Average EER (%) of Correlation filters, Individual PCA and FisherFaces for testing Processes 1, 2 and 3 at different illuminations

	Correlation Filters			Individual PCA			FisherFaces		
	I	II	III	I	II	III	I	II	III
Normal R1	1.5	-	-	0.2	-	-	0	-	-
Normal R2	7.1	-	0.4	14.8	-	10.3	3.8	-	0.0
Left R1	3.0	0.3	-	6.9	2.7	-	1.0	0.6	-
Left R2	14.5	5.2	1.2	14.2	10.0	1.1	5.5	4.3	0.1
Right R1	3.6	0.4	-	11.6	2.9	-	2.7	0.1	-
Right R2	17.3	0.3	2.7	20.6	17.1	3.3	6.0	3.2	1.8
Both R1	3.9	0.1	-	14.4	2.6	-	5.6	0.6	-
Both R2	16.5	7.2	0.1	21.5	13.0	0.5	5.0	3.3	0.0
No light R1	16.8	0.0	-	25.3	5.8	-	13.1	0.1	-
No light R2	19.2	4.1	1.2	29.2	16.6	5.1	13.1	7.0	3.1
Outdoor R1	9.9	0.8	-	18.9	5.6	-	7.6	1.3	-
Outdoor R2	23.3	16.2	0.0	36.2	32.0	3.4	18.2	16.0	0.3
Average	11.4	3.5	0.9	17.8	10.8	4.0	6.8	3.6	0.9

3.2   Processing

One of the major hurdles was pre-processing the images in the presence of scale, tilt and pose variations in order to make them suitable for testing. The distance between eyes was used for cropping the face regions. The eye locations were found semi-automatically due to the absence of accurate automatic eye location algorithms. The eye location in the first image of a set was found manually while the rest were found by correlating them with a UOTSDF filter built from the first image. Finally contrast stretching was done on all the images for some illumination normalization and then all the images were normalized to have unit energy.

4   Evaluation of Correlation Filters, Individual PCA and FisherFaces

We compared the verification results of the three methods - Correlation Filters, Individual PCA and FisherFaces, on the processed images. The testing was further divided into three processes to study the effect of time as well as of illumination and how much the performance improves by adding samples of these to the training set.

For each of the three methods (i.e., Correlation Filters, Individual PCA and FisherFaces), the error rates were found based on a threshold between authentics and impostors. Thresholds were placed on PSRs for Correlation filters, residues for Individual PCA, and on the projections for FisherFaces and used to find Equal Error Rates (EER).

4.1   Training with Normal Ambient Indoor Lighting of Round 1

The first process put the algorithms through the toughest test and tested the effects of distortion and time scale. 20 images from Set 1 of Round 1 of the normal ambient indoor lighting were used for training. Testing was done on all images of different illuminations except the training set. 4 UOTSDF filters using 5 consecutive training images each were built for each person. For Individual PCA, eigenvectors corresponding to 98% of energy, which was approximately 15, were stored for each person. For FisherFaces, projection was done Universal PCA using the maximum number of eigenvectors, i.e., 398, followed by Fisher LDA for maximum discrimination.

4.2   Training with Samples from all Illuminations of Round 1

Process 2 helps us understand the effects of images taken a certain time period apart. Samples of different illuminations from Set 1 of Round 1 were added to normal ambient light images. Since the outdoor images are very different, training and testing was separated between outdoor and indoor illuminations. Every 4th image was used, resulting in 25 training images (5 from ambient light and 5 each from other indoor light conditions) for indoor illuminations and 10 training images (5 from ambient light and 5 from outdoor light conditions) per person. For the indoor illumination test, approximately 17 eigenvectors were retained in Individual PCA to preserve 98%

energy while 498 eigenvectors from Universal PCA were used for projecting the images before Fisher LDA. The selective incremental updating approach for choosing and building multiple UOTSDF filters was taken leading to 5 to 12 filters being built depending on the person.

4.3   Training with Samples from all Illuminations as well as from Different Rounds

Process 3 was the simplest test that we put the methods through since images from different time scales were also included in the training set. Every 4th image from Set 1 of Round 2 of different illuminations was added to the training set of Process 2. By selective incremental updating, 9 to 22 UOTSDF filters per person were used, while the other methods used eigenvectors corresponding to 98% energy. FisherFaces used 221 eigenvectors for the indoor illumination test.

Table 2 shows the error rates for correlation filters, Individual PCA and FisherFaces for the three processes. As expected, the error rates reduce from Process 1 to Process 3 for all the 3 methods showing that adding the expected distortions to the training set improves performance. The error rates for images captured a month later are significantly larger in Process 1 and 2, implying that time scale plays a large role in face images.

It should be noted that FisherFaces uses impostor distributions that is not used by the 2 other methods. The number of eigenvectors used in Universal PCA is also very large contributing to some unfairness in comparison. Further, in practical applications, impostor distributions may not always be available.

By adding training images having distortions, correlation filters incorporate distortion tolerance. Hence by incrementally adding images of different illuminations to the correlation filters in Process 2, and different rounds in Process 3, the EER reduces and is smaller than that of Individual PCA and comparable to that of FisherFaces, in spite of the much larger number of eigenvectors used compared to the filters used. Hence we can say that correlation filters generalize better when the training set includes samples of expected distortion.

5   Conclusions

The cell phone camera database has been used to study the performance of some face verification algorithms in real life situations. The database has scale and pose distortions in addition to illumination and time-scale variations. The fish eye lens causes further distortion in the images with changes in orientation. Although there can be further improvement in performance with more efficient normalization, it is not possible to completely remove these distortions. Time scale plays a significant role in face images and error rates are larger for images captured a month later. If the correlation filters are incrementally updated with images over time, error rates reduce. The UOTSDF filter provides an easy way of incremental updating using images captured over a period of time without the need of building the filter from scratch. Despite the tough database, correlation filters performed better on an average than Individual PCA. The performance using a small number of incrementally updated

UOTSDF filters is comparable to that of FisherFaces using a large number of eigenvectors. By incrementally updating the correlation filters over time with images of different illuminations, the average EER is about 1% for images captured a month later. Correlations filters are efficient in terms of both memory usage and run time, which make them ideal, if they are to be implemented in low storage and computation capability devices such as cell phones or PDAs. In addition, correlation filters are shift invariant, provide graceful degradation and closed form solutions making their use attractive.

Acknowledgement

References

[1]   Turk, M., Pentland, A.: Eigenfaces for Recognition. J. Cognitive Neuroscience, Vol. 3, No. 1 (1991)
[2]   Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. Fisherfaces – recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol.19, Issue 7 (1997) 711-720
[3]   Vijaya Kumar, B.V.K.: Tutorial Survey Of Composite Filter Designs For Optical Correlators. Applied Optics, Vol. 31 (1992) 4773-4801
[4]   Réfrégier, Ph.: Optimal trade-off filters for noise robustness, sharpness of the correlation peak, and Horner efficiency. Optics Letters, Vol. 16 (1991) 829-831
[5]   Vijaya Kumar, B.V.K., Carlson, D. W., Mahalanobis, A.: Optimal Trade-Off Synthetic Discriminant Function Filters For Arbitrary Devices. Optics Letters, Vol. 19, No. 19, (1994) 1556-1558
[6]   Savvides, M., Venkataramani, K., Vijaya Kumar, B.V.K.: Incremental Updating of Advanced Correlation Filters for Biometric Authentication System. IEEE International Conference on Multimedia and Expo (ICME), Vol. 3 (2003) 229-232
[7]   Vijaya Kumar, B.V.K., Savvides, M., Xie,C., Venkataramani, K., Thornton, J.: Using Composite Correlation Filters for Biometric Verification. SPIE AeroSense, Vol. 5106 (2003)
[8]   Venkataramani, K: Reduced complexity correlation filters for fingerprint verification. MS Report. ECE department, Carnegie Mellon University (2002)
[9]   Liu, X., Chen, T., Vijaya Kumar, B.V.K.: Face Authentication for Multiple Subjects Using Eigenflow. Pattern Recognition, Special issue on Biometrics, Vol. 36, Issue 2, (2003) 313-328
[10]   Duda, R., Hart, P., Stork, D.: Pattern Classification. Second edition. NewYork: Wiley (2001)

Facial Expression Recognition
Based on Dimension Model of Emotion
with Autonomously Extracted Sparse Representations

Young-suk Shin

Department of Information and telecommunication Engineering, Chosun University
#375 Seosuk-dong, Dong-gu, Gwangju, 501-759, Korea
ysshin@mail.chosun.ac.kr

Abstract. This paper presents a facial expression recognition system based on dimension model of internal states with autonomously extracted sparse representations. Sparse representations of facial expressions are extracted to the three steps. In the first step, Gabor wavelet representation can extract edges of face components. In the second step, sparse features of facial expressions are extracted using fuzzy C-means(FCM) clustering algorithm on neutral faces, and in the third step, are extracted using the Dynamic Linking Model(DLM) on expression images. Finally, we show the recognition of facial expressions based on the dimension model of internal states using a multi-layer perceptron. With dimension model we have improved the limitation of expression recognition based on basic emotions, and have extracted features automatically with a new approach using FCM algorithm and the dynamic linking model.

1 Introduction

Currently, most facial expression recognition systems use the six principle emotions of Ekman [1]. Ekman considers six basic emotions: happiness, surprise, fear, anger, disgust, sadness; and categorizes facial expressions with these six basic emotions. Most research on facial expression recognition includes studies using the basic emotions of Ekman [2, 3, 4, 5], therefore these studies have limitations for recognition of natural facial expressions which consist of several other emotions and many combinations of emotions. Here we describe research extended on the dimension model of internal states for recognizing not only facial expressions of basic emotions but also expressions of various emotions.

Previous works on facial expression processing [2, 3, 4, 5] are similar in that they first extract some features from the images, then these features are used as inputs into a classification system. This paper describes a new hybrid approach for automatic feature extraction. To extract edge of major face components, the average value of the image's 2-D Gabor wavelet coefficient histogram on all the images was used. The hybrid approach using fuzzy c-means(FCM) clustering algorithm and dynamic linking model(DLM) is proposed to extract sparse local features from edges on expression images extracted previously. This conclusion demonstrates recognition of

facial expressions based on the two-dimensional structure of emotion using a neural network.

2   Sparse Representation of Facial Expression

To extract information of facial expression, we use 287 images of facial expressions, 640 × 480 gray level images almost in the frontal pose. Original images have been rescaled and cropped such that the eyes are roughly at the same position with a distance of 60 pixels in the final image. This section describes the process of three steps for feature extraction.

1. Preprocessing with Gabor Wavelets: For edges of major facial components, an average value of the image's 2-D Gabor wavelet coefficient histogram is used. The general form of two dimensional Gabor wavelets function is given by Daugman [6]. k determines the frequency of the wave and at the same time controls the width of the Gaussian window, σ .

$$\psi_{\vec{k}}(\vec{x}) = \frac{k^2}{\sigma^2}\exp(-\frac{k^2 x^2}{2\sigma^2})[\exp(i\vec{k}\cdot\vec{x}) - \exp(-\frac{\sigma^2}{2})] \qquad (1)$$

To detect features of major face components, we use a specific frequency band, a wave number, k=0.78, and 5 distinct orientations in 22.5 ° steps between 0 and π, and chose σ=π. The complex valued $\psi_{\vec{k}}$ applied to each image combines an even and odd part. We use only the magnitudes because they represent local information of an image in a smoothly varying way. Let G be the set of Gabor function $\psi_{\vec{k}}$ to be applied   to   *I*.   G   is G_1, G_2 .   The   computation   proceeds   as   follows:

$\omega_1 = \sum\sum G_1 I, \quad \omega_2 = \sum\sum G_2 I, \quad \varpi = \sqrt{(\omega_1^2 + \omega_2^2)}.$

Fig. 1(a) shows the result of the 2-D Gabor coefficients histogram using the magnitudes of Gabor coefficients from an expression image. This means these coefficients completely capture local facial feature points in special frequency and special orientation. Thus, we applied the average value   of 2-D Gabor coefficient histogram to extract local facial feature points. The average value of Gabor coefficients histogram is controlled by optional value ±α since experimental images may be a noise. Fig. 1(b) shows the resulting image which applied an optional value to an average value of the Gabor coefficients histogram.

| Red | Green | Blue | ☑ Greyscale |

(a)   (b)
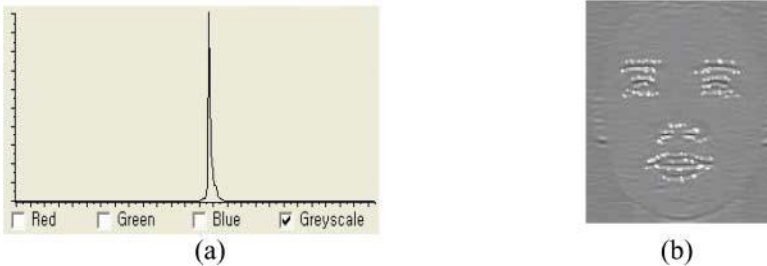
Fig. 1. (a) 2-D Gabor coefficient histogram. (b) Extracted edges of major face components

2. Sparse Feature Points Extraction Using FCM Clustering Algorithm on Neutral Face: Extracted feature points are similar to edges of major facial components. Since Gabor vectors with neighboring pixels are highly correlated and redundant, it is sufficient to use sparse pixels on a face. We thus pick out sparse feature points based on the FCM clustering algorithm in edges extracted from the 2-D Gabor wavelet coefficient histogram. FCM-algorithm [7] applies to neutral facial images that is used as a template to extract sparse feature points from edges of major facial components on expression images. The potentiality of FCM-algorithm can be demonstrated by their application in clustering tasks which involve a large number of feature vectors of high dimension and a large number of clusters [8]. In FCM clustering, the degree of the assignment of the feature vector $\mathbf{x}_i \in X$ into various clusters is measured by the membership function $u_{ij} \in [0,1]$, which satisfy the properties $\sum_{i=1}^{c} u_{ij} = 1, \forall_j = 1,...,N$. The cost function for FCM is $J(U,c_1,...,c_c) = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c}\sum_{j}^{N} u_{ij}^m d_{ij}^2$.

c_i is the cluster center of fuzzy group i; $d_{ij} = \|c_i - \mathbf{x}_j\|$ is the Euclidean distance between ith cluster center and jth data point ; and $m \in [1 < m ,\infty]$ is a weighting exponent. The necessary conditions for $J(U,c_1,...,c_c)$ to reach a minimum are

$$c_i = \sum_{j=1}^{N} u_{ij}^m X_j \Big/ \sum_{j=1}^{N} u_{ij}^m \text{ and } u_{ij} = 1 \Big/ \sum_{k=1}^{c} (\frac{d_{ij}}{d_{kj}})^{2/(m-1)}.$$

We determined sparse feature points using the following steps: (1) Initialize the membership matrix U with random values between 0 and 1 such that the constraints in $\sum_{i=1}^{c} u_{ij} = 1$ are satisfied. (2) Calculate c fuzzy cluster centers ($c_i, i = 1,2,....,c$) using c_i. (3) Compute the cost function according to $J(U,c_1,...,c_c)$, and stop if either it is below a certain tolerance value or its improvement over previous iteration is below a certain threshold. (4) Compute a new U using u_{ij}, then go to (2). Fig. 2(a) shows a result that extracted sparse pixel points by FCM algorithm.

3. Sparse Feature Points Extraction Using DLM on Expression Image: Each neutral face plays a standard role to decide the degree of expression change against an expression image. To match point to point feature points on an expression face against each feature point on a neutral face, it consists of two different domains, which are called the neutral domain (N) and the expression domain (E). The expression domain contains the jets of the Gabor transformation. The Gabor jet $\vec{J}(x_i)$

refers to the set of Gabor magnitudes obtained by sampling the image at the point $\vec{x_i}$ with sampling functions of all sizes (frequencies) and orientations. Sparse feature extraction using DLM on expression images is guided by a function S in $S(\vec{J}_i^N, \vec{J}_i^E) = \vec{J}_i^N \cdot \vec{J}_i^E \Big/ \|\vec{J}_i^N\|\|\vec{J}_i^E\|$ which determines the similarity between neutral face jet,

$\vec{J}_i^N$ and   expression image jet, $\vec{J}_i^E$. The entire wavelet family   consists of two frequency bands, the wave number $k = \|\vec{k}\| = (\pi/4, \pi/8)$ using inverse pixels and seven different orientations from 0° to 180°, differing in 30° steps.

(a)   (b)
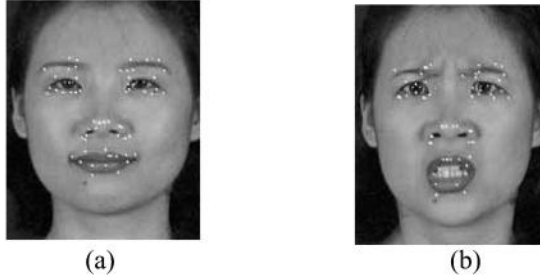
Fig. 2. (a) Sparse pixel points extracted with FCM algorithm on neutral face (c=60, m=2). (b) Sparse pixel points extracted with DLM on expression image

The linking procedure is performed under the constraint that the matching points found in the expression face have approximately the same topological relations as the preselected points in the neutral image. A match point should be chosen in the neutral face and then computed in the Euclidean distance between the preselected point in neutral face and each point in the expression image in $\vec{\Delta}_{ij}^{NE} = \vec{x}_i^{N} - \vec{x}_j^{E}$. This evaluates the quality of local topological preservation. The dynamic linking of selected points in the neutral face image to points in the expression image is formulated as an optimization problem. The cost function H in $H = \vec{\Delta}_{ij}^{NE} + \sum S(\vec{J}_i^{N}, \vec{J}_j^{E})$ to be optimized measures the quality of proposed point matches. We chose a special form for cost function. The feature on the expression images was accepted if the cost function H satisfies two conditions at the same time : (1) Reach to the minimum value. (2) Do not exceed a maximum distance value that the matching points found in the expression face have approximately the same topological relations as the preselected points in the neutral image(see Fig. 2(b)).

3   Database Based on Two-Dimensional Structure of Emotion

A database of facial expression images consists of 500 facial expression images of males and females under well controlled lighting condition. Expressions were divided into two dimensions(pleasure-displeasure and arousal-sleep dimension) on a nine point scale according to the study of internal states through the semantic analysis of words related with emotion by Kim et al. [9] using expressive 83 words. For experiment we used 11 expressions in a set of 44 internal state expressions from each of 6 person. The 11 expressions are happiness, surprise, sadness, disgust, fear, satisfaction, comfort, distress, tiredness, worry(including neutral face). The result of the dimension analysis of 44 emotion words related to internal emotion states is shown in Fig. 3. A few of these are shown in Fig. 4. This paper shows the recognition of facial expressions on dimension model.
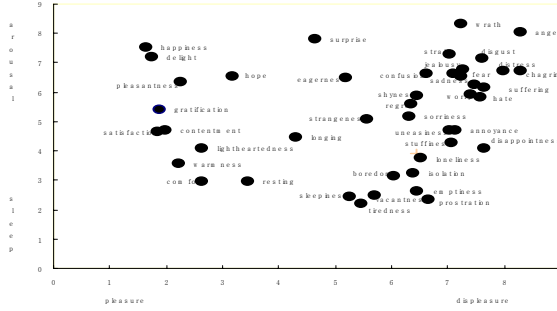
Fig. 3. Dimension Model: dimension analysis of 44 emotion words

Fig. 4. Examples from the facial expression database

4   Recognition Results and Evaluation

The system for facial expression recognition uses a three-layer neural network. The first layer is the distance values from each feature point on a neutral face to each feature point on an expression face which are normalized by size from 0 to 1. The second layer is 240 hidden units and the third layer is two output nodes to recognize the two dimensions: pleasure-displeasure and arousal-sleep. Training applies error back propagation algorithm which is well known to the pattern recognition field. The activation function of hidden units uses the sigmoid function. 250 images for training and 37 images excluded from the training set for testing are used. The first test verifies with the 250 images trained already. Recognition result produced by 250 images trained previously showed 100% recognition rates. The rating result of facial expressions derived from the semantic rating of emotion words by subjects is compared with experimental results of a neural network (NN). The similarity of recognition result between human and NN is computed in $S(\vec{H},\vec{N}) = \dfrac{\vec{H}\cdot\vec{N}}{\left\|\vec{H}\right\|\left\|\vec{N}\right\|}\min\left(\dfrac{\left\|\vec{H}\right\|}{\left\|\vec{N}\right\|},\dfrac{\left\|\vec{N}\right\|}{\left\|\vec{H}\right\|}\right)$.

The dimension values of human and NN in each two dimension are given as vectors of $\vec{H}$ and $\vec{N}$.

Table 1. The result data of expression recognition between human and NN

Emotion words	Human(Mean)		Neural Network		Recognition on Neural Network	Similarity
	P – D	A – S	P –D	A – S		
happiness	1.65	7.53	3.88	3.44	lightheartedness	0.54
			4.92	4.6	boredom	0.71
			2.86	5.86	pleasantness	0.82
			1.31	5.69	gratification	0.75
			4.43	4.8	longing	0.73
satisfaction	1.85	4.65	1.49	6.07	pleasantness	0.79
			2.14	4.96	contentment	0.92
			6.32	5.9	shyness	0.52
comfort	2.61	2.98	5.0	5.7	strangeness	0.52
			3.65	3.64	lightheartedness	0.77
sadness	7.22	6.57	7.07	5.23	shyness	0.89
			3.7	6.37	hope	0.72
			6.62	7.12	surprise	0.91
tiredness	5.44	2.2	7.94	6.29	strain	0.56
			4.06	4.05	sleepiness	0.90
			4.39	4.28	longing	0.89
			4.8	5.09	strangeness	0.76
			6.39	5.65	uneasiness	0.65
worry	7.4	5.96	6.89	6.09	confusion	0.97
			7.39	6.84	strain	0.94
surprise	4.65	7.8	4.55	8.29	surprise	0.95
			4.61	7.67	surprise	0.98
			4.65	5.60	hope	0.79
disgust	7.93	6.74	6.35	3.42	isolation	0.68
			7.33	6.14	hate	0.91
			7.68	6.03	distress	0.98
			6.05	6.72	surprise	0.86
Fear	7.25	6.77	6.75	4.49	sorriness	0.80
			6.43	5.21	stuffiness	0.83
			6.68	7.97	disgust	0.94
			7.30	7.96	chagrin	0.91
			5.91	4.17	isolation	0.72
distress	7.46	6.29	7.48	7.16	disgust	0.94
			4.28	5.81	hope	0.72
			4.77	4.97	boredom	0.70
			5.60	4.11	boredom	0.71
			5.81	5.05	strangeness	0.79

Table 1 describes a degree of similarity of expression recognition between human and NN on two-dimensional structure of emotion. In Table 1, the result of expression recognition of NN is matched to the most nearest emotion word in 44 emotion words related to internal emotion states. The result of expression recognition of NN looks very similar to the result of expression recognition of human. The

displeasure emotions of high level arousal dimension seems like fairly generalized and stabled emotions in expression space. Such expressions are surprise, disgust, fear, distress, and worry. It seems that the quantity of physical changes can be detected easily. High level arousal emotions in the pleasure dimension also seems like a important component to discriminate the expression images. For instance, satisfaction, tiredness, and comfort show low level arousal dimension (sleep dimension) in the pleasure dimension, while happiness shows high level arousal dimension in the pleasure dimension.

5   Conclusion

With dimension model this paper has improved the limitation of expression recognition based on basic emotions, and has extracted features automatically with a new approach using FCM-algorithm and the dynamic linking model. This method allows several advantages: it is the reduction of redundancy of information, it needs less data storage, it reduces the dimensionality of the learning problem. This study is a new approach of human's emotion processing, it is interesting to note in this context that machine vision may represent various emotions similar to human with the combination of each dimension in the internal emotion states. In the future, to demonstrate the effectiveness of the proposed scheme, we are planning comparison experiments with classic facial expression recognition system.

References

[1]   Ekman, P.: Universal and cultural difference in facial expressions of emotions. In: J. K. Cole(Ed.), Nebraska symposium on motivation, Lincoln: University of Nebraska Press, (1972) 207-283

[2]   Essa, I., Pentland, A.:Coding, analysis, interpretation, and recognition of facial expressions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19 (1997) 757-763

[3]   Lien, J.: Automatic recognition of facial expressions using hidden Markov models and estimation of expression intensity. Ph.D. Thesis, Carnegie Mellon University, (1998)

[4]   Tian, Y.L, Kanade, T., & Cohn, J. F.: Recognizing Action Units for Facial Expression Analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(2), (2001) 97-116

[5]   Cohen, I., Sebe, N., Garg, A., Chen, L. S., Huang, T. S.: Facial expression recognition from video sequence:temporal and static modeling. Computer Vision and Image Understanding , In Press (2003)

[6]   Daugman, J: Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. Journal of the Optical Society of America 2 (1985) 1160-1169

[7]   Bezdek, J.C.: Fuzzy mathematics in pattern classification. Ph.D. thesis, Applied Math. Center, Cornell University, Ithaca (1973)

[8]   Karayiannis, N.B., Pai, P.-I.: Fuzzy vector quantization algorithms and their application in image compression. IEEE Transactions on Image Processing, (1995)

[9]   Kim, Y., Kim, J., O, S., O, K., Chung, C.: The study of dimension of internal states through word analysis about emotion. Korean Journal of the Science of Emotion and Sensibility, 1 (1998) 145-152

Hallucinating Face by Eigentransformation
with Distortion Reduction

Xiaogang Wang and Xiaoou Tang

Department of Information Engineering, The Chinese University of Hong Kong
Shatin, Hong Kong
{xgwang1,xtang}@ie.cuhk.edu.hk
http://mmlab.ie.cuhk.edu.hk

Abstract. In this paper, we propose a face hallucination method using eigentransformation with distortion reduction. Different from most of the proposed methods based on probabilistic models, this method views hallucination as a transformation between different image styles. We use Principal Component Analysis (PCA) to fit the input face image as a linear combination of the low-resolution face images in the training set. The high-resolution image is rendered by replacing the low-resolution training images with the high-resolution ones, while keeping the combination coefficients. Finally, the nonface-like distortion in the hallucination process is reduced by adding constraints to the principal components of the hallucinated face. Experiments show that this method can produce satisfactory result even based on a small training set.

1   Introduction

For face identification, especially by human, it is often useful to render a high-resolution face image from the low-resolution one. This technique is called face hallucination or face super-resolution. A number of super-resolution techniques have been proposed in recent years [1][2][3][5][7][8][10]. Most try to produce a super-resolution image from a sequence of low-resolution images [1][5][7]. Some other approaches are based on learning from the training set containing high- and low-resolution image pairs, with the assumption that high-resolution images are Markov Random Field (MRF) [3][7][10]. These methods are more suitable for synthesizing local texture, and are usually applied to generic images without special consideration on the property of face images.

Baker and Kanade [8][9] develop a hallucination method based on the property of face image. Abandoning the MRF assumption, it infers the high frequency components from a parent structure by recognizing the local features from the training set. Liu et. al. [2] develop a two-step statistical approach integrating global and local models. Both of the two methods use complicated probabilistic models and are based on an explicit resolution reduction function, which is sometimes difficult to obtain in practice.

Instead of using a probabilistic model, we implement a face hallucination method using PCA to represent the structural similarity of face images. This method treats the

hallucination problem as the transformation between two different image styles. It is closely related to the work in [11][12][13], in which a style transformation approach was applied to photo-sketch transformation. In a similar way, we could transform face images from low-resolution to high-resolution based on mapping between two groups of training samples without deriving the resolution reduction function. The hallucinated face image is rendered from the linear combination of training samples. Even using a small training set, this method can produce satisfactory results.

2   Hallucination by Eigentransformation with Distortion Reduction

Viewing a 2D image as a vector, the process of resolution reduction can be formulated as [8]

$$\bar{I}_l = H\bar{I}_h + \bar{n} . \tag{1}$$

Here, $\bar{I}_h$ is the high-resolution face image vector to be rendered and $\bar{I}_l$ is the observed low-resolution face image vector. H is the transformation matrix involving blurring and downsampling process. The term $\bar{n}$ represents the noise perturbation to the low-resolution face image captured by the camera. Multiresolution analysis in spatial domain can produce different frequency bands. Because of the face structural similarity, the high frequency component is not independent of and thus can be inferred from the low frequency component encoded in the downsampled low-resolution image. Many super-resolution algorithms assume this dependency as homogeneous Markov Random Fields (MRFs), i.e. the pixel only relies on the pixels in its neighborhood. This is an assumption for general images, but not optimal for the face set without considering face structural similarity.

Studies [6] on the face set have shown that a face image can be reconstructed from eigenfaces in the PCA representation. PCA utilizes the face distribution to decompose face structure into uncorrelated frequency components, thus can encode face information more concisely. Our algorithm first employs PCA to extract as much useful information as possible from a low-resolution face images, and then renders a high-resolution face image by eigentransformation.

We represent M low-resolution training face images by a matrix, $[\bar{l}_1,...\bar{l}_M]$, where $\bar{l}_i$ is the image vector. In PCA, a set of eigenvectors $E_l = [e_1,...,e_K]$, also called eigenfaces, are computed from the ensemble covariance matrix,

$$C = \sum_{i=1}^{M} (\bar{l}_i - \bar{m}_l)(\bar{l}_i - \bar{m}_l)^T = LL^T , \tag{2}$$

where $\bar{m}_l$ is the mean face and $L = [\bar{l}_1 - \bar{m}_l,...,\bar{l}_M - \bar{m}_l] = [\bar{l}'_1,...\bar{l}'_M]$.

For a low-resolution face image $\bar{x}_l$, a weight vector is computed by projecting it onto eigenfaces,

$$\bar{w}_l = E_l^T (\bar{x}_l - \bar{m}_l). \tag{3}$$

This is a face representation based on eigenfaces. $\bar{x}_l$ can be reconstructed from the K eigenfaces,

$$\bar{r}_l = E_l \bar{w}_l + \bar{m}_l . \tag{4}$$

According to the singular value decomposition theorem, E_l also can be computed from,

$$E_l = L V_l \Lambda_l^{-1/2} , \tag{5}$$

where V_l and Λ_l are the eigenvector and eigenvalue matrix for $L^T L$. We have

$$\bar{r}_l = L V_l \Lambda_l^{-1/2} \bar{w}_l + \bar{m}_l = L\bar{c} + \bar{m}_l = \sum_{i=1}^{M} c_i \bar{l}'_i + \bar{m}_l , \tag{6}$$

where $\bar{c} = V_l \Lambda_l^{-1/2} \bar{w}_l = [c_1, c_2, \ldots, c_M]^T$. This shows that the input low-resolution face image $\bar{x}_l$ can be reconstructed from the optimal linear combination of the M low-resolution training face images. Replacing $\bar{l}'_i$ by its high-resolution sample $\bar{h}'_i$, and replacing $\bar{m}_l$ by the high-resolution mean face $\bar{m}_h$, we get,

$$\bar{x}_h = \sum_{i=1}^{M} c_i \bar{h}'_i + \bar{m}_h . \tag{7}$$

$\bar{x}_h$ is expected to be an approximation to the real high-resolution face image.

$\bar{x}_h$ should meet two conditions in order to accurately approximate the original high-resolution face image. First, after resolution reduction of $\bar{x}_h$, the output should produce the low-resolution input face image. From Eq. (1), without considering the noise perturbation, the transformation between high- and low- resolution face images can be approximated as a linear operation. For the training set, we have

$$\bar{l}'_i = H\bar{h}'_i , \tag{8}$$

$$\bar{m}_l = H\bar{m}_h . \tag{9}$$

From (6) and (7), replacing $\bar{l}'$ and $\bar{m}_l$ with (8) and (9), we have

$$\bar{r}_l = \sum_{i=1}^{M} c_i H\bar{h}'_i + H\bar{m}_h = H\left(\sum_{i=1}^{M} c_i \bar{h}'_i + \bar{m}_h \right) = H\bar{x}_h . \tag{10}$$

Since $\bar{r}_l$ is an optimal reconstruction to $\bar{x}_l$, $\bar{x}_h$ leads to a good approximation to $\bar{x}_l$ after resolution reduction. Second, $\bar{x}_h$ should be face-like at the high-resolution level. Eq. (7) shows that $\bar{x}_h$ is the linear combination of high-resolution face images, so it should approximately be face-like at high-resolution level. We can further reduce some nonface-like distortions by reconstructing $\bar{x}_h$ from the high-resolution eigenfaces. Let E_h and $\Lambda_h = diag(\lambda_1, \ldots, \lambda_K)$ be the eigenface and eigenvalue matrixes computed from the high-resolution training images. The principal components of $\bar{x}_h$ projected onto the high-resolution eigenfaces are

$$\bar{w}_h = E_h^T (\bar{x}_h - \bar{m}_h). \tag{11}$$

The eigenvalue λ_i is the variance of high-resolution face images on the ith eigenface. If the principal component $w_h(i)$ is much larger than $\sqrt{\lambda_i}$, nonface-like distortion may be involved for the ith eigenface dimension. We apply constraints on the principal components,

$$\vec{w}'_h(i) = \begin{cases} w_h(i) & |w_h(i)| \le a\sqrt{\lambda_i} \\ sign(w_h(i))*a\sqrt{\lambda_i} & |w_h(i)| > a\sqrt{\lambda_i} \end{cases}, \qquad a > 0 \qquad (12)$$

We use $a\sqrt{\lambda_i}$ to bound the principal component. Here, a is a positive scale parameter. The final hallucinated face image is reconstructed by

$$\vec{x}'_h = E_h^T \vec{w}'_h + \vec{m}_h . \qquad (13)$$

Because of the structural similarity among face images, in multiresolution analysis, there exists strong correlation between the high and low frequency bands. For high-resolution face images, PCA can compact these correlated information onto a small number of principal components. Then, in the eigentransformation process, these principal components can be inferred from the principal components of the low-resolution face image by mapping between the high- and low-resolution training pairs. Therefore, some information in the high frequency bands is partially recovered.

In practice, the low-resolution image is often disturbed by noise which has a flat distribution on all the eigenvectors. For low-resolution face images, the energy on small eigenvectors is sometimes overwhelmed by noise. If the face data is modeled as a Gaussian distribution, the components on different eigenvectors are independent. So the information on these noisy components is lost, and cannot be recovered. By selecting an optimal eigenface number K, we can remove the noise. Since $\vec{r}_l$ is reconstructed from the K eigenfaces, given an optimal value of K, $\vec{r}_l$ encodes the maximum amount facial information recoverable in the low-resolution face image.

3   Experiments

The hallucination experiment is conducted on a data set containing 188 individuals with one face image for each individual. Using the "leave-one-out" methodology, at each time, one image is selected for testing and the remaining are used for training. In preprocessing, the face images are aligned by the two eyes, and the image size is fixed at 117×125. Images are blurred by averaging neighbour pixels and down sampled to low-resolution images with size 23×25.

Some hallucination results are shown in Fig. 1. Compared with the input image and the Cubic B-Spline interpolation result, the hallucinated face images have much clearer detail features. They are good approximation to the original high-resolution images.

In Fig. 2, we add Gaussian noise to the low-resolution face image. If no constraint is add to the principal components, the hallucinated face image in Fig. 2 (d) is with noise distortion and somewhat nonface-like. Adding constraints to the principal components using Eq. (12), the reconstructed face images remove most of the noise

distortion and retain most of the facial characteristics as shown in Fig. 2 (e). Here, we set the parameter *a* at 2.

As discussed in Section 2, by selecting the eigenface number in eigentransformation, we could control the detail level by keeping the maximum facial information while removing most of the noise disturbance. In Fig. 3, we add zero mean Gaussian noises with four different standard deviations (σ) to the low-resolution face image. When only 50 eigenfaces are used in the eigentransformation, the hallucinated face images lose some individual characteristics. Although the edges and contours are clear, the hallucinated faces are more like a mean face. When eigenface number is increased to 180, more individual characteristics are added to the hallucinated face images.   For relatively small noise ($\sigma = 0.03, 0.05$), these characteristics are similar to the original high-resolution face image. But for large noise ($\sigma = 0.1, 0.2$), even though the hallucinated faces are still face-like, the added characteristics start to deviate from those of true face.

(a) Input 23×25 (b) Cubic B-Spline   (c) Hallucinated   (d) Original 117×125
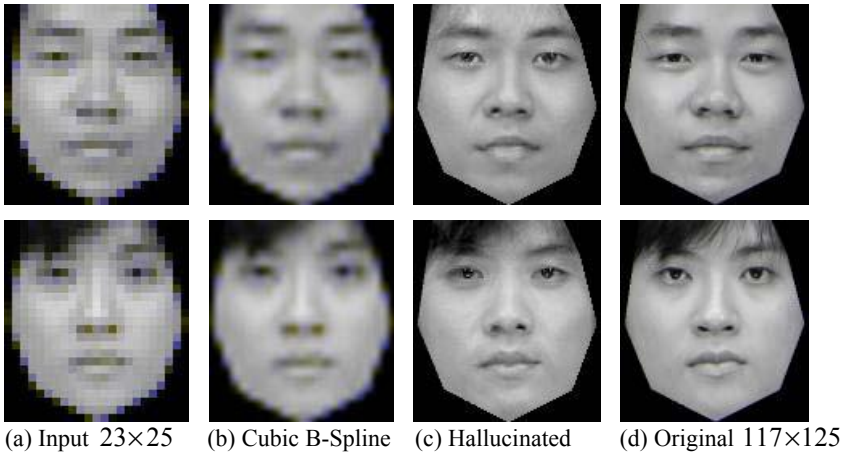
Fig. 1. Hallucinated face images by eigentransformation

(a)   (b)   (c)   (d)   (e)
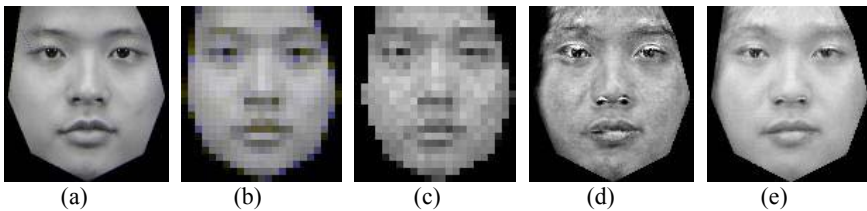
Fig. 2. Adding constraints to the principal component of hallucinated face image. (a) Original high resolution face image; (b) Low resolution face image; (c) Low-resolution face image added zero mean 0.03 standard deviation Gaussian noise; (d) Hallucinated face image from (c) without constraints on the principal components; (e) Hallucinated face images from (c) with constraints on principal components. (e) is more face-like and less noisy comparing to (d), and it retains most of the facial characteristics of (d)

(a)

($\sigma = 0.03$)   ($\sigma = 0.05$)   ($\sigma = 0.1$)   ($\sigma = 0.2$)

(b)

(K=50, $\sigma = 0.03$) (K=50, $\sigma = 0.05$) (K=50, $\sigma = 0.1$) (K=50, $\sigma = 0.2$)

(K=180, $\sigma = 0.03$)(K=180, $\sigma = 0.05$)(K=180, $\sigma = 0.1$)(K=180, $\sigma = 0.2$)

(c)
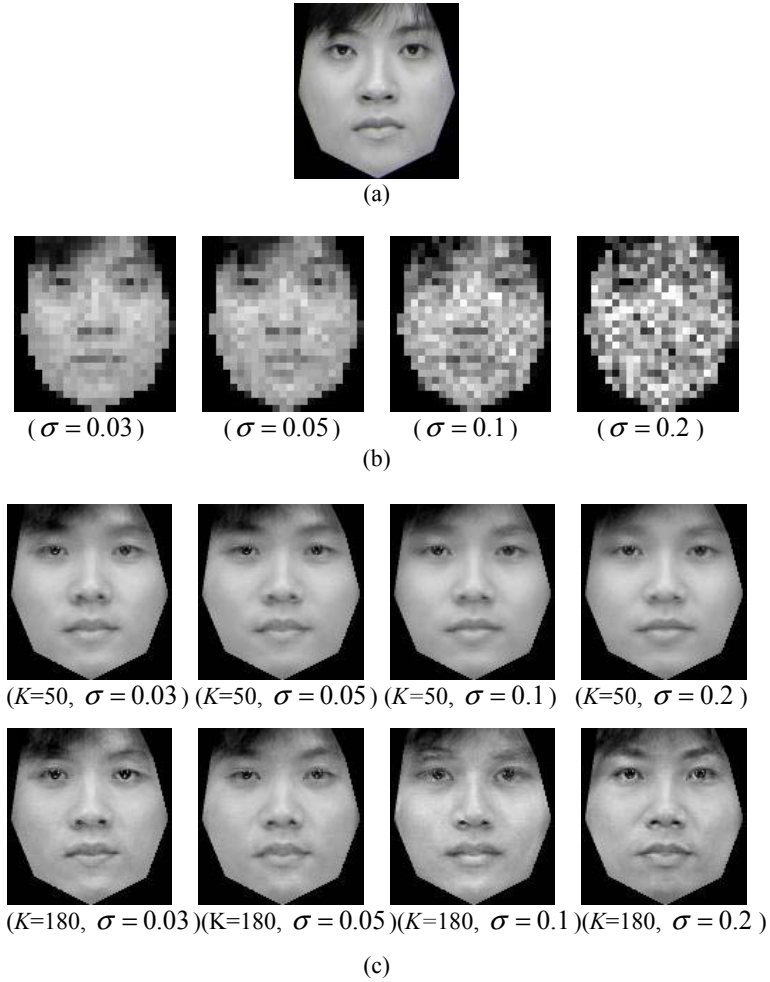
Fig. 3. Hallucinating face with additive zero mean, Gaussian noise. (a) Original high-resolution face image; (b) Low-resolution face images with noise; (c) Hallucinated face images using different eigenface number. K is the eigenface number in eigentransformation, and σ is the standard variation of Gaussian noise

4   Conclusion

Because of the structural similarity, face images can be synthesized from the linear combination of other samples. Based on this property of face images, hallucination can be implemented by eigentransformation. By selecting the energy level in the PCA representation, our method extracts maximum facial information from the low-resolution face images and is robust to noise. The resolution and quality of face images are greatly improved over the low-resolution images.

Acknowledgement

References

[1]   Patti, M. Sezan, and A. Tekalp, "Super-resolution Video Reconstruction with Arbitrary Sampling Latices and Nonzero Aperture Time," *IEEE Trans. on Image Processing*, Vol. 6, No. 8, pp. 1064-1076, 1997.

[2]   C. Liu, H. Shum, and C. Zhang, " A Two-Step Approach to Hallucinating Faces: Global Parametric Model and Local Nonparametric Model," *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition,* pp. 192-198, 2001.

[3]   J.D. Bonet, "Multiresolution sampling procedure for analysis and synthesis of texture images," *Proceedings of SIGGRAPH 97*, pp. 361-368, 1997.

[4]   K. Fukunnaga, "Introduction to Statistical Pattern Recognition", Academic Press, second edition, 1991.

[5]   M. Elad and A. Feuer, "Super-Resolution Reconstruction of Image Sequences," *IEEE Trans. on PAMI,* Vol. 21, No. 9, 1999.

[6]   M. Turk and A. Pentland, "Eigenface for Recognition," *J. of Cognitive Neuroscience*, Vol. 3, No. 1, pp. 71-86, 1991.

[7]   R. Hardie, K. Barnard, and E. Armstrong, "Joint MAP registration and high-resolution image estimation using a sequence of undersampled images," *IEEE Trans. on Image Processing*, Vol. 6, No. 12, pp. 1621-1633, 1997.

[8]   S. Baker and T. Kanade, "Limits on Super-Resolution and How to Break them," *IEEE Trans. on PAMI*, Vol. 24, No. 9, pp. 1167-1183, 2002.

[9]   S. Baker and T. Kanade, "Hallucinating Faces," *Proceedings IEEE International Conference on Automatic Face and Gesture Recognition,* pp. 83-88, 2000.

[10]   W.T. Freeman and E.C. Pasztor, "Learning Low-Level Vision," *Proceedings of IEEE International Conference on Computer Vision*, 1999.

[11]   X. Tang and X. Wang, "Face Photo Recognition Using Sketch," *Proceedings of ICIP,* pp. 257-260, 2002.

[12]   X. Wang and X. Tang, "Face Sketch Recognition," IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Image- and Video-Based Biometrics, Vol. 14, No.1, pp. 50-57, Jan. 2004.

[13]   X. Wang, and X. Tang, "Face Hallucination and Recognition," Proceedings of the 4th International Conference on Audio- and Video-Based Person Authentication (AVBPA), 2003.

An Optimal Subspace Analysis for Face Recognition

Haitao Zhao[1,2], Pong C. Yuen[1], and Jingyu Yang[2]

[1] Department of Computer Science, Hong Kong Baptist University, Hong Kong
{htzhao,pcyuen}@comp.hkbu.edu.hk
[2] Department of Computer Science, Nanjing University of Science and Technology
Nanjing 210094, P. R. China
jyyang@njust.edu.ch

Abstract. Fisher Linear Discriminant Analysis (LDA) has recently been successfully used as a data discrimination technique. However, LDA-based face recognition algorithms suffer from a small sample size (S3) problem. It results in the singularity of the within-class scatter matrix S_w. To overcome this limitation, this paper has developed a novel subspace approach in determining the optimal projection. This algorithm effectively solves the small sample size problem and eliminates the possibility of losing discriminative information.

1 Introduction

The objective of Fisher Linear Discriminant Analysis LDA is to find a projection X^*, from original sample feature space to a transformed lower-dimensional feature space, i.e.,

$$X^* : R^n \to R^m, n > m, \quad X^* = \arg\max_X \frac{\det(X^T S_b X)}{\det(X^T S_w X)}$$

LDA suffers from a small sample size (S3) problem. This problem occurs when the sample size is smaller than the dimension of the feature vectors. Under this situation, S_w becomes singular and direct calculation of its inverse is not feasible. In view of this limitation, this paper compares and evaluates the existing LDA-based methods and proposes a new solution to this problem. It effectively solves the small sample size problem and eliminates the possibility of losing discriminative information.

The rest of this paper is organized as follows: Section 2 reviews the related work on LDA-based algorithms in face recognition; Section 3 introduces our new optimal subspace algorithms; Section 4 gives comparison analysis and discussion; the comparison experiments are shown and discussed in Section 5; and. Section 6 concludes this paper.

2   Previous Work

2.1 Related Work

Suppose there are c known pattern classes of N training samples in total, and the original feature space is n-dimensional. S_b, S_w and S_t denote between-class scatter matrix, within-class scatter matrix and total scatter matrix respectively. As we know, they are all non-negative definite, and $S_t = S_b + S_w$.

Fisher discriminant criterion function is defined as follows: $J(X) = \dfrac{X^T S_b X}{X^T S_w X}$

where X is an n-dimensional nonzero vector. The goal of LDA is to optimize the extracted features for the purpose of classification.

One classical kind of LDA-based methods includes Fisherfaces [3], EFM [9] and the uncorrelated LDA [8]. These methods may remove the small eigenvalues (including zero eigenvalues) in S_w means that we discard the null space of S_w which contains useful discriminant information. In turn, the performance will be degraded.

To deal with this problem, Chen [5] proposed to keep the null space of S_w. Their idea is sound and the developed theory provides a solid foundation, but their algorithms have to run in the high-dimensional original feature space and would suffer from computational problems. In view of the limitation of Chen's method, Yu [2] proposed a new method, called Direct LDA. This method suffers from the performance limitation. This is because discarding the null space of S_b would lead to the losing of the null space of S_w indirectly. In 2002, Huang [4] proposed to remove the null space of S_t, and then keep the null space of S_w', finally discard the null space of S_b'', where S_w' and S_b'' are the dimension reduced version of S_w and S_b respectively. But in Huang method, the matrix S_b'' may not be full rank and some discriminatory information maybe lost.

3   Proposed Method

A new method is proposed to solve S3 problem in LDA. The new method optimizes both the performance and complexity.

Lemma 1[1] Suppose that A is non-negative definite matrix, X is n-dimensional vector, then $X^T A X = 0$ if and only if $AX = 0$.

Lemma 2[4] If S_t is singular, $X^T S_t X = 0$ if and only if $X^T S_w X = 0$ and $X^T S_b X = 0$.

Since S_w, S_b are non-negative definite and $S_t = S_b + S_w$, the above lemma is easy to get.

Since S_t is non-negative definite, there exist an orthogonal matrix U, such that $U^T S_t U = \Lambda$ where $\Lambda = diag(a_1, \cdots, a_n)$, $a_1, \cdots, a_n$ are eigenvalues of S_t, and

$a_1 \geq a_2 \geq \cdots \geq a_p > 0$, $a_{p+1} = a_{p+2} = \cdots = a_n = 0$. Let $\tilde{\Lambda} = diag(a_1, \cdots, a_p)$,

$W = [u_1, u_2, \cdots, u_p]\tilde{\Lambda}^{-\frac{1}{2}}$ and $\overline{W} = (u_{p+1}, \cdots, u_n)$.

Consider $\tilde{S}_w = W^T S_w W$. It is easy to prove that $\tilde{S}_w$ is nonnegative definite, so there exist p orthogonal eigenvectors $\xi_1, \cdots, \xi_p$ and the associated eigenvalues $\lambda_1 \leq \cdots \leq \lambda_p$, such that $\tilde{S}_w \xi_j = \lambda_j \xi_j$, for $j = 1, \cdots, p$.

Let $X_1 = W\xi_1, \cdots, X_p = W\xi_p$, we can get the following theorem.

Theorem 1 eigenvectors $X_1, \cdots, X_p$ and corresponded eigenvalues $\lambda_1, \cdots, \lambda_p$ are the solution of the eigenequation $S_w X = \lambda S_t X$, $j = 1, \cdots, p$.

Proof:

Since $\tilde{S}_w \xi_j = \lambda_j \xi_j$, ($j = 1, \cdots, p$), that is $W^T S_w W \xi_j = \lambda_j \xi_j$,

Let $\zeta_j = \left.\begin{bmatrix} \xi_j \\ 0 \\ \vdots \\ 0 \end{bmatrix}\right\} n - p$, then dimension of ζ_j is n .

Let $V = (W, \overline{W})$, by the definition of $\overline{W}$, it follows that $\overline{W}^T S_t \overline{W} = 0$. By lemma 1, we have $\overline{W}^T S_w \overline{W} = 0$. Since S_w is non-negative definite matrix, by lemma 2, we have $S_w \overline{W} = 0$.

Consider $V^T S_w V \zeta_j = \begin{pmatrix} W^T \\ \overline{W}^T \end{pmatrix} S_w (W, \overline{W}) \begin{pmatrix} \xi_j \\ 0 \end{pmatrix} = \begin{pmatrix} W^T S_w \\ \overline{W}^T S_w \end{pmatrix} (W, \overline{W}) \begin{pmatrix} \xi_j \\ 0 \end{pmatrix}$

$$= \begin{pmatrix} W^T S_w W & W^T S_w \overline{W} \\ \overline{W}^T S_w W & \overline{W}^T S_w \overline{W} \end{pmatrix} \begin{pmatrix} \xi_j \\ 0 \end{pmatrix} \tag{1}$$

Since $S_w \overline{W} = 0$, equ.(1) is equal to

$$V^T S_w V \zeta_j = \begin{pmatrix} W^T S_w W & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \xi_j \\ 0 \end{pmatrix} \quad \begin{pmatrix} W^T S_w W \xi_j \\ 0 \end{pmatrix} \quad \begin{pmatrix} \lambda_j \xi_j \\ 0 \end{pmatrix} \quad \lambda_j \zeta_j$$

hence $S_w V \zeta_j = \lambda_j (V^T)^{-1} \zeta_j$.. \tag{2}

Since $V^T S_t V \zeta_j = \begin{pmatrix} W^T S_t W & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \xi_j \\ 0 \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \xi_j \\ 0 \end{pmatrix} = \zeta_j$,

we have $S_t V \zeta_j = (V^T)^{-1} \zeta_j$. \tag{3}

Substituting equ. (3) into equ. (2), we have $S_w V \zeta_j = \lambda_j S_t V \zeta_j$.

Since $V \zeta_j = (W, \overline{W}) \begin{pmatrix} \xi_j \\ 0 \end{pmatrix} = W \xi_j = X_j$,

it follows that $S_w X_j = \lambda_j S_t X_j$, $j = 1, \cdots, p$.

Corollary 1 $X_i^T S_b X_j = \begin{cases} 1-\lambda_j & i=j \\ 0 & i \neq j \end{cases}$ $i,j=1,\cdots,p$

Since $X_i^T S_t X_j = \delta_{ij} = \begin{cases} 1 & i=j \\ 0 & i \neq j \end{cases}$, $X_i^T S_w X_j = \begin{cases} \lambda_j & i=j \\ 0 & i \neq j \end{cases}$ $i,j=1,\cdots,p$ and

$S_b = S_t - S_w$, it follows that $X_i^T S_b X_j = \begin{cases} 1-\lambda_j & i=j \\ 0 & i \neq j \end{cases}$ $i,j=1,\cdots,p$.

The proposed LDA method is subjected to S_t-orthogonal constraint. From the statistical correlation point of view, the projection is also optimal [8].

Corollary   2   If $\lambda_1 = \lambda_2 = \cdots = \lambda_u = 0$, $0 < \lambda_{u+1} < \cdots < \lambda_p$, eigenvectors $X_{u+1}, \cdots, X_p$ and   corresponded   eigenvalues

$\dfrac{1-\lambda_{u+1}}{\lambda_{u+1}}, \dfrac{1-\lambda_{u+2}}{\lambda_{u+2}}, \cdots, \dfrac{1-\lambda_p}{\lambda_p}$ are the solution of the eigenequation $S_b X = \lambda S_w X$,

$j=u+1, \cdots, p$.

Since $S_w X_j = \lambda_j S_t X_j$, $j=u+1, \cdots, p$, it follows that $S_w X_j = \lambda_j (S_b + S_w) X_j$. As

$0 < \lambda_{u+1} < \cdots < \lambda_p$, then we can get $S_b X_j = \dfrac{1-\lambda_j}{\lambda_j} S_w X_j$, $j=u+1, \cdots, p$.

By Theorem 1, we can draw the following conclusion:

(1) If $\lambda_1 = \lambda_2 = \cdots = \lambda_r = 0$, $0 < \lambda_{u+1} < \cdots < \lambda_p$, by lemma 2 and Corollary 1, it follows that $X_j^T S_b X_j \neq 0$, $X_j^T S_w X_j = 0, j=1, \cdots, u$.

So the Total-scatter(u) $\sum\limits_{j=1}^{u} X_j^T S_b X_j$ is maximum. As for $0 < \lambda_{u+1} < \cdots < \lambda_p$, by Corollary 2, it follows that $Y = [X_{u+1}, \cdots, X_p]$ is the optimal projection According to the criterion function $J(Y) = \dfrac{\det(Y^T S_b Y)}{\det(Y^T S_w Y)}$.

(2) If $0 < \lambda_1 < \cdots < \lambda_p$, by Corollary 2, it follows that $Y = [X_1, \cdots, X_p]$ is the optimal projection According to the criterion function $J(Y) = \dfrac{\det(Y^T S_b Y)}{\det(Y^T S_w Y)}$.

The new algorithm and the computational consideration are described as follows:
First get the prewhitening transformation matrix W , which is defined by $W = [u_1, u_2, \cdots, u_p] \widetilde{\Lambda}^{-\frac{1}{2}}$. Next, let $\widetilde{S}_w = W^T S_w W$, and $v_1, \cdots, v_d$ are its eigenvectors

corresponding to the first d smallest eigenvalues. Let $V = (v_1, \cdots, v_d)$, then the optimal subspace LDA algorithm is $Y = P^T X$, where $P = WV$.

4   Analysis

Since the within-class scatter matrix includes the face variation for the same individuals and the between-class scatter matrix includes the variation between mean face images [6], we can neither overlook the discriminatory information within the null space of within-class scatter matrix or overlooks the discriminatory information outside the null space of between-class scatter matrix. Comparing with Fisherface method [3], Direct LDA method [2], and Huang's method [4], Our algorithm is the best one that fulfills the idea above.

By the experiment results, our algorithm outperforms other three typical methods. Especially when the sample size is small, which is the problem we definitely want to resolve. As to the computational complexity [4], the most time-consuming procedure, eigen-analysis, is performed on two matrices ($c \times c, c-1 \times c-1$) in Direct LDA, three matrices ($N \times N, (N-1) \times (N-1), c \times c$) in Huang method and three matrices (one of $N \times N$, two of $(N-c) \times (N-c)$) in Fisherface method. While, our method performs on two matrices ($N \times N$ and $N-1 \times N-1$). It can be found our method gets the best results with a relatively low computational complexity.

5   Experiment Results

To demonstrate the efficiency of our method, extensive experiments are performed on different face image databases.

5.1 Using ORL Database

In ORL database, there are 40 persons and each person consists of 10 images with different facial expressions, small variations in scales and orientations. The resolution of each image is 112×92.

We randomly selected n images from each individual for training while the rest are for testing. It is known that the results will be different with different training images. To smooth out the fluctuations, the experiments are repeated 50 times and the recognition rate (%) is then calculated under a minimum distance classifier. For each value of n, we choose 39 as the finial dimension. The average accuracy is charted in Figure 2. For each experiment, the list of training images and testing images are recorded and stored. When comparing with other methods, same list will be used.

In order to compare the performance of the proposed method with existing methods, the same experiments are conducted using Fisherface method [3], Direct LDA method [2], and Huang's method [4]. Please note that, to have a fair comparison, the same lists of training images are used for all methods. The results are shown in Figure 2.

5.2 Using FERET Database

To experiment on more challenging data, we have selected 72 subjects from the FERET database, with 6 images for each subject. The six images are extracted from 4 different sets, namely Fa, Fb, Fc and duplicate [7]. All images are aligned at the centers of eyes and mouth and then normalized with resolution 112×92. This resolution is the same as that in ORL database.

The experiment setting for the FERET database is similar with that of ORL database. the number of training images is ranged from 2 to 5. The experiments are also repeated 50 times under a minimum distance classifier. For each value of n, we choose 71 as the finial dimension. the average accuracy is charted in Fig. 2.
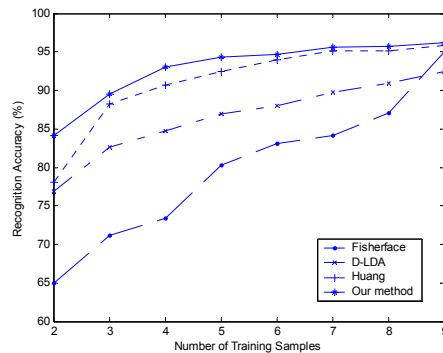
Fig. 1. Comparison with other methods on ORL databases
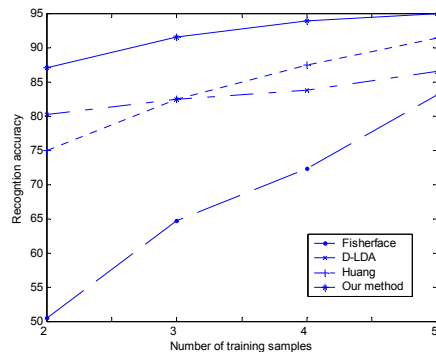
Fig. 2. Comparison with other method on the FERET dataset

6   Conclusions

This paper has developed and reported a new method for solving the small sample size (S3) problem in Fisher Linear Discriminant Analysis. A new idea of getting optimal subspace is proposed and a new algorithm is then derived. The proposed algorithm is an alternative to currently accepted methods that do not take advantages of both discriminant information within and outside of the null space of the within-class scatter matrix. Without the computation of the subspace space of the between-class matrix, this paper develops an optimal subspace LDA algorithm. Through theoretical derivation, we can find this algorithm effectively solves the S3 problem and eliminates the possibility of losing discriminative information.

Acknowledgement

References

[1]   K. Liu, Y. Cheng, J. Yang, "Algebraic feature extraction for image recognition based on an optimal discriminant criterion". Pattern Recognition, Vol. 26, No. 6, pp.903-911, 1993.

[2]   H. Yu and J. Yang, "A direct LDA algorithm for high-dimensional data --- with application to face recognition", Pattern Recognition, Vol. 34, pp.2067-2070, 2001.

[3]   P. N. Belhumeur, J. P. Hespanha, D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection", IEEE Trans. Pattern Anal. Mach. Intell., Vol. 19, No. 7. pp.711-720, 1997

[4]   R. Huang, Q. Liu, H. Lu and S. Ma, "Solving small sample size problem in LDA", Proceeding of International Conference in Pattern Recognition (ICPR 2002), Vol. 3, pp.29-32, Aug 2002.

[5]   L. Chen, H. Liao, M. Ko, J. Lin and G. Yu, "A new LDA-based face recognition system which can solve the small sample size problem", Pattern Recognition, Vol. 33 No. 10, pp.1713-1726, 2000

[6]   X. Wang, X. Tang, "Unified Subspace Analysis for Face Recognition", Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV' 03). 2003.

[7]   P. J. Phillips, H. Moon, S. A. Rizvi and P. J. Rauss, "The FERET evaluation methodology for face recognition algorithms". IEEE Trans. Pattern Anal. Mach. Intell., Vol. 22, No. 10. pp.1090-11104, 2000.

[8]   Z. Jin, J. Y. Yang, Z. S. Hu and Z. Lou, Face recognition based on the uncorrelated discriminant transform, Pattern Recognition, 2001,34, pp.1405-1416.

[9]   C. Liu and H. Wechsler, "Robust coding schemes for indexing and retrieval from large face databases", IEEE Trans. Image Processing, Vol. 9, No. 1, pp.132-137, 2000.

A Still-to-Video Face Verification System
Using Advanced Correlation Filters

Chunyan Xie[1], B. V. K. Vijaya Kumar[1], S. Palanivel[2], and B. Yegnanarayana[2]

[1]Dept. of ECE., Carnegie Mellon University, Pittsburgh, PA, 15213, USA
{kumar, chunyanx}@ece.cmu.edu
[2]Dept. of CSE, Indian Institute of Technology, Madras, Chennai-600 036, India.
{yegna,spal}@cs.iitm.ernet.in

Abstract. In this paper we propose to use the technology of multiple correlation filters to address the issues involved in still-reference-to-video-sequence face verification. In particular, we assume that the gallery consists of one or a few digital images of a person's face as reference and the probe set consists of video sequences as test data. The test data consists of a sequence of face images of a naturally moving person, captured from one or more cameras. An automatic face recognition system involves face/head tracking and cropping, correlation filter based face matching, and combining the evidence from multiple frames of the face images.

1 Introduction

Biometric verification is being investigated for many access control and e-commerce applications as biometric signatures offer the advantage that they cannot be stolen or forgotten like passwords. Examples of biometrics include face images, fingerprints, voice patterns, iris patterns, etc. Biometric verification (or authentication) refers to the task of determining whether a stored biometric matches a live biometric or not and thus determining whether the claimant of an identity is an authentic or an impostor. A closely related problem is biometric identification, in which a live biometric may be matched against a database of biometrics to determine the identity of the subject. Sometimes the term recognition is used to refer both of them. In this paper, we focus on the problem of subject verification based on face images.

Face verification is attractive because face images can be captured in a non-intrusive way, but the recognition performance drops for many face recognition methods when the test image does not enjoy the same geometric and photometric conditions as the training images, e.g., frontal view vs. side view. To address the problem of pose difference, video-to-video face recognition approaches have been proposed [1,2], where both gallery and probe sets consist of video sequences containing the face in some frames. Such video-based face recognition systems exploit much more information than still image based face recognition systems. However, most current video-based face recognition methods have been tested only for the cases where the training video sequence and the test video sequence are captured in the same conditions. Also, the video-to-video approaches require video

sequences for training, which may not be available in some applications. These are some of the factors that limit the application of video-to-video face recognition.

We propose in this paper a still-to-video face verification system based on correlation filters. The gallery consists of a few high-quality digital images of a person's face as reference and the probe set consists of video sequences of a person moving normally in an area. The test subject may not be cooperative, in the sense of providing defined views to the camera. In this scenario, our hypothesis is that those frames that have a view approximately corresponding to the views in the training images are able to provide partial evidence of the identity of the test subject. Combining those partial evidences from multiple frames may lead to a better verification decision.

For this method to succeed, the matching algorithm for a single frame should have a very low false acceptance rate, otherwise it is very easy to accumulate a lot of false evidence. Correlation filters such as the minimum average correlation energy (MACE) filters [3] usually emphasize the high frequency components of the training image, so there is less chance for a MACE filter to produce high confidence match with an imposter image So the MACE filter is suitable in our application.

The rest of this paper is organized as follows. In Sec. 2, we introduce the concept of face recognition using correlation filters and introduce the three correlation filters used in this work. In Sec. 3, a still-to-video face verification system is described and each component is introduced in detail. Our numerical experiments results are presented in Sec. 4 and final discussion is in Sec. 5.

2   Advanced Correlation Filters

The basic concept of still-to-still face verification [3] using a correlation filter is illustrated in Fig. 1. There are two stages, namely the enrollment stage and the verification stage. During the enrollment stage, one or multiple images of an individual's face are acquired. These multiple training images should reflect the expected variability (perhaps due to rotation, scale changes, illumination changes, etc.) in the face image. The 2-D Fourier transforms (FT) of these training face images are used by a correlation filter design algorithm to determine a single frequency-domain array (called a correlation filter) that is stored. In the verification stage, the user presents a live face image and its 2-D FT is multiplied by the stored frequency domain array and the inverse 2-D FT of this product results in the correlation output. If the filter is well designed, we should observe a strong peak in the correlation output when the input image is from an authentic and no such discernible peak if the input is from an impostor. The location of the peak indicates the position of the input image and thus provides automatic shift-invariance. One measure of the peak sharpness is the peak-to-sidelobe-ratio (PSR), [3] defined as follows

$$PSR = \frac{Peak - mean(sidelobe)}{std(sidelobe)},$$

(1)

where the sidelobe refers to a small region centered at, but excluding the peak.
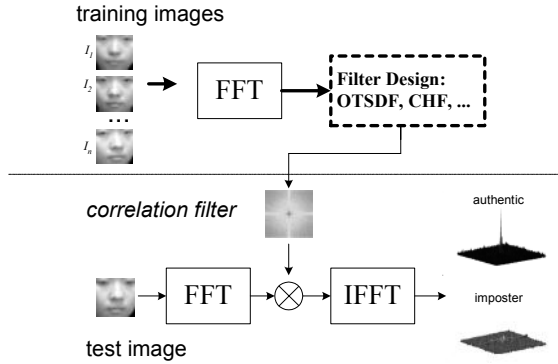
Fig 1. The concept of still-to-still face verification using correlation filter
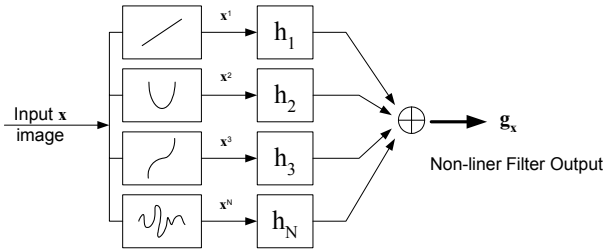
Fig 2. N-th Order polynomial correlation filter architecture [4]

Many advanced correlation filter have been developed with different criteria such as maximal noise tolerance, and maximal peak sharpness [3-6].

Optimal Tradeoff Filter (OTF). The minimum variance synthetic discriminant function (MVSDF) filter [3] was developed to maximize tolerance to input noise and the minimum average correlation energy (MACE) filter [3] was developed to suppress sidelobes in correlation output and thus produce a sharp peak. However, the MVSDF filter emphasizes low spatial frequencies whereas the MACE filter emphasizes high spatial frequencies and thus they are conflicting. Using the multi-criteria optimization approach, an optimal tradeoff filter (OTF) that optimally trades off distortion tolerance against discrimination was introduced in [5].

Optimal Tradeoff Circular Harmonic Function Filter (OTCHF) [6]. A common distortion in acquiring face images is in-plane rotation of the face image. The coordinate transformations such as polar mappings can be combined with Fourier transforms to design circular harmonic function (CHF) [6] filters that can handle in-plane rotation. Since the face images are mapped into polar coordinates, in-plane rotations appear as shifts in the polar domain and shifts can be handled by correlation filters. The CHF filter design optimally trades off among various correlation filter performance criteria while achieving specified in-plane rotation response of the correlation peak. Although polar mappings are used during filter design, filter use

during the verification stage is just like in all other correlation filter cases and no coordinate transformation is needed during the verification stage.

Polynomial Correlation Filter (PCF). The main idea underlying PCFs [4] is illustrated in Fig. 2. Although any point nonlinearity can be used for PCFs, we consider powers (e.g. x^2, x^3, etc.) for reasons of analytical simplicity. The analysis and the form of the solution remain the same irrespective of the non-linearity used, with the only restriction that the nonlinearity should be point nonlinearity. The objective is to find the filters $h_i(m,n)$ such that the structure shown in Fig. 2 optimizes a performance criterion of interest. More details of PCF filter design can be found elsewhere [4].

3   Still-to-Video Face Verification System

The proposed still-to-video face verification system has two stages: the training process based on one or more still images and the automatic face verification based on video sequences. A face template is cropped from one training image of a subject, and multiple correlation filters are then built from that template. The test system consists of three components: a face detecting and tracking block to crop the face image from each frame, a face verification block that uses multiple correlation filters to derive partial evidences from each frame, and an evidence accumulation block to summarize evidence and yield a score for final face verification decision.

Cropping face images from each frame is implemented in two steps. In step one, a region of interest (ROI) of human head is detected from each frame by using the motion information derived from difference image of two consecutive frames [7]. In step two, we use a template matching method to accurately locate the face region in the ROI. This exhaustive search is not very efficient, but still computationally affordable since the searching range has been narrowed by the ROI.

The verification decision is based on a video sequence, but at frame level we use correlation filters to derive partial evidences. OTSDF filter has been shown good verification performance for still-to-still face verification across illumination and expression variation, so it is used as a baseline filter in this paper. For each frame, three correlation filters are used to generate three PSRs. The PSR values are used as partial evidences. We use a naïve combination method that produces a weighted sum of matching scores, for which the weights are chosen by trial and error. Since different correlation filter methods focus on different performance criteria, the combination of multiple correlation filters can lead to a better decision.

$$s_i = \begin{cases} 0 & PSR_i < T_1 \\ PSR_i - T_1 & T_1 \leq PSR_i < T_2 \\ \exp(PSR_i - T_2) + (T_2 - 1 - T_1) & otherwise \end{cases}, \qquad Score = \sum_{i=1}^{N} s_i \qquad (2)$$

One property of MACE-type correlation filter is that it usually emphasizes high frequency components, and therefore there is little chance for a correlation filter to generate high PSR values with impostors, which implies that a higher PSR value indicates matching with high confidence. Based on this property, we propose an evidence accumulation method that maps each PSR value to a score by giving more

weight to the higher PSR and then sums up those scores to generate a final score for each test sequence. The mapping function is a two-threshold monotonically increasing function shown in Eq. (2).

4   Experiments

In order to test the proposed system, we collected a video sequence database with 10 subjects. During the data collection, the subject is asked to pretend to be working in front of a computer with a webcam on top of the monitor. The subject may change his/her position, expression and orientation of the head. For each subject, 300 frames are collected as a test video sequence. Whenever possible, we asked the subject to submit several high quality still images with a frontal face. Otherwise, we take their pictures using a digital camera. The cropped gallery images and some examples from the probe video sequence of these 10 subjects are shown in Fig. 3.

We tested the OTSDF filter, the OTCHF filter, the PCF and the combination of the OTCHF filter and the PCF for each sequence. The OTSDF filter is used as a baseline method for comparison. For evidence accumulation, we tested a baseline method of direct summation of all PSR values for each sequence to generate a final score. We also tested the proposed nonlinear PSR accumulation method, for which we need to choose two thresholds for the two-level-threshold mapping function. These thresholds can be chosen based on imposters' face video sequences. In our experiment, we additionally collected other 4 people's face video sequences as imposter sequences. We tested these imposter sequences for each person in the 10-subject database and get the PSR values for imposters in each case. From the mean and variance of these PSR values we compute the proper thresholds for each subject in the database.
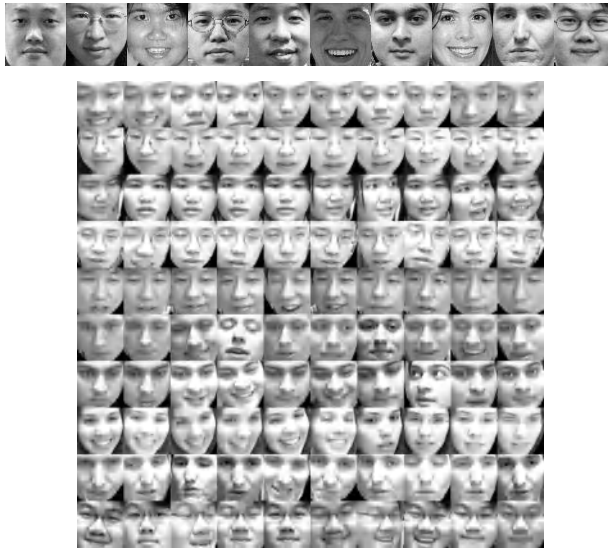
Fig. 3. The gallery set: including 10 templates manually cropped for 10 subjects; the probe set 100 sample images from the probe set: randomly chosen 10 for each subject

Table 1. Number of verification errors

	OTSDF	PCF	OTCHF	PCF+OTCHF
Direct summation	9	7	6	6
Weighted summation	6	1	6	0
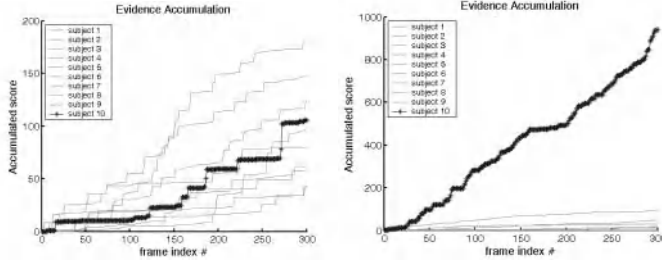
Fig. 4. The accumulated scores over frame index number of subject 10 for the OTSDF filter (left) and the PCF (right)
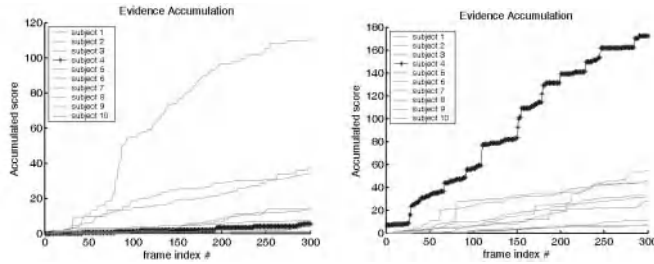
Fig. 5. The accumulated score over frame index number of subject 4, PCF filter (left) and OTCHF filter (right)

In our experiments if the sequence with the highest score has the same class label as the template, it represents a correct verification; otherwise, we record an error. The face verification performances are shown in Table 1. In each cell we show the number of errors.

We can see from the Table 1 that the weighted summation outperforms the direct summation, which means the proposed evidence combination method is able to effectively exploit video sequence information to get better verification performance. We also see that the advanced correlation filters, e.g., PCF filter and OTCHF filter, perform better than the OTSDF filter. By combining the results of different correlation filters, we get even better verification rate.

In Figs. 4 and 5, we illustrate the perforamnce of the PCF filter and OTCHF filter by means of two examples. In each plot, the thicker curve is for authentic and the lighter curves are for imposters. Fig. 4 shows evidence accumulation curves of an OTSDF filter and a fifth order PCF filter for subject-10. We can see that by using OTSDF filter there are three imposters whose accumulated verification scores are larger than the authentic, but by using PCF we essentially apply a nonlinear classifier for each frame and get the better verification performance. In our experiments, the

PCF generally outperforms its linear counterpart. In Fig. 5, we see the evidence accumulation curves for subject-4 by using the PCF and the OTCHF filter. In subject 4's testing sequence, a continuous tilt of the head is present, resulting in the in-plane rotation variation of his cropped face images. OTCHF filter is particularly designed to have in-plan rotation tolerance so it shows better performance than the PCF.

5   Conclusions

This paper introduced a still-reference-to-video-sequence face verification system. This system is useful in applications requiring the recognition of a subject in a video sequence when only one or a few still training images are available. The automatic face verification system consists of face detection and cropping, face matching and evidence accumulation procedures. Numerical experiments show that by using multiple correlation filters and nonlinear evidence accumulation method, the proposed system is able to effectively verify a video test sequence based on a still reference image. In the follow-on work, we will further investigate the multiple correlation filter combination technology and exploit temporal information in evidence accumulation methods.

Acknowledgement

References

[1]   S. Zhou, V. Krueger and R. Chellappa, "Probabilistic Recognition of human faces from video", Computer Vision - ECCV 2002, 7th European Conference on Computer Vision, Copenhagen, Denmark, Part III, pp. 681-697, 2002.

[2]   X. Liu and T. Chen, "Video-based face recognition using adaptive hidden Markov models." IEEE International Conference on Computer Vision and Pattern Recognition 2003, Madison, Wisconsin, June 2003.

[3]   B. V. K. Vijaya Kumar, M. Savvides, C. Xie, K. Venkataramani, J. Thornton and A. Mahalanobis: "Biometric verification with correlation filters". Applied Optics, Vol. 43, No.2, Jan. 2004.

[4]   B. V. K. Vijaya Kumar and A. Mahalanobis, "Recent advances in composite correlation filter designs," *Asian Journal of Physics*, Vol. 8, No. 4, pp. 407-420, 1999

[5]   Ph. Refregier, "Optimal trade-off filters for noise robustness, sharpness of the correlation peak, and Horner efficiency," Opt. Lett. 16, pp. 829-831, 1991.

[6]   B. V. K. Vijaya Kumar, A. Mahalanobis, and A. Takessian, "Optimal tradeoff circular harmonic Function correlation filter methods providing controlled in-plane rotation response". IEEE Trans. Image Processing, Vol. 9, No. 6, pp.1025-1034, 2000.

[7]   S. Palanivel, B. S. Venkatesh and B. Yegnanarayana, "Real time face authentication system using autoassocaitive neural network models". IEEE International Conference on multimedia and Expo., pp. 257-260, Baltimore, July 2003.

What Can I Tell from Your Face?

Enrico Grosso and Massimo Tistarelli

University of Sassari
Computer Vision Laboratori
Palazzo del Pousalid, piazza Duomo 6, 07041 Alghero (SS) – Italy
{grosso,tista}@uniss.it

Abstract. Among the many developed techniques for biometric recognition, face analysis seems to be the most promising and interesting modality. Nonetheless, there are still many open problems which need to be "faced" as well. For example, the information conveyed by faces (in fact too much) must be selectively extracted to obtain useful hints for recognition. The choice of optimal resolution of the face within the image, face registration and facial feature extraction are still open issues. This not only requires to devise new algorithms but to determine the real potential and limitations of existing techniques. It turns out that all methods based on the same biometric measurements have the same intrinsic limitations, which can be only overcome by the adoption of a multi-modal or multi-algorithmic approach.

1 Introduction

Every living creature depends on the ability to perceive the outside world. In turn, the correct perception requires to consistently organize the acquired sensory data, either visual, tactile, olfactory or acoustic. Organized "perceptual patterns" allow humans to perform a variety of tasks, all based on the recognition of precise data configurations. For example, the first and primary perceptual task of a living creature is the recognition of his/her mother. This is firstly based on simple recording and matching of olfactory data, but quickly develops adding information based on her voice and appearance. In the case of humans, as the child grows, quickly develops a surprising ability to analyze faces. The neural plasticity of infants allows to generalize the "mom recognition" task to learn how to recognize many different faces. The age, race, mood and other behavioral attitudes are perceived from the face image as well.

Reverting this human capability into the design of an information technology system is certainly a formidable task. Current research in face and gesture recognition is trying to "climb the hill" toward this direction [1-10]. Many difficulties arise from the enormous dimensionality of the search space when dealing with natural images. These findings enforce the need to devise simple and modular processing elements, which are functionally related to the selective extraction of collective information from face image streams. This paper presents few of these modules aimed at the authentication of a person's identity by matching holistic representations of the face.

2   Information Processing

Neural systems that mediate face recognition appear to exist very early in life. In normal infancy, the face holds particular significance and provides nonverbal information important for communication and survival [11]. Much is known about the neural systems that subserve face recognition in adult humans and primates [12].

The high specialization of specific brain areas for face analysis and recognition motivates the relevance of faces for social relations. On the other hand, this suggests that face understanding is not a low level process but involves higher level functional areas in the brain. These, in turn, must rely on a rich series of low level processes applied to enhance and extract face- specific features:

- Face detection and tracking. This process may involve the analysis of dynamic as well as geometric and photometric features on the face image.
- Facial features extraction. Facial features are not simply distinctive points on the face, but rather a collection of image features representing specific (and anatomically stable) areas such as the eyes, eyebrows, mouth, etc. Other, subject-specific features may be also included, such as a naevus.
- Face image registration and warping. Humans can easily recognize faces rotated and distorted up to a limited extent. The increase in time due to face rotation and distortion implies: the expectation of the geometric arrangement of features, and a specific process to organize the features (analogous to image registration and warping) before the recognition process can take place.
- Feature matching. This process involves the comparison between the extracted set of facial features and the same set stored in the brain. Feature extraction and matching (or memory recall) are not completely separated and sequential processes. From the eye scan paths recorded during face recognition experiments, it seems that, after moving the eyes over few general facial features, the gaze is directed toward subject-specific features, probably to enforce the expected identity.

Together with the defined general visual processes, the understanding of more abstract terms (such as age, race, gender, emotion) also requires the intervention of task-specific processes, such as motion analysis and facial features tracking for understanding emotion-specific patterns [3, 5, 6, 13].

As it is beyond the scope of this paper to trace all face-specific information processing, we will concentrate on face recognition and authentication, which not only are among the most studied aspects related to visual processing, but probably the most representative of the tasks involved in face image analysis.

3   Face Recognition

To achieve any visual task, including face recognition, humans are able to purposively control the flow of input data limiting the amount of information gathered from the sensory system [14-16]. The anatomy of the early stages of the human visual system is a clear example: despite the formidable acuity in the fovea

centralis (1 minute of arc) and the wide field of view (about 140x200 degrees of solid angle), the optic nerve is composed of only 1 million nerve fibers. The space-variant distribution of the cells in the retina allows a formidable data flow reduction [17].

For this reason face recognition requires the selection of few relevant attention areas which are the most informative within the face.

3.1 Extraction of Facial Features

A technique applied to detect the facial features relies on the application of morphological operators to enhance structured iconic data such as contours, valleys and peaks in the gray levels. This information is gathered to make hypotheses for the presence of specific facial features, into two steps:

- By first computing the cumulative values of the filtered image along the rows. The eyes correspond to the area with higher cumulative values;
- The same process is performed along the columns in the area corresponding to the eyes, determined at the previous step. The two maxima locate the two eyes.

In order to avoid false matches a geometrical constraint is enforced to the position of the eyes and mouth, which is to lie at the vertexes of a triangle. The values assumed by the angles of the triangle are bounded to $44° < \alpha_i < 84°$.

The exact position of the features is determined by computing the cross-correlation between the image and a feature template, within a 10x10 pixels window centered on the previously determined position. The template is obtained by cutting the eyes and mouth out of a sample image of the same subject. From an extensive test this choice demonstrated to give more accurate results than computing an average template. This is due to the fact that the averaging process deforms considerably the feature's shape.
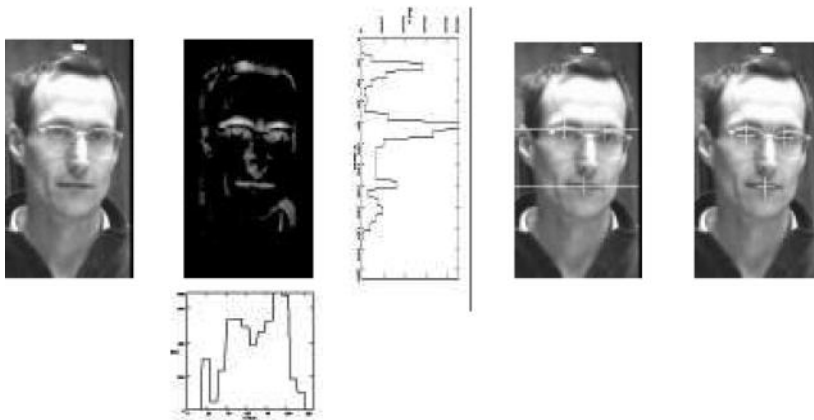
Fig. 1. Facial features detection. (Left) Original image. and valley image after morphological filtering. (Center) Cumulative values computed along the rows of the filtered image and along the columns of the window identified by the maxima extracted from the filtered image. (Right) First guess for the eyes and mouth position and final position of the facial features computed by template matching

Fig. 2. Extraction of space-variant fixations. (Left) Original image. (Right) Log-polar fixations

The three correlation values stemming from the eyes and mouth are averaged to obtain a score between −1 and 1. The estimated position is accepted only if the geometric constraint is satisfied <u>and</u> the matching score is higher than a given threshold. The discriminant value for the correlation score is determined by a validation test performed on a set of 2019 images, divided into two classes:

- all images (1609) where the facial features are partially occluded or not visible, plus all the images where the mean difference between the estimated and the manually determined feature positions is greater than a given threshold[1];
- all remaining images in the set (410).

The False Acceptance and False Rejection Rate were computed from the feature correlation scores of the two image sets. These measures represent the capability of separating the two classes (valid and wrong features). The score value corresponding to equal FAR and FRR determines the reliability of the estimated features.

3.2 Analysis of Matching Techniques

Two matching techniques are presented. In the former, the subject is represented by a collection of fixations from the face image. The matching is performed by computing the correlation between the representation of the reference subject and the acquired face image. The algorithm is based on the following steps:

1. Given the position of selected facial features (the eyes and the mouth), three log-polar fixations are extracted from the acquired image of the subject (see Fig. 2.).
2. The log-polar images are warped to simulate views as close as possible to the pose and orientation of the reference subject's face (almost parallel to the image plane).

3. Corresponding fixations are compared by computing the sum of the absolute value of gray level differences[2] and the normalized correlation. Two matching scores are obtained from each fixation independently.
4. The scores obtained by the log-polar fixations are combined to form a 6 components vector representing the similarity between the subject and the model.

---

[1] This threshold is determined statistically by computing the probability of locating the facial features correctly in more than 50% in a given image ensemble.

[2] To compute the difference, the gray levels of each log-polar fixation are first normalized to the range of intensity values of the corresponding facial feature of the reference subject.
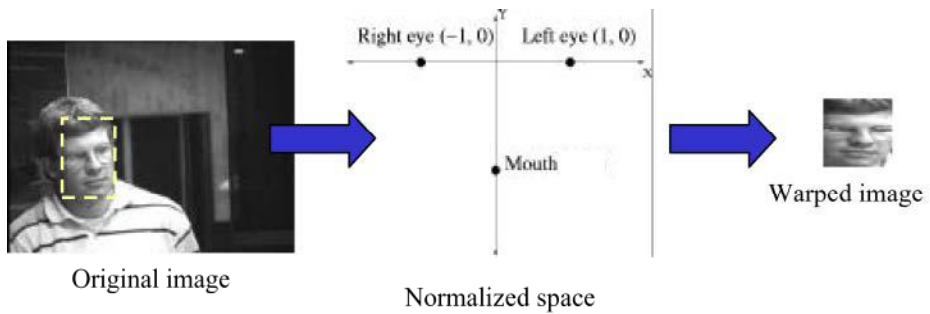
Fig. 3. Extraction of the face window and warping

The log-polar transformation is computed at frame rate by using special re-mapping software routines. This approach has several advantages over special hardware devices, like space-variant sensors: it is more flexible and allows the use of conventional, low-cost cameras. The feature extraction and matching process may be iterated over other subject-independent (like the eyebrows) or subject-dependent features, thus increasing the reliability of the recognition process.

The latter technique performs the matching over a single window containing the whole face [18]. As a major problem with template matching is the registration of the two images, the window is warped according to a feature space determined by the position of the facial features (see figure 3).

4   Evaluation of Matching Techniques

Every face matching scheme is devised to deliver a similarity score from each compared image-template pair. From the computed scores a statistical classifier allows to determine the similarity between a given model and the subject. The real potential of a matching method can be assessed by dividing the matching scores into two classes:

- scores obtained comparing all images of the same subject (*client tests*);
- scores obtained comparing all images of different subjects (*impostor tests*).

Given the entire ensemble of matching scores for the two classes (each score can be regarded as a vector within the class), the discrimination power can be defined through five statistical indexes:

- The intraset and interset distances (class separability indexes). These measurements define the distances among the elements of the same class and between the two classes. The intraset distance should be smaller than the interset distance.
- The Bayesian error probability.
- The false acceptance, false rejection and the equal error rate (FAR, FRR, EER). A one-dimensional representation of the distributions of the measurement vectors (set of matching scores) of the two classes, is determined by the Fisher transform:

$$w = S_w^{-1}(m_1 - m_2) \;\; ; \;\; S_w = S_1 + S_2 \;\; ; \;\; S_i = \sum_{j=1}^{Ni} \left(x_j^i - m_i\right)\!\left(x_j^i - m_i\right)^t \qquad (1)$$

N_i represents the number of elements of class "i" and m_i is the mean value.

The resulting curves represent the probability densities of the missed clients and the allowed impostors as a function of the matching score, and the integrals represent the false acceptance and false rejection rates [18, 19].

The described algorithms were tested on a subset (with, at least, two images for each subject) of the FERET database [6] yielding 2.5 EER for both methods [10]. The same tests were also performed on the Olivetti Resarch Lab database [20], using the five even images, out of the set of ten per subject, to simulate training, and the other five for testing. The matching result for each trial has been obtained using a max rule to combine the similarity scores, yielding 8.5% EER (performing full face matching).

To test the algorithms in the worst conditions, a more challenging database has been acquired, composed of 488 gray level images (8 bits per pixel) from 75 subjects. The image size is 384x288 pixels with the head size smaller than 80x120 pixels. The images have been acquired from a camera placed at the entrance of an industry working area. As the subjects were unaware, the faces have unpredictable size, expression, orientation and pose. Most of the pictures from the same subject were acquired at least days or even weeks apart. As it can be noticed, the log-polar matching yields a higher error than the full face matching technique. In fact, the low image resolution does not allow to capture sufficient gray level variations within the receptive fields.

Table 1. Performances of the face matching systems. The two columns on the left report the computed errors with the two described methods for the extraction of the facial features. As a reference, the last column reports the performances of a highly reputed commercial system.

	Full face matching		Log-polar matching	Commercial system based on LFA
	Subject-based template	Generic template	Subject-based template	
P_e	30.5%	36.5%	32%	31.5%
EER	16%	21.5%	17%	20%

5   Conclusion

The analysis of faces is one of the most important visual processes in humans. Two iconic matching algorithms have been considered and stressed to the worst possible working conditions by using an ad-hoc database. The aim was to understand the actual limits and failing modes not only of the specific method applied, but also of any general iconic-based matching technique.

From the error indices reported in this, as well as previous papers [18], it is possible to make the following considerations:

- The face registration and warping is critical for the entire recognition process.
- The performances of an identity verification system must be assessed on multiple and different data sets, to understand the real failing modes.
- There is an intrinsic limit in using a matching technique alone to discriminate a set of given clients from the class of all possible impostors.

The clear limitations of a single matching engine enforces the need for either a multi-level or a multi-algorithmic process, where several (at least two) cooperating "experts" are applied to the same authentication process.

References

[1]   Chellappa, R., Wilson, C.L., Sirohey, S.: Human and machine recognition of faces: A survey. Proceedings of the IEEE, Vol. 83 (1995) 705-740
[2]   Sinha, P., Poggio, T.: I think I know that face… *Nature*, Vol. 384 (1996) 404
[3]   Wiskott, L., Fellous, J. M., Kruger, N., von der Malsburg, C.: Face recognition and gender determination. In Proceedings Int.l Workshop on Automatic Face and Gesture Recognition, Zurich, Switzerland (1995) 92-97
[4]   Jain, A. K., Bolle, R., Pankanti, S.: Biometrics, Personal Identification in Networked Society. Kluwer Academic Publishers (1999)
[5]   Wechsler, H., Phillips, J. P., Bruce, V., Soulie, F., Huang, T. (eds.): Face Recognition. From Theory to Applications. NATO ASI Series F, Vol. 163. Springer-Verlag (1998)
[6]   Cottrell, G., Metcalfe, J.: Face, gender and emotion recognition using holons. In D. Touretzky (ed.) Advances in Neural Information Processing Systems, Vol. 3. Morgan Kaufmann, San Mateo, CA (1991) 564-571
[7]   Belhumeur, P. N., Hespanha, J. P., Kriegman, D. J.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Trans. on PAMI, Vol. 19, no. 7 (1997) 711-20
[8]   Edwards, G. J., Cootes, T. F., Taylor, C. J.: Face Recognition Using Active Appearance Models. In Proceedings of 5th ECCV, Springer Verlag (1998) 581-95
[9]   Liu, C., Wechsler, H.: Evolutionary pursuit and its application to face recognition. IEEE Trans. on PAMI, Vol. 22, no. 6 (2000) 570-582
[10]   Tistarelli, M., Grosso, E.: Active vision-based face authentication. Image and Vision Computing: Special issue on FIA, M. Tistarelli (ed.), Vol. 18, no. 4, (2000) 299-314
[11]   Darwin, C.: The expression of the emotions in man and animals. John Murray, London, U.K. (1965). (Original work published in 1872)
[12]   Goren, C., Sarty, M., Wu, P.: Visual following and pattern discrimination of face-like stimuli by newborn infants. Pediatrics*, Vol. 56 (1975) 544–549
[13]   Picard, R. W.: Toward computers that recognize and respond to user emotion. IBM System, Vol. 39, no. 3/4 (2000)
[14]   Ballard, D. H.: Animate vision. Artificial Intelligence, Vol. 48 (1991) 57-86
[15]   Aloimonos, Y. (ed.): Purposive, qualitative, active vision. CVGIP: Image Understanding, Vol. 56 (special issue on qualitative, active vision) (1992)
[16]   Tistarelli, M.: Active/space-variant object recognition. Image and Vision Computing, Vol. 13, no. 3 (1995) 215-226
[17]   Schwartz, E. L., Greve, D. N., Bonmassar, G.: Space-variant active vision: definition, overview and examples. Neural Networks, Vol. 8, no. 7/8 (1995) 1297-1308
[18]   Tistarelli, M., Lagorio, A., Grosso, E.: Understanding Iconic Image-Based Face Biometrics. In Proceedings of Int.l Workshop on Biometric Authentication 2002, Copenaghen, Denmark. LNCS, Vol. 2359. Springer Verlag (2002) 19-29

[19]   Grother P. J.: Cross validation comparison of NIST ocr databases. In Proceedings of the SPIE, Vol. 1906 (1993) 296-307

[20]   Samaria, F., Harter, A.: Parameterisation of a Stochastic Model for Human Face Identification. In Proceedings of 2nd IEEE Workshop on Applications of Computer Vision, Sarasota, FL. IEEE Computer Society Press (1994)

An Integrated Dual Factor Authenticator
Based on the Face Data and Tokenised Random Number

Andrew B. J. Teoh[1], David C. L. Ngo[1], and Alwyn Goh[2]

[1] Faculty of Information Science and Technology (FIST), Multimedia University
Jalan Ayer Keroh Lama, Bukit Beruang, 75450, Melaka, Malaysia
{bjteoh,david.ngo}@mmu.edu.my
[2] Corentix Laboratories,
32 Jalan Tempua 5, 47100 Puchong, Malaysia
alwyn_goh@yahoo.com.uk

Abstract. This paper proposed a novel integrated dual factor authenticator based on iterated inner products between tokenised pseudo random number and the user specific facial feature, which generated from a well known subspace feature extraction technique- Fisher Discriminant Analysis, and hence produce a set of user specific compact code that coined as BioCode. The BioCode highly tolerant of data captures offsets, with same user facial data resulting in highly correlated bitstrings. Moreover, there is no deterministic way to get the user specific code without having both tokenised random data and user facial feature. This would protect us for instance against biometric fabrication by changing the user specific credential, is as simple as changing the token containing the random data. This approach has significant functional advantages over solely biometrics ie. zero EER point and clean separation of the genuine and imposter populations, thereby allowing elimination of FARs without suffering from increased occurrence of FRRs.

1   Introduction

Many systems require reliable personal authentication infrastructure to recognize the identity of a claimant before granting access to him/her. The conventional secure measure involves possession of token or special knowledge like smart card and password. Unfortunately, the first two suffer a lack of security as they are easy being forgotten and stolen. Biometric surpasses these two methods by offering positive human identifier based on an intrinsic aspect of a human-being. However, the greatest vulnerability of biometric is impuissance to compromisation [1] – a major concern of the public for the privacy risk in biometric technology. In other words, when a biometric is compromised, like a password, is rendered unusable. This is exacerbated by the fact that when compromised, a new template cannot be assigned.

There is substantial research going on to find solutions/alternatives to the contemporary non-reissuable biometric. Some authors like Ratha et. al. [1] and Davida et al. [2] have introduced the terms *cancelable biometrics* and *private biometrics*. These terms are used to denote biometric data that can be cancelled and

replaced, as well as is unique to every application. Davide, et al. [3] have outlined three principal objectives of designing a cancelable/private biometrics:

i.   No same cancelable template can be used in two different applications.
ii. A new template can be reissued when compromised.
iii. Only combination of the hashed template and the biometric feature can contribute to the recognition process.

In this paper, a novel integrated dual factor authentication approach which combined tokenized random data with face feature to generate a unique compact binary code per person - BioCode is highlighted. The discretization is carried out by iterated inner product between the pseudo random number and face features that generated from the Fisher Discrimination Analysis (FDA)[4], and finally deciding each bit on the sign based on the predefined threshold. Direct mixing of random and biometric data is, in fact, an extremely convenient mechanism with which to incorporate serialised physical tokens, i.e smart card or USB token, thereby resulting in two factors (token+biometrics) credentials via tokenised randomisation. Hence, it protects against biometric fabrication without possession of the corresponding token. Tokenised discretisation also enables straightforward revocation via token replacement. Therefore BioCode are conformed to the above listed criteria. Biometric decision scheme, which is well defined under classical framework of statistical decision theory, is extremely hard to gain zero both in False Acceptance Rate (FAR) and False Reject Rate (FRR), due to the fact that the classes are difficult to completely separate in the measurement space [5]. Realization of relatively low FAR, i.e. acceptance of imposters, will yield relatively high FRR, i.e. rejection of genuine and otherwise. BioCode has significant functional advantages over solely biometrics ie. zero EER point and eliminate the occurrence of FAR without overly imperil the FRR performance.

2   BioCode Formulation Overview

BioCode formulation consists of two stages: (a) transformation of a raw face image to a highly discriminative representative feature – an user specific fisherprojections on Fisher subspace, with a moderate degree of offset tolerance; (b) a discretisation of the data via an iterated inner-product of tokenized random and user data. Note that inner-products are straightforwardly specified for more complex representations ie $s=\int dx \int dx'.a(x')b*(x-x')$ for square-integrable functions a, b$\in$ L^2. The details of the feature extraction and biometric discretisation components will be discussed as follow.

2.1 Feature Extraction – Fisher Discrimination Analysis

Fisher Discriminant Analysis (FDA) is a popularly known technique to maximize the ratio of between-class scatter to that of within-class scatter. In other words, it projects face image $I \in \mathbb{R}^N$ such that images of the same class are close to each other while images of different classes are far apart into Fisher subspace with $\mathbb{R}^{c-1}$ fisherprojections, with dimensionality $c<<N$ where c corresponding to the number of distinct users in the training images. The basis vectors calculated by FDA create the

Fisher subspace, which is also called FisherFace. More detail information can be obtained from [4].

2.2 Biometrics Discretisation

At this stage, the fisherprojections, $\omega \in \mathbb{R}^{c-1}$ is reducing down to a set of single bit, $b \in \{0,1\}^m$, with m the length of the bit string via a tokenised pseudo random number, $r \in \mathbb{R}^m$, which distributed according to uniform distribution U[-1 1].

BioCoding is described in terms of successive simplifications as follow:

(a) Raw intensity image representation: $I \in \mathbb{R}^N$, with N the image dimension.
(b) FDA representation in a vector format, $\omega \in \mathbb{R}^{c-1}$, with c corresponding to the number of distinct user in the training image.
(c) Discretization, $b \in \{0,1\}^m$, where $m \leq c-1$.

The BioCoding progression can be illustrated as in Fig. 1:
The primary concern from the security viewpoint eyes on information protection during the representational transformations, and in particular whether (or how) these transformations can be inverted to recover the input information, i.e. biometric fabrication - one of the major public concerns in biometrics application. The above-listed parameters are said to be zero knowledge representations of their inputs if the transformations are non-invertible, as in the case of cryptographic hash $h(r,k) : 2^m \times \underset{m'}{\forall} \, 2^{m'} \to 2^m$ for token serialisation r and secret knowledge (arbitrary-length password) k. Note the non-recovery of key-factors $\langle r, k \rangle$ from h(r, k), which motivates an equivalent level of protection for biometric feature. This is accomplished via token-specification of representations (c), such that $H(r,\omega) : 2^m \times \mathbb{R}^{c-1} \to 2^m$

Achieving (c) requires an offset-tolerant transformation by projected ω onto each random pattern, and the choice of a threshold, τ to assign a single bit for each projection:

1. Compute x = $\langle \omega, r \rangle$ with r, where $\langle \cdot, \cdot \rangle$ denote the inner product process.

2. Assign b(x) = $\begin{cases} 0 & x < \mu - \sigma \\ 1 & x > \mu + \sigma \\ \varnothing & x \in [\mu \pm \sigma] \end{cases}$

for experimental parameter μ and σ, the former of which should theoretically vanish due to above specification of ω relative to the population average. Note the specification of ω weights in step (1), hence placing relatively greater importance on the more principle fisherprojections. Extracted b is a broad measure of whether $\langle \omega, r \rangle$ are inline or opposed, with σ applied to exclude the perpendicular case. This exclusion relieves against data capture uncertainties in ω, which might otherwise result in bit-inversion for numerically small x. The relative geometries of ω and r are illustrated in Fig. 2.
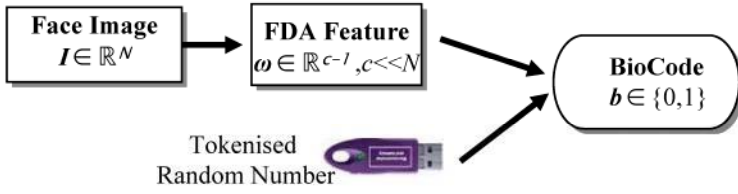
Fig. 1. BioCoding Progression

Repetition of this procedure to obtain multiple bits render the issue of inter-bit correlations, which is addressed via orthonormal set $\varsigma = \{r_k: k = 1, 2, \ldots, m\}$ with $m \leq c\text{-}1$. Each bit $b(x)$ is hence rendered independent of all others. Indeterminate bits $b = \emptyset$ is handled via replacement of near-perpendicular r_k with alternative r'_k, the net effect of which is bit-extraction via adjusted set $\varsigma - \underset{k\in\perp}{\forall} r_k + \underset{k\in\perp}{\forall} r'_k$. This reformulation is facilitated by the original stipulation on m, which allows up to $(c\text{-}1)-m$ replacements for inappropriate r_k..

The proposed discretisation via repeated inner-products then proceeds as follows:

1. Generate and orthonormalise $\varsigma + \forall r'$ via Gram-Schmidt procedure for
   $k = 1\ldots m\ldots(c\text{-}1)$
2. For each $k = 1\ldots m$
    1. Compute $x_k = \omega.r_k$
    2. While $x \in [\mu\text{-}\sigma, \mu+\sigma]$:
        1. Get next unused r'
        2. Reassign $r_k = r'$ in ς
        3. Recompute x_k
    3. Assign $b_k = b(x)$
3. Concatenate $\alpha = \underset{k}{\forall} b_k$
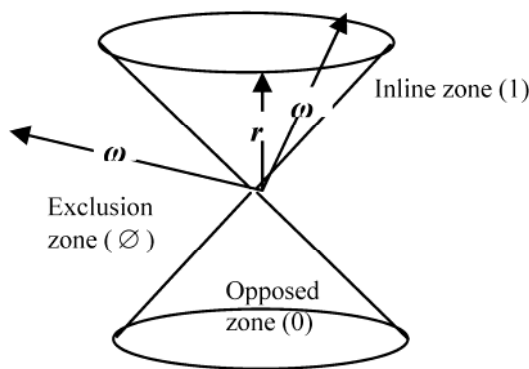
Fig. 2. Biometric fisherprojections placement in inline, opposed or exclusion zones

3   Experiments and Discussion

The method has been evaluated in terms of their receiver operating characteristic (ROC), the equal error rate (EER) and the Genuine/Imposter populations distribution achieved in FERET Face Database which available at URL http://www.itl.nist.gov/iad/humanid/feret. FERET contains over 1100 faces; however many of them are inappropriate for our experiments since they are partial or full profiles, or the individuals were only photographed twice Therefore, we selected 275 images that consist of 40 individuals, with major variation in pose and lighting. Many of these images were taken over different days and display significant differences in hairstyles, eyewear, and illumination. Three randomly selected images for each individual in the dataset were placed in the training set, and the remaining images were used for testing. Ten runs for each of three randomly selected images were performed with different, random partitions between training and testing images, and the results were averaged. As the adopted face image databases contain multiple variations, we first perform geometrical alignment and lighting adjustment before feature extraction.

Following are the abbreviations used for brevity in this paper:

- *fda*: denoting Fisher Discrimination Analysis with $\mathbb{R}^{c-1}$ where $c-1=39$
- *fda+d-m*: denoting 2^m discretisation without exclusion of weak inner-products, i.e threshold value, $\mu=\sigma=0$ where m ($\leq c-1$, $c=40$) is the bit length
- *fda+de-m*: denoting 2^m discretisation with σ error-correction based on analysis of inner-products computed from random and user-specific fisherprojections with exclusion parameter σ amounting to selection of the m most significant inner-products from r of size $m'>m$ ($m'\leq 39$)

The experimental data is acquired for $m = 10, 20, 30$ and 39 for *fda+d-m* as we cannot exceed the allowed number of bits (which equals to the feature length 39) in the used face database while $m = 10, 20$ and 30 for *fda+de-m* due to the later requires additional ω as well as r to exclude the 'weak' inner products. For the similarity matching, a simple Euclidean distance metric is adopted for *fda* whereas Hamming distance is used in *fda+d-m* and *fda+de-m*.

From Table 1, EER=0% of *fda+d-m* and *fda+de-m* compared to *fda* reveal the robustness of *fda+d-m* and *fda+de-m* in the verification task. All the ROC curves for *fda+d-m* and *fda+de-m* are extremely bowed, reaching as far as the lower left corner of Fig 3. Reaching that limit corresponds to achieving a correct accept rate of 100%, thereby the proposed methodology is efficient to overcome the FAR–FRR interdependency problem whereas using *fda* alone yield intolerable high FRR - 58.97% when FAR=0%. Since the verification rates are very high for $m= 20, 30$ and 39 (for *fda+d-m* and *fda+de-m*), another performance indicator is through the observation of range of threshold values, $t \in [0\ 1]$ when EER=0%: the bigger range of threshold value yield the better performance, as a large range of operating points (threshold) with zero errors can be obtained. Table 1 shows the range of threshold values that result in a zero error, for *fda+d-m* as well as *fda+de-m*. It can be observed that the range is getting wider when m grows, which implies system performance is boost for $m=30$ and $m=39$ for *fda+d-m* and *fda+de-m* respectively. When the

comparisons were make in between *fda+d-m* and *fda+de-m* at *m*=30, the threshold range of *fda+de-m*, 0.160 is much wider than the former, 0.074. This indicates that *fda+de-m* showed improved performance over *fda+d-m*.

Table 1. Performance Evaluation in terms of EER, FRR when FAR=0% and Threshold range when EER=0%

Setting	EER(%)	FRR(%) (FAR=0%)	Threshold Range when EER=0%
Fda	6.35	58.97	-
fda+d-10	1.44	15.39	-
fda+d-20	0.98	4.17	-
fda+d-30	0.00	0.00	0.063
fda+d-39	0.00	0.00	0.074
fda+de-10	0.11	2.93	-
fda+de-20	0.00	0.00	0.110
fda+de-30	0.00	0.00	0.160

Fig. 4. illustrated the genuine and imposter population distribution for *fda* and *fda+de-m*. The small overlapping in between genuine and imposter populations reveal that *fda* is superior to discriminate the genuine and imposter, thus make it favor in the classification task. However, genuine distribution of *fda+de-m* which about 1% of the bits differ and the disagreeing bits are very tightly packed around 50% in imposter population depicted in Fig. 4(b), indicate *fda+de-m* outweighs *fda*. This sharp drop-off is clearly seen and thus allow for specification of zero FAR without jeopardizing the FRR performance.
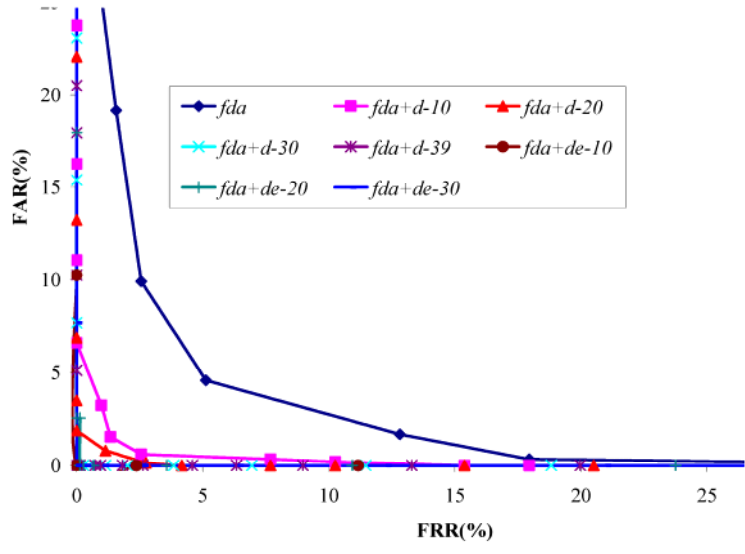
Fig. 3. ROC for fda, fda+d-m and fda+de-m
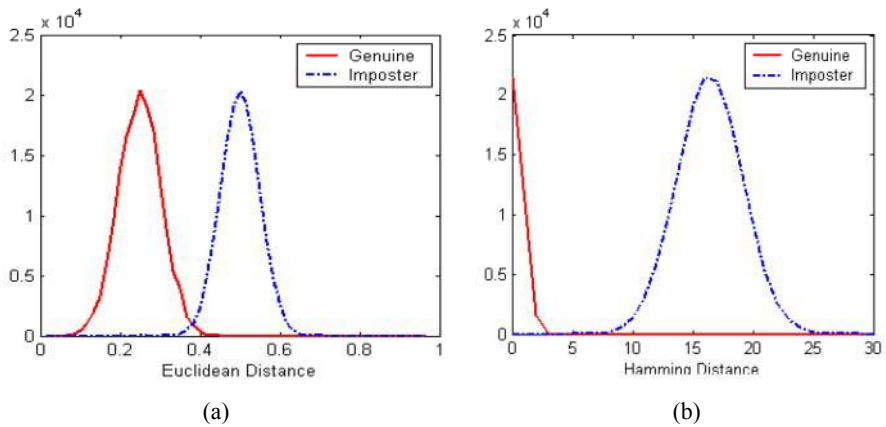
Fig. 4. ROC for (a) *fda* and (b) *fda+de-m*

4   Concluding Remarks

This paper described a novel error-tolerant discretisation methodology from user-specific face images and uniquely serialized tokens. The dual factor BioCoding has significant functional advantages over solely biometrics or token usage, such as extremely clear separation of the genuine and the imposter populations and zero EER level, thereby mitigate the suffering from increased occurrence of FRR when eliminate the FAR. The process of generating a token of pseudo-random vectors taking place only once for an individual, it can be considered secure in the sense that there is no way to recover the face data by getting hold on the token (one-way transformation). In addition, BioCoding technique also addressed the invasion of privacy issue, such as biometric fabrication. It could be alleviated through the user specific credential revocation via token replacement.

For the future work, we look forward to presenting a more comprehensive analysis with more challenging and bigger face database.

References

[1]   R. M. Bolle, J. H. Connel and N. K. Ratha. 2002. Biometric Perils and Patches. Pattern Recognition, 35, pp2727-2738.
[2]   Davida, G., Frankel, Y., & Matt, B. J. 1998. On enabling secure applications through off-line biometric identification. Proceeding Symposium on Privacy and Security. pp. 148-157.
[3]   Davide, M., Dario, M., Anil, K. J., & Salil, P. 2003. Handbook of fingerprint recognition New York: Springer. pp. 301-307.
[4]   Belhumeur, P. N., Hespanha, J. P. and Kriegman, D. J. 1997. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. IEEE Transactions On Pattern Analysis And Machine Intelligence, 19(7), pp 711-720.
[5]   John Daugman. 2002. "Biometric Decision Landscapes", Technical Report, No.482, Cambridge University Computer Laboratory.

A New Enhanced Nearest Feature Space (ENFS) Classifier for Gabor Wavelets Features-Based Face Recognition

Jianke Zhu, Mang I Vai, and Peng Un Mak

University of Macau
PO Box 3001, Macau, PR. China
fstmiv@umac.mo,
http://bbss.eee.umac.mo

Abstract. This paper proposes a new Enhanced Nearest Feature Space Classifier (ENFS) which inherits the generalization capability from Nearest Feature Space method. Additionally, estimated variance can optimize the class seperability in the sense of Bayes error, and has improve the classification power in reduced PCA subspace. Gabor wavelets representation of face images is an effective approach for both facial action recognition and face identification. Perform PCA dimensionality reduction on the downsampled Gabor Wavelets features can be effectively for face recognition. In our experiments, ENFS with proposed Gabor Wavelets Features shows very good performance, which can achieve 98.5% maximum correct recognition rate on ORL data set without any preprocessing step.

1 Introduction

Many techniques for face recognition have been developed with principles span across several disciplines [1]. Currently, by far the most popular dimensionality reduction technique in face recognition is to use subspace projections based on the Principal Component Analysis (PCA). PCA based dimensionality reduction for the face images was first proposed by Sirovich and Kirby [2]. Later, Turk and Pentland [3] demonstrated that this subspace representation could be used to implement a very efficient and successful face recognition system. The classifier also affects the recognition performance considerably. The simple nearest neighbor (NN) pattern classifier is widely employed for face recognition task. Stan Li et al. [4] proposed a kind of Nearest Feature Line (NFL) classifier which has shown to achieve lower classification error than the NN classifier. Jen-Tzung Chien et al. [5] have proposed Nearest Feature Plane (NFP) and Nearest Feature Space (NFS) classifier for robust decision in presence of wide facial variations, which outperformed the NFL. In [6, 7], C. Liu et al. proposed Probabilistic Reasoning Models (PRM) which combine the PCA technique and the Bayes classifier and show their good discriminant ability on the face recognition problem.

The whole system for our approach can be divided into three parts: features extraction (Section 2), dimension reduction & discriminant analysis (Section 3),

and classification (Section 4). Such scheme is widely used in subspace based methods. In this paper, Gabor Wavelets features are extracted by downsampled Gabor wavelets transform on the face images. The Gabor Wavelets features after performing PCA are employed for face identification. In [5], Jen-Tzung Chien only argued L_1 norm, but other similarity measure methods such as L_2 norm and angle distance are also very useful for pattern classification. We propose a new classifier for face recognition, namely Enhanced NFS method(ENFS). Our proposed method of Gabor Wavelets features using Enhanced NFS classifier achieves 98.5% correct face recognition accuracy for ORL data set [8].

2   Gabor Wavelets Transform Features Extraction

The Gabor wavelets transform provides an effective way to analyze images and has been elaborated as a frame for understanding the orientation and spatial frequency selective properties of simple cortical neurons [9]. They seem to be a good approximation to the sensitivity profiles of neurons found in visual cortex of higher vertebrates. The important advantages are infinite smoothness and exponential decay in frequency.

Let $\boldsymbol{I}(z)$ be the gray level distribution of an image, Gabor wavelets transform on $\boldsymbol{I}(z)$ can be written as a convolution of $\boldsymbol{I}(z)$ with a family of kernels ψ_k:

$$\boldsymbol{O}_k(z) = \boldsymbol{I}(z) * \psi_k(z) \tag{1}$$

where $*$ denotes the convolution operator, and $\boldsymbol{O}_k(z)$ is the convolution result at k. The Gabor wavelets (kernels) take the form of a plane wave restricted by a Gaussian envelope function [9]:

$$\psi_k(z) = \frac{\parallel k \parallel^2}{\sigma^2} e^{-\parallel k \parallel^2 \parallel z \parallel^2 / 2\sigma^2} [e^{ikz} - e^{-\sigma^2/2}] \tag{2}$$

where k determines the wavelength and orientation of the kernel $\psi_k(z)$ in image coordinates. The first term in bracket is oscillation part, and the second is dc part. The k is defined as $k(\mu, v) = k_v e^{i\phi_\mu}$, where μ and v define the orientation and scale of the Gabor kernels, $k_v = k_{max}/f^v$ and $\phi_\mu = \pi\mu/8$. k_{max} is the maximum frequency, and f is the spacing factor between kernels in the frequency domain.

The $\psi_k(z)$ forms a family that is self-similar under the application of the group of translations, rotations, and resizes. The effect of the dc term becomes negligible when the parameter σ, which determines the ratio of the Gaussian window width to wavelength, has sufficiently large value.

In [9], Lades et al. suggest that good result can be obtained by using Gabor wavelets of five different scales, $v \in \{0, \ldots, 4\}$, and eight orientations, $\mu \in \{0, \ldots, 7\}$, $\sigma = 2\pi$,$k_{max} = \pi/2$ and $f = \sqrt{2}$. The complex value combines both even (cosine-type) and odd (sine-type) parts. Fig.1 shows the even part. Fig.2 shows the Gabor wavelet transform on a sample image (128×128) at 5 scales and 8 orientations (the magnitude part). These representation results display scale, locality, and orientation properties corresponding to those
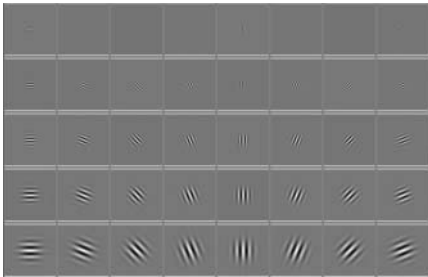
Fig. 1. Even part of Gabor wavelets kernels
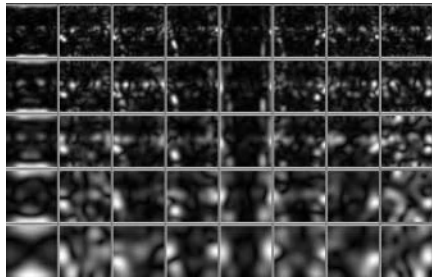
Fig. 2. A sample face image represented by the magnitude part of Gabor wavelets transform

displayed in Fig.1. The downsampled Gabor wavelet transform results by the factor of 64 formed a feature vector $\boldsymbol{g} \in \mathbb{R}^n$, where $n = 10240$.

3   Principle Component Analysis

The basic goal in PCA is to reduce the dimension of the data. The representation given by PCA is an optimal linear dimension reduction technique in the mean-square sense, and noise may be reduced. Consider a set of L feature vectors $\{\boldsymbol{g}_1, \boldsymbol{g}_2, \ldots, \boldsymbol{g}_L\}$ taking values in an n dimensional feature space. $\Sigma_{\boldsymbol{g}} \in \mathbb{R}^{n \times n}$ is defined as the covariance matrix of the augmented feature vector $\boldsymbol{g}$:

$$\Sigma_{\boldsymbol{g}} = \{[\boldsymbol{g} - \varepsilon(\boldsymbol{g})][\boldsymbol{g} - \varepsilon(\boldsymbol{g})]^T\} \tag{3}$$

where ε is the expectation operator. The PCA of a random vector $\boldsymbol{g}$ factorizes its covariance matrix $\Sigma_{\boldsymbol{g}}$ into the following form:

$$\Sigma_{\boldsymbol{g}} = \Phi \Lambda \Phi \quad \text{with} \quad \Phi = [\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, \ldots, \boldsymbol{\varphi}_n], \quad \Lambda = diag\{\lambda_1, \lambda_2, \ldots, \lambda_n\} \tag{4}$$

where $\Phi \in \mathbb{R}^{n \times n}$ is an orthogonal eigenvector matrix and $\Lambda \in \mathbb{R}^{n \times n}$ is a diagonal eigenvalues matrix with diagonal elements in decreasing order ($\lambda_1 \geq \lambda_2 \geq \ldots \lambda_n$). An immediate application of PCA is dimensionality reduction via

$$\boldsymbol{u} = T_{pca}{}^T[\boldsymbol{g} - \varepsilon(\boldsymbol{g})] \tag{5}$$

where eigenvectors $T_{pca} = [\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, \ldots, \boldsymbol{\varphi}_p]$, $p < n$ and $T \in \mathbb{R}^{n \times p}$. The lower dimensional vector captures the most expressive features of the original data. Then we use the projection $\boldsymbol{u}$ for classification.

4   Feature Classifiers

4.1   Nearest Neighbor Classifier

The NN decision is a simple nonparametric classifier by finding the neighbor within the minimum distance between the feature of query $\boldsymbol{u}$ and all prototypes

$\boldsymbol{u}_{train}^{l}$, $l = 1, 2, \ldots, L$. The feature vector $\boldsymbol{u}$ is classified to the closet training feature. The similarity measures used in our experiments to evaluate the efficiency of different representation and recognition methods include L_1 distance measure $\triangle_{L_1}$, L_2 distance measure $\triangle_{L_2}$, cosine distance measure $\triangle_{\cos}$ and independent Mahanlanobis distance measure $\triangle_{Mah}$. In order to make the computation feasible, the Mahanlanobis distance measure can be obtained by:

$$\triangle_{SimMah} = \sum_{i=1}^{m} \frac{(u_i - v_i)^2}{\varsigma_i^2} \tag{6}$$

where m is the number of remaining eigenvectors and ς_i^2 is the maximum likelihood estimation of the ith feature.

4.2   Nearest Feature Space Classifier

Stan Li et al. [4] showed that NFL method is superior to NN methods. In [5], Jen-Tzung Chien et al. expanded the NFL concept through extending the geometrical concepts of point and to plane and space in order to enlarge training prototype capacity. NN searches the nearest feature point; NFL searches the nearest feature line constructed by two prototype feature points in each class, and so on. And NFS method has the advantage to find the nearest feature distance between query and prototype space.

We find that searching the nearest feature distance should be finally degenerated to a problem tackle with distance between query feature point and its projection feature point in expanded feature space, plane and line according to the geometrical concepts. So the problem can be eventually solved through nearest neighbor method, where the projection prototype feature point of query feature point is replaced with the training prototype. Chien et al. only argued L_1 norm, but other measure methods such as L_2 norm and angle distance are very useful for pattern classification task, too. So we extend the distance concept in [5] to squared Euclidean distance and angle distance. Since NFS shows priority than NFL and NFP by Chien et al's conclusions, our paper only focusses on the NFS-related methods.

Assuming all the training prototypes belong to c classes, and each class ω_i has N_k sample respectively. The r training prototypes $\{\boldsymbol{u}_{\omega 1}, \boldsymbol{u}_{\omega 2}, \ldots, \boldsymbol{u}_{\omega r}\}$ is part of class ω spanning a space $S_\omega = span(\boldsymbol{u}_{\omega 1}, \boldsymbol{u}_{\omega 2}, \ldots, \boldsymbol{u}_{\omega r})$. The projection of query feature vector $\boldsymbol{u}$ on S_ω can be determined by applying the Gram-Schmidt process or build a matrix $P = [\boldsymbol{u}_{\omega 1}, \boldsymbol{u}_{\omega 2}, \ldots, \boldsymbol{u}_{\omega r}]$, the projection $\boldsymbol{u}_P$ is derived as:

$$\boldsymbol{u}_P = P(P^T P)^{-1} P^T \boldsymbol{u} \tag{7}$$

After applying NN classifier for $\boldsymbol{u}$ and $\boldsymbol{u}_P$, the query feature vector can be labeled. So the similarity measure except Mahanlanobis distance for NN can also be used in NFS. In particular, the projection and probed features must be normalized for angle distance similarity measure.

4.3   Enhanced Nearest Feature Space Classifier

In [6], C. Liu et al. argue that the conditional probability density function for
each class is implemented in the reduced PCA subspace. It is modeled using the
within class scatter and the Maximum A Posteriori (MAP) classification rule .
The PRM model assumes in the PCA space the within-class covariance matrices
are identical and diagonal under $\Sigma_i = \Sigma_I = diag\{\sigma_1^2, \sigma_2^2, \cdots, \sigma_c^2\}$. Each diagonal
component σ_i^2 is estimated by the sample variance in the one dimensional PCA
space:

$$\sigma_i^2 = \frac{1}{c} \sum_{k=1}^{c} \{ \frac{1}{N_k - 1} \sum_{j=1}^{N_k} (u_{ji}^{(k)} - m_{ki})^2 \} \tag{8}$$

where $u_{ji}^{(k)}$ is the ith element of the Gabor Wavelets Features $g_j^{(k)}$ which belongs
to class ω_k, m_{ki} is the ith of $\boldsymbol{m}_k$, and $\boldsymbol{m}_k = E\{\boldsymbol{g} \mid \omega_k\}$. Thus the MAP rule
specifies the following classifier:

$$\triangle_{PRM} = \sum_{j=1}^{c} \frac{(u_j - m_{ij})^2}{\sigma_j^2} \tag{9}$$

NFS method can enlarge the prototype capacity, and increases the generaliza-
tion ability. Mahanlanobis distance takes the variance of the training prototypes
into account and thus makes the similarity measure insensitive large residuals
in areas of high variance. PRM models can estimate the within class variance,
which can optimize the class seperability in the sense of Bayes error. We are
motivated to combine them together. Taking the same assumption used in PRM
model, the variance ς_i^2 in (6) can be replaced of the estimated variance σ_i^2 of
within-class covariance. Using the L_2 distance between the probe feature vector
$\boldsymbol{u}$ and the projection $\boldsymbol{u}_P$, the distance eventually can be derived as:

$$\triangle_{ENFS} = \sum_{i=1}^{c} \frac{(u_i - u_{P_i})^2}{\sigma_i^2} \tag{10}$$

We then call this classifier as Enhanced Nearest Feature Space (ENFS) classifier.

5   Experiments

5.1   Experimental Setup

Experiments are conducted by ORL dataset for our proposed Gabor Wavelets
Features with extended NFS classifier. The ORL dataset consists of 400 frontal
faces: 10 tightly cropped images of 40 individuals with variations in pose, illu-
mination, facial expression and accessories. The size of each image is 92×112
pixels, with 256 grey levels per pixel.
    We test the proposed method in both the performance against conventional
eigenfaces method [3] and the recent face recognition scheme such as Kernel
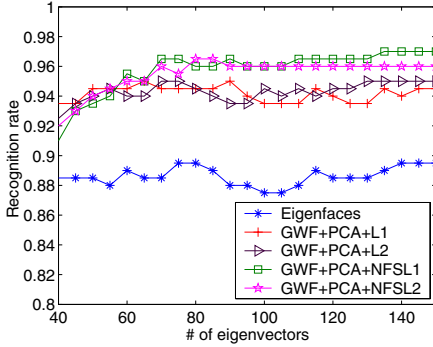
Fig. 3. Compare NFS with NN using PCA on Gabor Wavelets Features
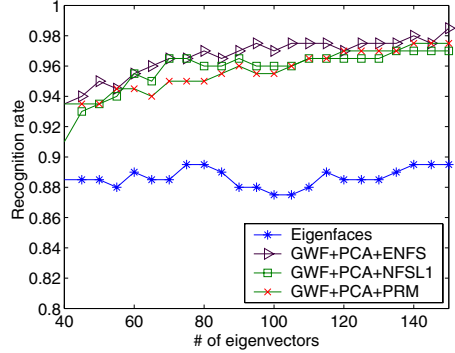
Fig. 4. Compare ENFS with NFS using PCA on Gabor Wavelets Features

PCA [10], NFL [4], NFP and NFS[5]. In this research, eigenfaces [3] combined with NN classifier method is viewed as baseline system. To allow comparisons, the same training and test set sizes are used as in [4, 10], i.e., the first 5 images for each subject are the training images and the remaining 5 images (unseen during training) are used for testing, and no overlap exists between the training and test face images. To facilitate the Gabor wavelet transform for face images, all images in the dataset are scaled to 128×128. In our experiment, most of angle distance similarity measure and Mahanlanobis distance measure are close to the performance of L_1 norm and L_2 norm, we therefore only focus on the latter measure in the following experimental results.

5.2   Effects of Classifiers

We first experiment with original ORL images by applying the Eigenfaces method [3]. Fig.3 shows face recognition performance of the Eigenfaces method when using between 40 and 150 eigenvectors in PCA dimensionality reduction step. It can be seen from Fig.3 that the standard Eigenfaces method has 87.5% to 89.5% correct recognition rates using between 40 and 150 eigenvectors.

The scaled ORL images are used for all flowing experiments. Fig.3 also shows the correct recognition rates obtained by NN and NFS using enlarged number of eigenvectors after performing PCA on the downsamped Gabor Wavelets features. We observe that NFS classifier with various similarity measures is regularly better than NN classifier. As the number of eigenvectors increases, performance improves and then saturates. The face recognition performance curves indicate that downsampled Gabor Wavelets Features based methods is superior to pixels value based Egienfaces method. And NFS classifier with L_1 norm is marginally better than L_2 norm.

When performing PCA on the downsampled Gabor Wavelets Features, the correct recognition rates plotted in Fig.4 are obtained by NFS classifier and

our proposed ENFS classifier respectively. Fig.4 indicates that ENFS classifier performs better than NN classifier.

The experiment compares ENFS classifier with PRM classifier is also be done. Fig.4 shows the results obtained by employing ENFS classifier and PRM classifier on the downsampled Gabor Wavelets Features. The recognition curve shows that PRM is also an effective classifier, but our proposed classifier works better even. For our proposed method, the best performance of 98.5% correct recognition rate is achieved when the number of eigenvectors is 145, which clearly outperformed NN, NFS and PRM classifier based methods in our experiments.

6   Summary and Conclusions

The experiments in this paper suggest the following conclusions:

1) Downsampled Gabor wavelets transform of face images as features for face recognition in subspace approach is superior to pixel value based one.

2) We propose a new Enhanced NFS classifier which shows better discriminant power than NN, NFS and PRM classifier in our experiments. It mainly benefits from employing features space projection for expanding the prototype capacity and increase generalization power. The classification performance has been enhanced in reduced PCA subspace using Bayes Rule to estimate the within class variance at the same time. This scheme make the classifier more robust and precise than the original one. Such classifier couple with Gabor Wavelets Features achieves best correct face recognition accuracy for ORL data set(98.5% max). In addition, NFS classifier using different similarity measures may yield greater discriminant ability.

References

[1] R. W. Chellappa, C. L. and Sirohey, S.: Human and machine recognition of faces: a survey. Proceedings of the IEEE **83** (1995) 705–741.   124

[2] M. Kirby, and L. Sirovich: Application of the Karhunen-Loeve procedure for the characterization of human faces. IEEE Trans. PAMI **12** (1990) 103–108.   124

[3] M. A. Turk, and A. P. Pentland: Face recognition using eigenfaces. CVPR (1991). 124, 128, 129

[4] S. Z. Li, and J. Lu: Face recognition using the nearest feature line method. IEEE Trans. Neural Networks **10** (1999) 439–443.   124, 127, 129

[5] J.-T. Chien, and C.-C. Wu: Discriminant waveletfaces and nearest feature classifiers for face recognition. IEEE Trans. PAMI **24** (2002) 1644–1649.   124, 125, 127, 129

[6] C. Liu, and H. Wechsler: Probabilistic reasoning models for face recognition. CVPR (1998).   124, 128

[7] C. Liu, and H. Wechsler: Robust coding schemes for indexing and retrieval from large face databases. IEEE Trans. Image Processing **9** (2000) 132–137.   124

[8] OlivettiOracle Research Lab (ORL): http://mambo.ucsc.edu/psl/olivetti.html. 125

[9] M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Wurtz, and W. Konen: Distortion invariant object recognition in the dynamic link architecture. IEEE Trans. Computers. **42** (1993) 300–311.   125

[10] K. I. Kim, K. Jung, and H. J. Kim: Face recognition using kernel principal component analysis. IEEE Trans. Signal Processing Letters **9** (2002) 40–42.   129

A Two-Stage Dimensional Reduction Approach to Low-Dimensional Representation of Facial Images

Jongmoo Choi and Juneho Yi

School of Information & Communication Engineering
Sungkyunkwan University
300 Chunchun-Dong, Jangan-Gu Suwon 440-746, Korea
{jmchoi,jhyi}@ece.skku.ac.kr

Abstract. We present a two-stage dimensional reduction approach to low-dimensional representation. When facial feature data need to be stored in low capacity storing devices, low-dimensional representation of facial images is very important. Our approach is composed of two consecutive mappings of the input data. The first mapping is concerned with best separation of the input data into classes and the second focuses on the mapping that the distance relationship between data points before and after the map is kept as closely as possible. We claim that if data is well-clustered into classes, features extracted from a topology-preserving map of the data are appropriate for recognition when low-dimensional features are to be used. We have presented two novel methods: FLD (Fisher's Linear Discriminant) combined with SOFM (Self-Organizing Feature Map) method and FLD combined with MDS (Multi-Dimensional Scaling) method. Experimental results using Yale, AT&T and FERET facial image databases show that the recognition performance of our methods degrades gracefully when low-dimensional features are used.

1 Introduction

The problem on extremely low-dimensional image representation for face recognition has little been investigated while many researchers study on face recognition robust to illumination, posture and facial expression changes. In practical biometric user authentication systems, low-dimensional feature extraction is one of the most important problem. When facial feature data need to be stored in low capacity storing devices such as bar codes and smart cards, extremely low-dimensional image representation of facial data is very important. It can also be used for data transmission in the internet or mobile environments. Moreover, it is applicable to real-time identification in the case of a large database.

The algorithms like PCA (Principal Components Analysis) [1], FLD (Fisher's Linear Discriminant) [2] and ICA (Independent Components Analysis) [3] can be used for reduction of the dimension of the input data but are not appropriate for low-dimensional representation of high dimensional data because their recognition performance degrade significantly. Although SOFM (Self-Organizing

Feature Map) [4], PP (Projection Pursuit) [5] and MDS (Multi-Dimensional Scaling) [6] can be employed for low-dimensional data representation, these techniques are suitable for data representation in low-dimensions, usually two or three dimensions. They try to represent the data points in a such way that the distances between points in low-dimensional space correspond to the dissimilarities between points in the original high dimensional space. However, these techniques do not yield high recognition rates mainly because they do not consider class specific information. Our idea is that these methods incorporated with class specific information can provide high recognition rates.

In this research, we present a two-stage dimensional reduction approach to low-dimensional data representation of which the recognition performance degrades gracefully. The proposed approach reduces the dimension of high-dimensional input data as much as possible, while preserving the information necessary for the pattern classification. Our idea is that if data is well-clustered into classes, features extracted from a topology-preserving map of the data are appropriate for recognition when low-dimensional features are to be used. Based on this idea, we apply a mapping to the input data to achieve the most separation of classes, followed by another mapping to preserve the topology of the data that the first map produces. By "topology-preserving map", we mean that vectors in the neighborhood in the input space are projected in the neighborhood in the output space [4].

To experimentally prove our claim, we have presented two novel methods for extremely low-dimensional representation of data with graceful degradation of recognition performance. It is composed of two consecutive mappings of the input data. The first mapping is concerned with best separation of the input data into classes and the second focuses on the mapping in the sense that the distance relationship between data points is kept. Our methods are implemented as the following. The first method employs FLD and SOFM. SOFM preserves the distance relationship before and after the data is transformed. This way, it is possible to represent data in low-dimensions without serious degradation of recognition performance. The second method uses FLD and MDS. The MDS preserves the distance relationship before and after the data is transformed as closely as possible.

The following section gives a brief overview of the feature extraction and dimensional reduction methods that have preciously been used for object recognition. In section 3, we describe the proposed methods: FLD combined with SOFM method and the FLD combined with MDS method. Let us call them 'FLD+SOFM' and 'FLD+MDS' methods, respectively. We report the experimental results on the recognition performance of FLD+SOFM and FLD+MDS methods in section 4.

2   Dimensional Reduction and Topology-Preserving Map

There have been reported many algorithms for dimensional reduction and feature extraction. One group of dimensional reduction methods can be referred

to as *topology-preserving mapping* and another group as *well-clustered mapping*. Among the former group are SOFM, MDS and GTM (Generative Topographic Mapping) [7] and these methods are used mainly for data visualization or data compression. FLD, Kernel FLD [8] and multi-layer neural networks are examples of the latter group and are mostly used for pattern classification [9].

We can achieve very low-dimensional data representation with graceful degradation of performance by using a *topology-preserving map* when the data is well clustered into classes. However, the typical facial image data in real environments do not have well-clustered distribution and it is not guaranteed to achieve high classification performance by a *topology-preserving map* although we can get a low-dimensional data set. Accordingly, we have to focus more on the discriminant power rather than dimensional reduction in the case.

3   Two-Stage Dimensional Reduction

We present a two-stage dimensional reduction approach to low-dimensional data representation by applying two different maps in a row. The first stage is only concerned with best separation of classes. Once the data is rendered well-separated into classes by the first stage map, the second stage map only focuses on preservation of topological continuity before and after the map of the data. As previously described, the idea is based on the fact that if data is well-clustered into classes, features extracted from a topology-preserving map of the data are appropriate for recognition when extremely low-dimensional features are to be used.

3.1   Method I: FLD+SOFM

Let us $\mathbf{x}_k \in \mathbb{R}^N, k = 1, \cdots, M$ be a set of training data. FLD produces a linear discriminant function $\mathbf{f}(\mathbf{x}) = \mathbf{W^T}\mathbf{x}$ which maps the input data onto the classification space. FLD finds a matrix $\mathbf{W}$ that maximizes

$$J(\mathbf{W}) = \frac{|\mathbf{W}^T \mathbf{S}_b \mathbf{W}|}{|\mathbf{W}^T \mathbf{S}_w \mathbf{W}|} \tag{1}$$

where $\mathbf{S}_b$ and $\mathbf{S}_w$ are between- and within-class scatter matrices, respectively. $\mathbf{W}$ is computed by maximizing $J(\mathbf{W})$. That is, we find a subspace where, for the data projected onto the subspace, between-class variance is maximized while minimizing within-class variance. As a result of the first map, we obtain $\mathbf{z} = \mathbf{W}^T\mathbf{x}$.

After the stage of FLD, the next stage maps $\mathbf{z}$ onto a low-dimensional feature space $\mathbf{f} = \mathbf{G}(\mathbf{z})$ by SOFM. SOFM is a kind of competitive network. SOFM first determines the winning neuron using a competitive layer. Next, weight vectors for all neurons within a certain neighborhood of the winning neuron are updated using the Kohonen rule [4]. When a vector is presented, the weights of the winning neuron and its neighbors move toward the input pattern. After learning, the neurons of the output layer will be a feature map revealing a distance relationship within input patterns.

3.2   Method II: FLD+MDS

Let us $\mathbf{x}_k \in \mathbb{R}^N, k = 1, \cdots, M$ be a set of observations and $\mathbf{D}$ be a dissimilarity matrix. Classical MDS is an algebraic method to find a set of points in low-dimensional space so that the dissimilarity are well-approximated by the interpoint distances.

In summary, the inner product matrix of raw data $\mathbf{B} = \mathbf{X^T X}$ can be computed by $\mathbf{B} = -\frac{1}{2}\mathbf{HDH}$, where $\mathbf{X}$ is the data matrix $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_M] \in \mathbb{R}^{N \times M}$ and $\mathbf{H}$ is a centering matrix $\mathbf{H} = \mathbf{I} - \frac{1}{\mathbf{M}}\mathbf{1^T 1}$. $\mathbf{B}$ is real, symmetric and positive semi-definite. Let the eigendecomposition of $\mathbf{B}$ be $\mathbf{B} = \mathbf{V \Lambda V^T}$, where $\mathbf{\Lambda}$ is a diagonal matrix and $\mathbf{V}$ is a matrix whose columns are the eigenvectors of $\mathbf{B}$. The matrix $\hat{\mathbf{X}}$ for low-dimensional feature vectors can be obtained as $\hat{\mathbf{X}} = \mathbf{\Lambda}_\mathbf{k}^{\mathbf{1/2}}\mathbf{V}_\mathbf{k}^\mathbf{T}$ where $\mathbf{\Lambda}_k^{1/2}$ is a diagonal matrix of k largest eigenvalues and $\mathbf{V}_k$ is its corresponding eigenvectors matrix. Thus, we can compute a set of feature vectors, $\hat{\mathbf{X}}$, for a low-dimensional representation. See [10] for a detailed description.

We could not map new input vectors to features by using the classical MDS because the map is not explicitly defined in the method [11]. We used a method that achieves mapping onto an MDS subspace via PCA based on the relationship between MDS and PCA. Let $\mathbf{Y_{MDS}}$ be a set of feature vectors in an MDS subspace and $\mathbf{Y_{PCA}}$ be a set of feature vectors in a PCA subspace. Let $\mathbf{\Lambda_{MDS}}$ denotes the diagonal matrix of eigenvalues of inner product matrix $\mathbf{B}$. Then, the relationship between PCA and MDS is

$$\mathbf{Y_{PCA}} = \mathbf{\Lambda_{MDS}}^{1/2}\mathbf{Y_{MDS}}. \tag{2}$$

For the purpose of low-dimensional feature extraction, we need to compute projections onto FLD and MDS subspaces. Let $\mathbf{p}$ be an input pattern, then the feature vector in FLD+MDS space becomes

$$\mathbf{f_{FLD+MDS}} = (\mathbf{\Lambda_{PCA}^{-1/2}})\mathbf{W_{PCA}^T W_{FLD}^T}\,\mathbf{p}. \tag{3}$$

See [12] for a detailed description.

4   Experimental Results

We have evaluated the recognition performance of the proposed FLD+SOFM and FLD+MDS methods as follows.

4.1   Experiment I: FLD+SOFM with Yale and AT&T Databases

We have used Yale [13] and AT&T [14] databases. Yale database contains 165 images of 15 persons and AT&T database contains 400 images of 40 persons. We tightly cropped and normalized all the facial images.

In the SOFM stage, the entire training patterns are represented by the indices of neurons corresponding to two-dimensional map. In testing, only the node that is the most similar to the given input pattern is activated. As a result,

Table 1. Correct Recognition Rates (%) (C: number of class)

Dimension	Methods	Yale (C=15)	AT&T (C=40)
2	PCA	16.4	11.9
	FLD	41.8	11.9
	SOFM	64.3	71.3
	FLD+SOFM	96.4	86.2
	FLD+MDS	65.5	42.5
C-1	PCA	87.3	94.0
	FLD	98.2	94.8
	FLD+MDS	100.0	91.8

input patterns are classified into classes of the activated nodes. In the proposed method, the number of input neurons in SOFM is the same as the dimension of feature vectors obtained from the FLD stage. The output layer represents a two dimensional square map.

We have applied cross validation because the performance of the SOFM algorithm varies depending on the initial parameters. First, we change the number of grids. After learning using multiple SOFMs, we evaluate the performance using the validation set. We have decided the number of neurons as the number of grids that have the highest average recognition performance. Secondly, after the number of neurons is settled, multiple SOFMs with various initial parameters are learned by the learning set. Then we select the SOFM that has high performance corresponding to the upper 10% in the validation set.

As shown in Table 1, FLD+SOFM method performs better than the others in the case of very low-dimensional representation. The recognition rate of FLD is high (98.2%) when a sufficient number, C-1, of features are used. However, the recognition rate degraded significantly to 41.8% when only two dimensional representation of the data is used. The recognition rate of SOFM is higher than that of FLD when two dimensional representation is employed.

4.2   Experiment II: FLD+MDS with FERET Database

We have compared the recognition performance of FLD [2] and the proposed FLD+MDS method using a part of FERET database [15]. The whole set of images, U, consists of three subsets named 'ba', 'bj' and 'bk'. Basically, the whole set U contains images of 200 persons and each person in the U has three different images within the 'ba', 'bj' and 'bk' sets. The 'ba' set is a subset of 'fa' which has images with normal frontal facial expression. The 'bj' set is a subset of 'fb'. The images of 'fb' have some other frontal facial expressions. The 'ba' and 'bj' set contain 200 images of 200 persons, respectively. The 'bk' set is equal to the 'fc' of which images were taken with different cameras and under different lighting conditions. The 'bk' set contains 194 images of 194 persons.

For the experiment, we have divided the whole set U into training set (T), gallery set (G) and probe set (P). No one within the training set (T) is included in
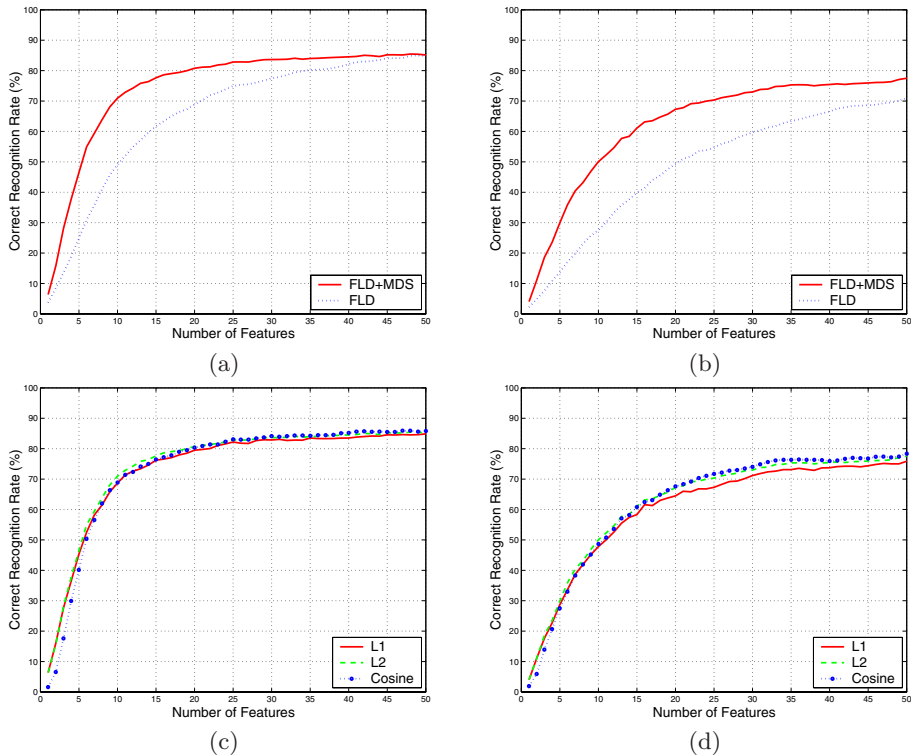
Fig. 1. Comparison of recognition rates: (a) and (b) represent recognition rates for 'ba'-'bj' set and 'ba'-'bk' set, respectively. (c) and (d) represent recognition rates for various distance measures in the case of 'ba'-'bj' set and 'ba'-'bk' set, respectively

the gallery and the probe sets. The experiment consists of two sub-experiments; The first experiment is concerned with evaluation regarding normal facial expression changes. We use the 'ba' set as the gallery and the 'bj' set as the probe. The second experiment is to evaluate the performance under illumination changes. We have assigned the 'ba' set to the gallery and the 'bk' set to the probe. In addition, we randomly selected 50% of the whole set in each sub-experiment in order to reduce the influence of a particular training set because a facial recognition algorithm based on statistical learning depends on the selection of training images. Thus, a training set contains 100 persons in each sub-experiment.

As shown in Figure 1, FLD+MDS method performs better than the others in the case of low-dimensional representation. The experimental results show that low-dimensional data representation with graceful degradation of recognition performance can be achieved by using an inter-distance preserving map after the input data is rendered well clustered into classes. The recognition rate for a given number of features in these figures was obtained by averaging thirty experiments.

We can see that there is no significant performance difference between the three distance measures (L1, L2 and cosine).

5   Conclusion

This research features a novel approach to low dimensional reduction of facial data that do not give significant degradation of the recognition rate. We have proposed two methods. The FLD+SOFM method achieves very accurate recognition rates although only two dimensional features are used for recognition. The FLD+MDS method also outperforms FLD method when represented in a low-dimensional space. These results experimentally prove that if data is tightly clustered and well separated into classes, a few features extracted from a topology-preserving map of the data are appropriate low dimensional features for recognition without significant degradation of recognition performance.

Our methods are practically useful for face recognition, especially when facial feature data need to be stored in low capacity storing devices such as bar codes and smart cards. It is also readily applicable to real-time face recognition in the case of a large database.

Acknowledgements

References

[1] Turk, M., Pentland, A.: "Eigenfaces for Recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991. 131
[2] Belhumeur, P., Hespanha, J., Kriegman, D.: "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Trans. on PAMI*, vol. 19, no. 7, pp. 711–720, 1997. 131, 135
[3] Bartlett, M. S., Martin, H., Sejnowski, T. J.: "Independent Component Representations for Face Recognition," *Proceedings of the SPIE*, Vol. 3299, pp. 528–539, 1998. 131
[4] Kohonen, T.: *Self-Organizing Maps*, Springer-Verlag, 1995. 132, 133
[5] Friedman, J. K., Tukey, J. W.: "A Projection Pursuit Algorithm for Exploratoty Data Analysis," *IEEE Trans on computers*, vol. 23, pp. 881–889, 1974. 132
[6] Duda, R. O., Hart, P. E., Stork, D. G.: *Pattern Classification*, John Wiley & Sons, Inc., 2001. 132
[7] Bishop, C. M., Svensén, M.: "GTM: The Generative Topographic Mapping," *Neural Computation*, Vol. 10. No. 1, pp. 215–234, 1998. 133
[8] Mika, S., Rätsch, G., Weston, J., Schölkopf, B., Müller, K. R.: "Fisher Discriminant Analysis with Kernels," *IEEE Neural Networks for Signal Processing IX*, pp. 41–48, 1999. 133

[9] Carreira-Perpoñán, M.: "A Reivew of Dimension Reduction Techniques," *Technical Report CS-96-09*, Dept. of Computer Science University of Sheffield, 1997. 133

[10] Pcekalska, E., Paclík, P., Duin, R. P. W.: "A Generalized Kernel Approach to Dissimilarity-based Classification," *Journal of Machine Learning Research*, vol. 2, pp. 175–211, 2001. 134

[11] Chandrasiri, N. P., Park, M. C., Naemura, T., Harashima, H.: "Personal Facial Expression Space based on Multidimensional Scaling for the Recognition Improvement", Proc. IEEE ISSPA'99, pp. 943–946, 1999. 134

[12] Choi, J. M., Yi, J. H.: "Low-Dimensional Image Representation for Face Recognition," *Workshop on Multimodal User Authentication*, 2003. 134

[13] http://cvc.yale.edu/projects/yalefaces/yalefaces.html. 134

[14] http://www.uk.research.att.com/facedatabase.html. 134

[15] Phillips, P. J., Moon, H. J., Rizvi, S. A., Rauss,. P. J.: "The FERET Evaluation Methodology for Face-Recognition Algorithms," *IEEE Trans. on PAMI*, vol. 22, No. 10, pp. 1090–1104, 2000. 135

Face Recognition
with 3D Model-Based Synthesis

Xiaoguang Lu[1], Rein-Lien Hsu[1], Anil K. Jain[1],
Behrooz Kamgar-Parsi[2], and Behzad Kamgar-Parsi[2]

[1] Michigan State University
East Lansing, MI 48824, USA
{lvxiaogu,hsureinl,jain}@cse.msu.edu
[2] Office of Naval Research
800 N. Quincy St., Arlington, VA 22217, USA

Abstract. Current appearance-based face recognition system encounters the difficulty to recognize faces with appearance variations, while only a small number of training images are available. We present a scheme based on the analysis by synthesis framework. A 3D generic face model is aligned onto a given frontal face image. A number of synthetic face images are generated with appearance variations from the aligned 3D face model. These synthesized images are used to construct an affine subspace for each subject. Training and test images for each subject are represented in the same way in such a subspace. Face recognition is achieved by minimizing the distance between the subspace of a test subject and that of each subject in the database. Only a single face image of each subject is available for training in our experiments. Preliminary experimental results are promising.

1 Introduction

After decades of research [1], face recognition is still a very challenging topic. Current systems can achieve a good performance when the test image is taken under similar conditions as the training images. However, in real applications, a face recognition system may encounter difficulties with intra-subject facial variations due to varying lighting conditions, different head poses and facial expressions. Most of the face recognition methods are appearance-based [2, 3, 4, 5, 6] which require that several training samples be available under different conditions for each subject. However, only a small number of training images, are generally available for a subject in real applications, which can not capture all the facial variations.

A human face is a 3D elastic surface, so the 2D image projection of a face is very sensitive to the changes in head pose, illumination, and facial expression. Utilizing 3D facial information is a promising way to deal with these variations [5, 6, 7, 8, 9, 10, 11, 12]. Adopting Waters' animation model [9] as our generic face model, we propose a face recognition system that synthesizes various facial variations to augment the given training set which contains only

a single frontal face image for each subject. Both the training and test images are subjected to the model adaptation and synthesis in the same way. We use the synthetic variations to construct an affine subspace for each subject. The recognition is achieved by matching the subspace of the test image with that for each of the subjects in the training database. Yamaguchi et al. [13] used the minimal principal angle between two subspaces, which are generated from two sets of images, to measure the dissimilarity between them. However, the mean of the set of images is not taken into account. We use an alternative distance metric to measure the dissimilarity between two affine subspaces.

2   Face Synthesis

2.1   Face Alignment

The face alignment module adapts a 3D generic face model [10, 9] onto a face image to extract facial shape and texture information. Waters' animation model contains 256 vertices and 441 triangular polygons for one half of the face. The other half of the face can be generated using the symmetry assumption of the human face. The face alignment is based on two processes: labeling and adaptation. In *labeling*, feature vertices are chosen from the 3D model. Currently, the 2D projected positions of these feature vertices (totally 115 feature vertices in our experiments) are labeled manually in the given intensity image. Figures 1(a)(b) illustrates these feature vertices. In *adaptation*, vertices in the 3D model, other than the feature vertices, are adjusted iteratively based on the propagation of feature vertex displacement [14].

Since there is no depth information available in the given 2D intensity image, the depth information (Z coordinate in Fig. 1(c)) of each model vertex is adapted by scaling the original generic model based on the average value of the global scaling factors (in the X and Y coordinates) of the generic model. Although the reconstructed face is not very realistic (the model is not dense enough), Fig. 1(d) shows that the face identity is preserved in the frontal view.

After face alignment, each model vertex is mapped with an intensity value at its corresponding coordinate on the face image. Texture values over non-vertex regions of each triangular facet on the 3D face model are interpolated by the intensity values of facet's vertices. See Fig. 1(d) for an example.

2.2   Eye Augmentation

There is no mesh for the eye regions in Water's model, so we create an additional mesh in the eye region to augment the original 3D mesh model. In each eye region, based on the vertices on its boundary, a grid is generated. The augmented eye mesh is obtained by using the Delaunay triangulation on these vertices. These vertices are adjusted according to the adaptation of the boundary vertices (see Fig. 3(b) for an example of the eye augmented reconstruction result).
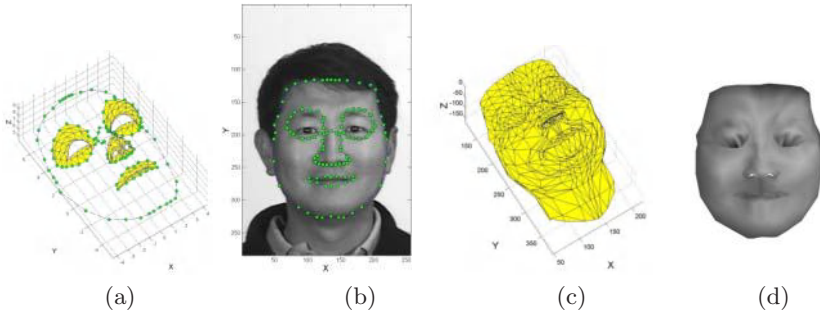
Fig. 1. Face alignment: (a) feature vertices shown as "beads" on the 3D generic face model; (b) overlaid on a given intensity face image; (c) adapted 3D face model; (d) reconstructed images using the model shown in (c) with texture mapping

2.3   Facial Variation Synthesis

The face synthesis module synthesizes variations in head pose, illumination, and facial expression as follows. Rotating the face model and projecting it to the image plane can generate different pose variations. Lighting is simulated by adding a virtual light source around the reconstructed face surface. Phong shading technique is employed to render lighting effects on the face surface [15]. We use Terzopoulos and Waters [9, 10] approach of physics-based synthetic facial tissue and a set of anatomically motivated facial muscle actuators to synthesize facial expressions, see figure 2. Figure 3 shows the given intensity face image and several synthesis results from the adapted 3D model. Different types of synthesis (pose, lighting and expression) are done independently, so the combination of different types of synthesis is seamless.
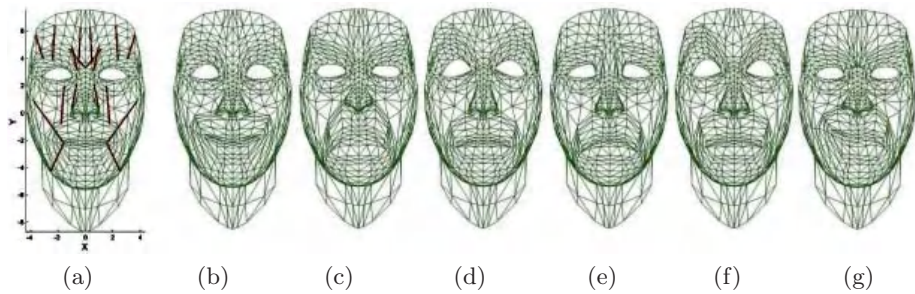
Fig. 2. Expression synthesis through 18 muscle contractions. The generic face mesh is: (a) shown in neutral expression (the dark bars represent 18 muscle vectors); distorted with six facial expressions (b) happiness ; (c) anger; (d) fear; (e) sadness; (f) surprise; (g) disgust
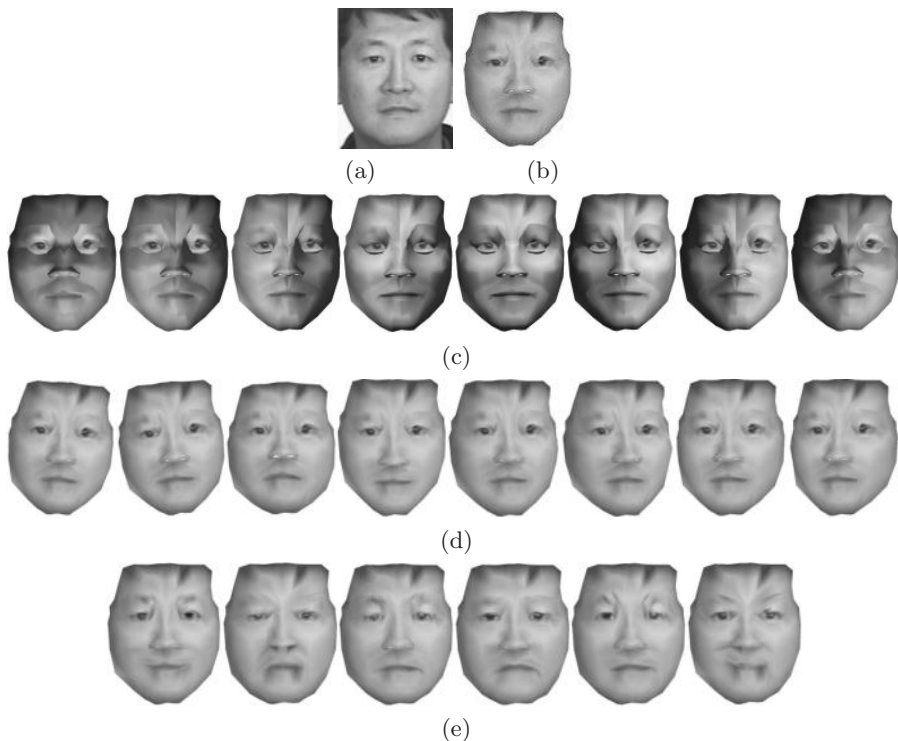
Fig. 3. Synthesis results: (a) input intensity image; (b) the reconstructed image; (c) image variations under 8 different lighting conditions; (d) 8 different pose variants; (e) synthesis results for 6 different expressions

3   Face Matching

3.1   Subspace Construction

For each subject, besides the input training image, a number of synthetic images are available after face synthesis. An affine subspace is constructed based on the original and synthetic images of this subject for face representation.

In the classical subspace analysis, an image is represented as a high-dimensional vector by concatenating each row (or column) of the image. Given a set of linearly independent vectors, Gram-Schmidt orthogonalization can be used to obtain a set of orthogonal basis vectors to span the subspace. An affine subspace [16] M is of the form $M = M_0 + L$, where M_0 is a fixed vector and L is a subspace. Here, we call M_0 as the center of the affine subspace. Therefore, the sample mean of each set of images is taken into account in the following matching. Let $X_i(i = 1, \ldots, N)$ be the set of image vectors (original image and synthesized image vectors) for one subject, M be the affine subspace of this subject. The center M_0 of M is calculated as $M_0 = \frac{1}{N} \sum_{i=1}^{N} X_i$. Then image
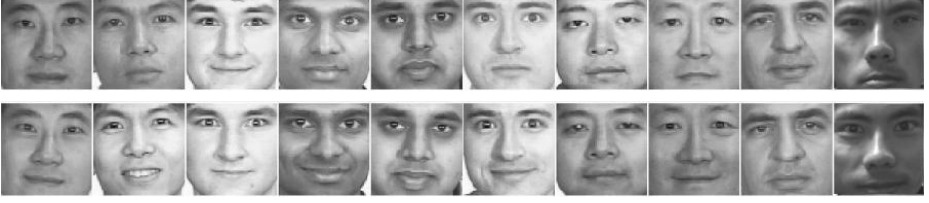
Fig. 4. Sample face images in the data set. All cropped images are aligned using the two eye centers

vectors $\tilde{X}_i = X_i - M_0$, are used to generate a set of basis vectors to span the subspace L, i.e., $L = \sum_{i=1}^{N} w_i \tilde{X}_i$, where w_i is a weight on $\tilde{X}_i$. Thus any vector X in this affine subspace can be represented as

$$X = M_0 + \sum_{i=1}^{N} w_i(X_i - M_0). \tag{1}$$

3.2 Subspace Matching

The distance between two affine subspaces (DAS) is defined as the minimum Euclidean distance between any two vectors from two different affine subspaces.

$$DAS = \min \|P - Q\|, \tag{2}$$

where P and Q are any vectors coming from two different affine subspaces. Finding this minimum distance is an optimization problem, whose close-form solution is derived in Appendix A.

4   Experiments and Discussion

For each of the ten subjects, a frontal image is captured once a week over a period of five weeks. As a result, 5 images for each subject are captured, resulting in a total of 50 images. The size of face area in the images is approximately 128×128. Figure 4 shows 20 cropped images in our database, two samples for each subject. One image from the 5 images of each subject is randomly selected to construct the training set. Use the rest of the data set as the test set. This process is repeated 50 times. The same adaptation, synthesis and affine subspace analysis procedures are applied to each image.

In our experiments, a total of 8 images per subject with different poses are synthesized. They correspond to the in-plane-rotation to the left and right by 5 degrees (2 images are synthesized); tilt up and down by 10 degrees (2 synthetic images); pan to left and right by 5 and 10 degrees (4 synthetic images). The light source is placed above the face and 8 different angles are chosen to render the simulated lighting effect. A total of 8 synthetic images per subject are
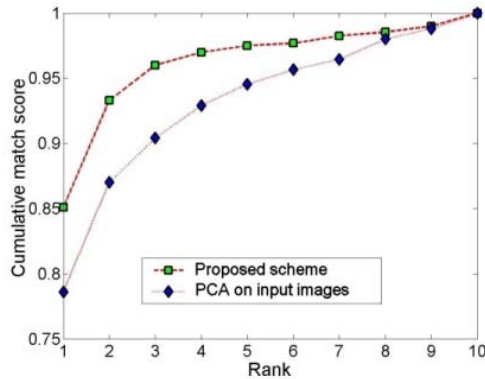
Fig. 5. Performance of proposed method and classical PCA based (without data augmentation by synthesis) method

generated. Using the approach and parameters proposed in [9], we synthesize 6 different expressions per subject. Thus, a total of 22 synthetic images for each subject are obtained. Twenty two images based on one input image per subject are illustrated in Fig. 3. The images are closely cropped[4]. Cropped images are aligned by the centers of the two eyes and normalized to 64×64. All image vectors are normalized to be of unit length. For comparison purposes, the classical eigenface framework [2] (PCA) is applied to the data set without any synthesis augmentation. We captured 5 additional images for each subject. This additional data set is used for augmenting the original data to construct the eigenspace.

The matching results are illustrated in Fig. 5. On this data set our proposed method outperforms the PCA-based (without synthesis) method, indicating that the proposed synthesis for recognition scheme is promising.

5   Conclusions and Future Work

We have proposed a human face recognition framework with 3D model based synthesis scheme to augment data set for face recognition when only a single training sample for each subject is available. A novel distance metric are utilized to measure the dissimilarity between two affine subspaces, which are constructed by the synthetic set of images. We are currently investigating the automatic alignment of a 3D generic model onto an intensity image. Experiments on a large public domain face database are also being carried out.

Acknowledgements

References

[1] Zhao, W., Chellappa, R., Rosenfeld, A., Phillips, P.: Face recognition: A literature survey. CVL Technical Report, University of Maryland (2000), ftp://ftp.cfar.umd.edu/TRs/CVL-Reports-2000/TR4167-zhao.ps.gz   139
[2] Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience **3** (1991) 71–86   139, 144
[3] Lee, K. C., Ho, J., Kriegman, D. J.: Nine points of light: acquiring subspaces for face recognition under variable lighting. In: Proc. CVPR. (2001) 519–526   139
[4] Belhumeur, P. N., Hespanha, J. P., Kriegman, D. J.: Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. IEEE Trans. PAMI **19** (1997) 711–720   139, 144
[5] Zhao, W., Chellappa, R.: Face recognition using symmetric shape from shading. In: Proc. CVPR. (2000) 286–293   139
[6] Sim, T., Kanade, T.: Combining models and exemplars for face recognition: An illuminating example. In: Proc. CVPR 2001 Workshop on Models versus Exemplars in Computer Vision. (2001)   139
[7] Blanz, V., Vetter, T.: A morphable model for the synthesis of 3d faces. In: Proc. ACM SIGGRAPH. (1999) 187–194   139
[8] Romdhani, S., Blanz, V., Vetter, T.: Face identification by matching a 3d morphable model using linear shape and texture error functions. In: Proc. ECCV. Volume 4. (2002) 3–19   139
[9] Parke, F. I., Waters, K.: Computer Facial Animation. A. K. Peters Ltd. (1996), http://crl.research.compaq.com/publications/books/waters/Appendix1/appendix1.html   139, 140, 141, 144
[10] Terzopoulos, D., Waters, K.: Analysis and synthesis of facial image sequences using physical and anatomical models. IEEE Trans. PAMI **15** (1993) 569–579   139, 140, 141
[11] Essa, I., Pentland, A.: Coding, analysis, interpretation, and recognition of facial expressions. IEEE Trans. PAMI **19** (1997) 757–763   139
[12] Guenter, B., Grimm, C., Wood, D., Malvar, H., Pighin, F.: Making faces. In: Proc. ACM SIGGRAPH. (1998) 55–66   139
[13] Yamaguchi, O., Fukui, K., ichi Maeda, K.: Face recognition using temporal image sequence. In: Proc. IEEE FG'98, Nara, Japan (1998) 318–323   140
[14] Hsu, R., Jain, A.: Face modeling for recognition. In: Proc. IEEE ICIP. Volume 2. (2001) 693–696   140
[15] Foley, J., van Dam, A., Feiner, S., Hughes, J.: Computer Graphics: Principles and Practice. 2nd ed. Addison-Wesley, New York (1996)   141
[16] Oja, E.: Subspace Methods of Pattern Recognition. Research Studies Press (1983)   142

Appendix A: Distance between Two Affine Subspaces

The distance between two affine subspaces (denoted as DAS) is defined as the minimum distance between any vectors in the two affine subspaces. Let M_u and M_v be two affine subspaces. Any vectors u (in M_u) and v (in M_v) can be represented as $u = \mu_u + U t_u, v = \mu_v + V t_v$, where μ_u and μ_v are the centers of M_u and M_v, U and V are the basis matrixes spanning the subspace L of M_u and M_v, t_u and t_v are the coefficients vectors, respectively.

$$H = (u - v)^T (u - v) = (\mu_{uv} + U t_u - V t_v)^T (\mu_{uv} + U t_u - V t_v), \qquad (3)$$

where H is the square of the DAS, . Taking the derivative of H, find the u and v (i.e. t_u and t_v) that minimizes H.

$$\begin{cases} \frac{\partial H}{\partial t} = (\mu_{uv} + Ut_u - Vt_v)^T U \\ \frac{\partial H}{\partial t} = (\mu_{uv} + Ut_u - Vt_v)^T V \end{cases} \tag{4}$$

$$(\mu_{uv}^T \; \mu_{uv}^T) = (t_u^T \; t_u^T) \begin{pmatrix} -U^T U & -U^T V \\ V^T U & V^T V \end{pmatrix}. \tag{5}$$

If the rightmost matrix in Eq. 5 is not singular, the projection coefficients can be derived as follows:

$$(t_u^T \; t_u^T) = (\mu_{uv}^T U \; \mu_{uv}^T V) \begin{pmatrix} -U^T U & -U^T V \\ V^T U & V^T V \end{pmatrix}^{-1}. \tag{6}$$

ICA Based Face Recognition Robust
to Partial Occlusions and Local Distortions

Jongsun Kim, Jongmoo Choi, and Juneho Yi

School of Information & Communication Engineering
Sungkyunkwan University
300, Chunchun-dong, Jangan-gu Suwon 440-746, Korea
{jskim,jmchoi,jhyi}@ece.skku.ac.kr

Abstract. We propose a novel method using a perfectly local facial representation based on ICA. We named our method "LS-ICA method". In the LS-ICA method, basis images are made from their corresponding ICA basis images simply by removing non-salient regions. The LS-ICA basis images display perfectly local characteristics because they contain only locally salient feature regions. This enables us to effectively realize the idea of "recognition by parts" for face recognition. Experimental results using AT&T, Harvard, FERET and AR databases show that the recognition performance of the LS-ICA method outperforms that of PCA and ICA methods especially in the cases of facial images that have partial occlusions and local distortions such as changes in facial expression and at low dimensions.

1   Introduction

PCA and ICA are the most widely used subspace projection techniques in the face recognition research [2-3]. These techniques help to reveal low dimensional structures of patterns observed in high dimensional spaces. Figure 1 (a) and (b) show facial image representations using PCA and ICA basis images, respectively, that are computed from a set of images randomly selected from the AR database. PCA basis images display global properties in the sense that they assign significant weights to potentially all the pixels. This accords with the fact that PCA basis images are just scaled versions of global Fourier filters [13]. In contrast, ICA basis images are spatially more localized, highlighting salient feature regions corresponding to eyes, eyebrows, nose and lips.

The local property of ICA basis images makes the performance of ICA based recognition methods better than PCA based methods in terms of robustness to partial occlusions and local distortions, such as changes in facial expression, because spatially local features only influence small parts of facial images. Thus, ICA techniques have popularly been applied to the problem of face recognition [3-5]. However, ICA basis images do not display perfectly local characteristics in the sense that pixels that do not belong to locally salient feature regions still have some non-zero weight values. These pixel values in non-salient regions appear as noise and contribute to the degradation of the recognition performance as shown in the experimental results.

(a) PCA representation = $(e_1, e_2, e_3, e_4, e_5, ..., e_n)$

(b) ICA representation = $(b_1, b_2, b_3, b_4, b_5, ..., b_n)$

(c) LS-ICA representation = $(b'_1, b'_2, b'_3, b'_4, b'_5, ..., b'_n)$

Fig. 1. Facial image representations using (a) PCA, (b) ICA and (c) LS-ICA basis images: A face is represented as a linear combination of basis images. The basis images were computed from a set of images randomly selected from the AR database. In the basis images of LS-ICA, non-salient regions of ICA basis images are removed. Using LS-ICA basis images, the concept of "recognition by parts" can be effectively implemented for face recognition

This research features a novel method using a perfectly local facial representation based on ICA. We named our method "LS-ICA (locally salient ICA) method". In the LS-ICA method, basis images contain only locally salient feature regions. The idea of "recognition by parts" can be effectively realized for face recognition using LS-ICA basis images since each LS-ICA basis image represents only locally salient regions. These regions correspond to important facial feature regions such as eyes, eyebrows, nose and lips. "Recognition by parts" [11-12] has been a popular paradigm in the object recognition research because the approach can be successfully applied to the problem of object recognition with occlusion. Our method for face recognition is characterized by two ideas. The first is the creation of LS-ICA basis images from ICA basis images and uses them to represent faces. Each LS-ICA basis image is made from its corresponding ICA basis image simply by removing non-salient regions. The second idea is to use ICA basis images in the decreasing order of class separability so as to maximize the recognition performance. We have tested three different distance metrics (L1 norm, L2 norm, and cosine angle) for PCA, ICA and LS-ICA method. The idea is simple but works great. Experimental results show that LS-ICA performs better than PCA and ICA, especially in the cases of partial occlusions and local distortions such as changes in facial expression. In addition, the performance improvement of LS-ICA over ICA based methods is much greater as we decrease the dimensionality (i. e., the number of basis images used).

The rest of this paper is organized as follows. Section 2 describes ICA versus PCA in terms of spatial locality of features. Section 3 explains the proposed LS-ICA method. Section 4 presents experimental results.

2   ICA versus PCA

PCA and ICA are the most widely used subspace projection techniques that project data from a high-dimensional space to a lower-dimensional space [2-3]. PCA addresses only second-order moments of the input. It is optimal for finding a reduced representation that minimizes the reconstruction error, but it is not optimal for

classification. ICA is a generalization of PCA that decorrelates the high-order statistics in addition to the second-order moments. Much of information about characteristic local structure of facial images is contained in the higher-order statistics of the images. Thus ICA, where the high-order statistics are decorrelated, may provide a more powerful representational basis for face recognition than PCA, where only the second-order statistics are correlated. Fig. 2. illustrates PCA and ICA axes for the same 2D distribution. PCA finds an orthogonal set of axes pointing in the directions of maximum covariance in the data, while ICA attempts to place axes pointing in the directions of spatially localized and statistically independent basis vectors [10].

   As previously described, global properties of faces may be more easily captured by PCA than ICA. As shown in Figure 1, ICA basis images are more spatially localized and never overlap unlike their PCA counterpart [1]. Since spatially localized features only influence small parts of facial images, ICA based recognition methods are less susceptible to occlusions and local distortions than are global feature based methods such as PCA. We can compute ICA basis images using various algorithms such as InfoMax [3], FastICA [4] and Maximum likelihood [5].
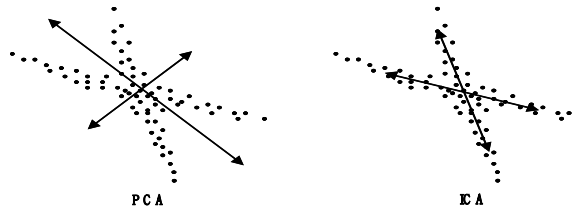
Fig. 2. PCA and ICA axes for an identical 2D data distribution [10]

3   The LS-ICA (Locally Salient ICA) Method

The LS-ICA method features the use of new basis images made from ICA basis images that are selected in the decreasing order of class separability. Only salient feature regions are contained in the LS-ICA basis images. As in most algorithms that employ subspace projection, the LS-ICA method computes a projection matrix, off-line from a set of training images. Let W_{ls-ica} denote the projection matrix. The columns of W_{ls-ica} are LS-ICA basis images. During recognition, given an input face image $\mathbf{x}$, it is projected to $\mathbf{\Omega}' = W_{ls-sica}^{T}\mathbf{x}$ and classified by comparison with the vectors $\mathbf{\Omega}_{T}$'s that were computed off-line from a set of training images.

   First, we preprocess training images by applying histogram equalization and scale normalization, where the size of images is adjusted so that they have the same distance between two eyes. Second, we compute ICA basis images, using the FastICA algorithm [4]. The FastICA method computes the independent components that become uncorrelated by a whitening process and then maximizes non-Gaussianity of data distribution by using kurtosis maximization. We then compute a measure of class separability, r, for each ICA basis vector and sort the ICA basis vectors in the decreasing order of class separability [3]. To computer r for each ICA basis vector,

the between-class variability $\sigma_{between}$ and within-class variability σ_{within} of its corresponding projection coefficients of training images are obtained as follows.

$$\sigma_{between} = \Sigma_i \left(M_i - M \right)^2 \tag{1}$$

$$\sigma_{within} = \Sigma_i \Sigma_j \left(b_{ij} - M_i \right)^2 \tag{2}$$

M and M_i are the total mean and the mean of each class, and b_{ij} is the coefficient of the jth training image in class i. The class separability, r, is then defined as the ratio

$$r = \frac{\sigma_{between}}{\sigma_{within}} . \tag{3}$$

Third, we create LS-ICA basis images from the ICA basis images selected in the decreasing order of the class separability. This way, we can achieve both dimensional reduction and good recognition performance. To create an LS-ICA basis image, we apply a series of operations to its corresponding ICA basis image as shown in Fig. 4. In order to detect locally salient regions, we simply find extreme values by thresholding a histogram of pixel values (Figure 3 (b)), followed by the application of morphological operations to find a blob region (Figure 3 (d)). As a result, we get an LS-ICA basis image (Figure 3 (e)) where only pixels in the blob regions have grey values copied from the corresponding pixels in the original ICA image. The values of the rest of the pixels in the image are set to zero. These LS-ICA basis images are used to represent facial images as shown in Figure 1 (c).

4   Experimental Results

We have used several facial image databases such as AT&T [6], Harvard [7], FERET [8] and AR [9] databases in order to compare the recognition performance of LS-ICA with that of PCA and ICA methods. For fair comparisons with PCA and ICA based methods, PCA and ICA basis images were also used in the decreasing order of class separability, r. In the case of the ICA method, the recognition performance was greater when the basis images were ordered in terms of class separability. However, the PCA method did not show any noticeable performance difference between the ordering in the class separability and the orginal ordering in terms of eigenvalues.

(a) ICA basis image   (b) detection of extreme values   (c) erosion   (d) dilation   (e) LS-ICA basis image
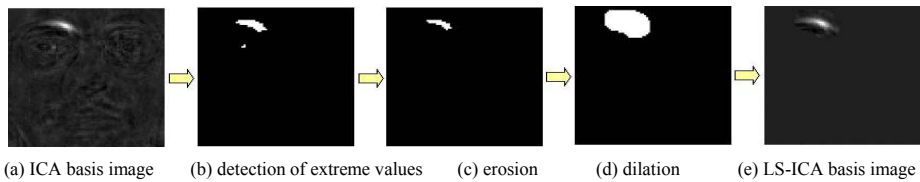
Fig. 3. Illustration of creating an LS-ICA basis image

Table 1. Facial databases used in the experiment

Database	The number of total images	The number of persons	The number of training images	The number of testing images
AT&T	400	40	200	200
Harvard	165	5	82(83)	83(82)
FERET	605	127	127	478
AR	800	100	200	600

Table 1 lists the number of training and testing images used in each facial image databases for the experiment. Fig. 4. shows example images from these databases. In the AT&T database, all the images are taken against a dark homogeneous background and the subjects are in an up-right, frontal position with tolerance for some side movement. In Harvard database, a subject held his/her head steady while being illuminated by a dominant light source. In the FERET database, we have used a subset of the images of subjects under significantly different lighting and facial expression. The AR database contains local distortions and occlusions such as changes in facial expression and sunglasses worn.

All images were converted to 256 gray-level images and background regions were removed. We have also applied histogram equalization to both training and testing images in order to minimize variations of illumination. We have experimented using thirty different sets of training and testing images for each database. We have computed recognition performances for three different distance measures (L1, L2, cosine) since we are concerned with performance variations independent of the distance measure used [1].

Fig. 5. shows the recognition performances of PCA, ICA and LS-ICA methods for the four facial databases. The recognition rate of the LS-ICA method was consistently better than that of PCA and ICA methods regardless of distance measures used. ICA also consistently outperformed PCA except the case where the L1 measure was used for the FERET database. What is more interesting is that LS-ICA method performed better than the other methods especially at low dimensions. This property is very important when we need to store facial feature data in a low capacity storing devices such as smart cards and barcodes. To clearly show this, we displayed in Figure 6 the performance improvement of the LS-ICA method over the ICA method. The performance improvement was the greatest in the case of the AR database, as we expected. The AR database contains local distortions and occlusions such as sunglasses worn. The LS-ICA method that only makes use of locally salient information can achieve a higher recognition rate than ordinary ICA methods that are influenced by pixels not belonging to salient regions. The experimental results show that, especially at low dimensions, LS-ICA basis images better represent facial images than ICA basis images.

Fig. 4. Example images from AT&T (top left), Harvard (top right), FERET (bottom left) and AR (bottom right) facial databases
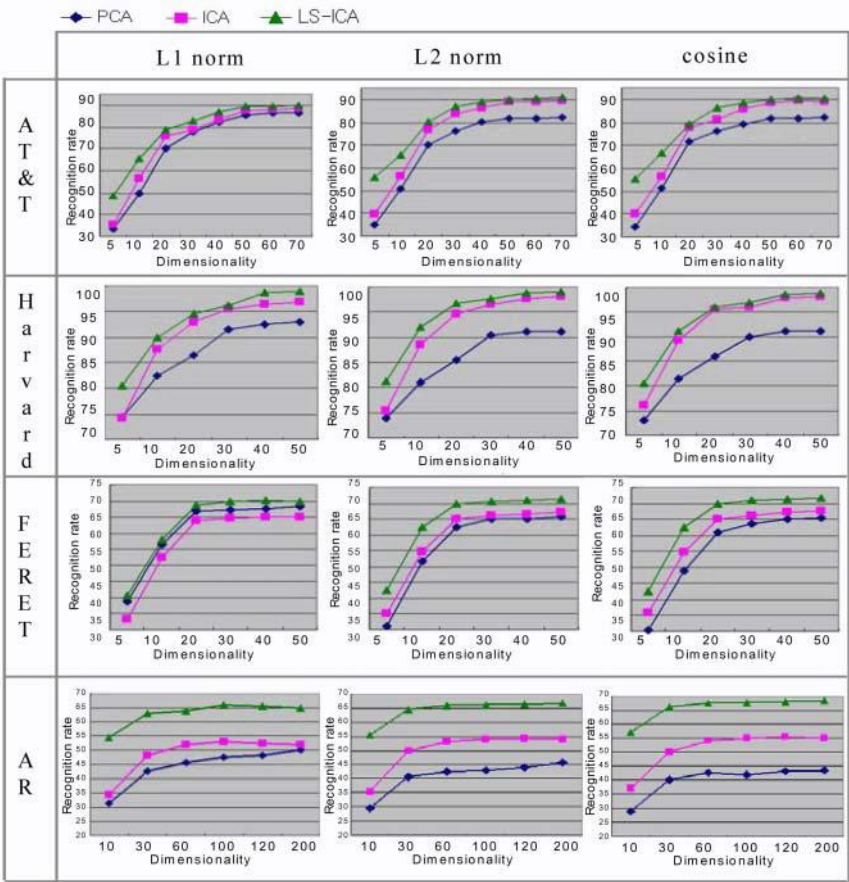
Fig. 5. The recognition performance of PCA, ICA and LS-ICA methods for the four facial databases. The recognition rates represent the average performance of the experiment using thirty different sets of training and testing images
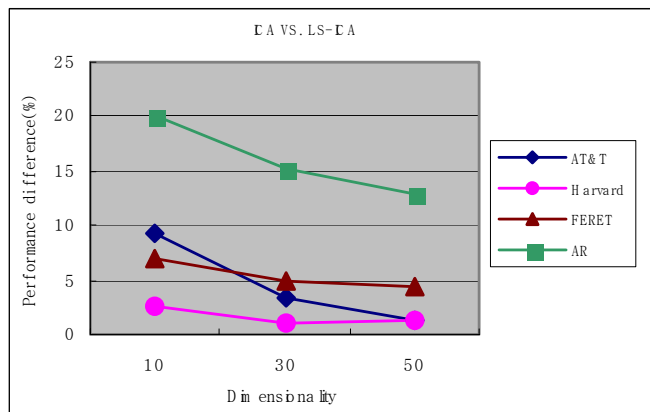
Fig. 6. The performance improvement of LS-ICA method over ICA method for the four facial databases. The performance improvement was the greatest in the case of AR database that contains local distortions and occlusions such as sunglasses worn

5   Conclusions

We have proposed the LS-ICA method that only employs locally salient information in order to maximize the benefit of applying the idea of "recognition by parts" to the problem of face recognition under partial occlusion and local distortion. The performance of the LS-ICA method was consistently better than the ICA method regardless of the distance measures used. As expected, the effect was the greatest in the cases of facial images that have partial occlusions and local distortions such as changes in facial expression.

Acknowledgements

References

[1]   K. Back, B. A. Draper, J. R. Beveridge, and K. She, "PCA vs ICA: A comparison on the FERET data set," Joint Conference on Information Sciences, Durham, N. C., 2002.
[2]   M. A. Turk and A. P. Pentland, "Eigenfaces for recognition," Cognitive Neuroscience, vol.3, no.1, pp. 71-86, 1991.
[3]   M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski, "Face Recognition by Independent Component Analysis," IEEE Transaction on Neural Networks, Vol 13, pp. 1450-1464, 2002.
[4]   Aapo Hyvarinen and Erki Oja, "Independent component analysis: a tutorial," http://www.cis.hut.fi/~aapo/papers/IJCNN99_tutorialweb/, 1999.

[5]   J. F. Cardoso, "Infomax and Maximum Likelihood for Source Separation," IEEE Letters on Signal Processing, vol. 4, pp. 112-114, 1997.

[6]   http://www.uk.research.att.com/facedatabase.html.

[7]   http://cvc.yale.edu/people/faculty/belhumeur.htm.

[8]   P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET Evaluation Methodology for Face Recognition Algorithms," Pattern Analysis and Machine Intelligence, vol. 22, pp. 1090-1104, 2000.

[9]   M. Martinez and R. Benavente, "The AR face database," CVC Tech. Report #24, 1998.

[10]   M. S. Bartlett, "Face Image Analysis by Unsupervised Learning," Foreword by T. J. Sejnowski, Kluwer International Series on Engineering and Computer Science, Boston: Kluwer Academic Publishers, 2001.

[11]   P. Pentland, "Recognition by parts," IEEE Proceedings of the First International Conference on Computer Vision, pp. 612—620, 1987.

[12]   D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," Nature, vol. 401, pp.788-791, 1999.

[13]   J. Bell and T. J. Sejnowski, "The independent components of natural scenes are edge filters," Advance in Neural Information Processing Systems 9, 1997.

One-Pass Incremental Membership Authentication by Face Classification

Shaoning Pang[1], Seiichi Ozawa[2], and Nikola Kasabov[1]

[1] Knowledge Engineering & Discover Research Institute
Auckland University of Technology, Private Bag 92006, Auckland 1020, New Zealand
{spang,nkasabov}@aut.ac.nz
[2] Graduate School of Science and Technology, Kobe University
Kobe 657-8501, Japan
ozawa@eedept.kobe-u.ac.jp

Abstract. Real membership authentication applications require machines to learn from stream data while making a decision as accurately as possible whenever the authentication is needed. To achieve that, we proposed a novel algorithm which authenticated membership by a one-pass incremental principle component analysis(IPCA) learning. It is demonstrated that the proposed algorithm involves an useful incremental feature construction in membership authentication, and the incremental learning system works optimally due to its performance is converging to the performance of a batch learning system.

1 Introduction

Membership authentication problem [1] can be depicted as follows. Consider a certain human group G with N members, which is the universal set. If there exists an arbitrary subgroup M such that $M \subset G$ and $|M| < N/2$, then it is a membership group, and the remaining peoples $\bar{M} = G - M$ makes a non-membership group. Thus, membership authentication problem is to distinguish the membership class M from the non-membership class $\bar{M}$ in the human group.

Obviously, the membership authentication by face classification can be simply treated as a 2-class classification problem, in which either member or non-member is judged by the system as a human tries to get authentication. However, in the real situation of online membership authentication, the difficulties of the problem are as follows, (1)Both the size of membership/non-membership group and the members in the membership group/non-membership group are dynamically changed. (2)There are little within-class similarities, either for the membership class or for the nonmembership class. (3)Face images for training are provided as a stream dataset, which means that except for an initial dataset, other datasets are added incrementally.

To deal with the first three points of difficulty, our previous work [1] proposed an SVM ensemble method for membership authentication in dynamic face groups. To achieve a dynamical authentication, we performed the membership authentication in terms of binary classification without revealing the

individual's identity. To deal with the classification difficulty from little within-class similarity, we used a SVM ensemble [2] combining several binary SVM classifiers. Here, towards solving the third difficulty, we introduce a algorithm of one-pass incremental membership authentication(IMA).

2   Incremental Membership Authentication: Problem Definition

Given $M = \{\boldsymbol{x}_i\}$ is a membership face group of G, and $\bar{M} = G - M$ is the non-membership face group, then membership authentication by face classification can be depicted by a typical binary classification problem as,

$$f(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{x} \in M, \\ -1 & \text{otherwise.} \end{cases} \tag{1}$$

Suppose f can be implemented by a specific classification model C with a set of parameters $W = \{w_1, w_2, ...w_\xi\}$, where ξ is determined by the used model in practice. In this paper, we simply use a K-NN classification model, then G represents the K-NN computation, and w is the prototypes (reference centers).

For the batch learning of Eq.(1), the whole training dataset is given before learning is started. The learning procedure can be illustrated as the following formula,

$$C(w_1, w_2, ...w_\xi) \leftarrow Train(M, \bar{M}). \tag{2}$$

In contrast to the batch learning mode, incremental learning samples are provided as a stream of data. It follows that the incremental learning must be a procedure of recursive learning that can memorize the new coming faces, while not forget the faces that are already learned before. Thus the incremental authentication can be formulated as a recursive computation,

$$C(w_1^t, w_2^t, ...w_\xi^t) \leftarrow \tag{3}$$
$$Train(C(w_1^{(t-1)}, w_2^{(t-1)}, ..., w_\xi^{(t-1)}), M_t, \bar{M}_t)$$

where t identifies the current time, M_t and $\bar{M}_t$ denote current training set of membership and non-membership, respectively.

It is possible to apply batch learning method to learn stream data. But that requires system always to keep the previous training data and perform a repeating learning from scratch. The continually accumulating data might cause a memory disaster for the system.

3   One-Pass Incremental Learning

3.1   Incremental Principal Component Analysis (IPCA)

Since the original PCA is not suited for incremental learning purposes, Hall and Martin devised a method to update eigenvectors and eigenvalues in an incremental way [4].

Assume that N training samples $x_i \in R^n$ $(i = 1, \cdots, N)$ have been presented so far, and an eigenspace model $\Omega = (x, U, \Lambda, N)$ is constructed by calculating the eigenvectors and eigenvalues from the covariance matrix of x_i, where x is a mean input vector, U is a $n \times k$ matrix whose column vectors correspond to the eigenvectors, and Λ is a $k \times k$ matrix whose diagonal elements correspond to the eigenvalues. Here, k is the number of dimensions of the current eigenspace that is often determined such that a specified fraction of energy in the eigenvalue spectrum is retained.

Let us consider the case that the $(N + 1)$th training sample y is presented. The addition of this new sample will lead to the changes in both of the mean vector and covariance matrix; therefore, the eigenvectors and eigenvalues should also be recalculated. The mean input vector x is easily updated as follows:

$$x' = \frac{1}{N + 1}(Nx + y) \tag{4}$$

Therefore, the problem is how to update the eigenvectors and eigenvalues.

When the eigenspace model Ω is reconstructed to adapt to a new sample, we must check if the dimensions of the eigenspace should be changed or not. If the new sample has almost all energy in the current eigenspace, the dimensional augmentation is not needed in reconstructing the eigenspace. However, if it has some energy in the complementary space to the current eigenspace, the dimensional augmentation cannot be avoided. This can be judged from the norm of the following residue vector h:

$$h = (y - x) - Ug \tag{5}$$

where $g = U^T(y - x)$. When the norm of the residue vector h is larger than a threshold value η, it must allow the number of dimensions to increase from k to $k + 1$, and the current eigenspace must be expanded in the direction of h. Otherwise, the number of dimensions remains the same.

It has been shown that the eigenvectors and eigenvalues should be updated based on the solution of the following intermediate eigenproblem [4]:

$$\left(\frac{N}{N + 1} \begin{bmatrix} \Lambda & 0 \\ 0^T & 0 \end{bmatrix} + \frac{N}{(N + 1)^2} \begin{bmatrix} gg^T & \gamma g \\ \gamma g^T & \gamma^2 \end{bmatrix} \right) R = R\Lambda' \tag{6}$$

where $\gamma = h^T(y - x)$, R is a $(k + 1) \times (k + 1)$ matrix whose column vectors correspond to the eigenvectors obtained from the above intermediate eigenproblem, Λ' is the new eigenvalue matrix, and 0 is a k-dimensional zero vector. Using this solution R, we can calculate the new $n \times (k + 1)$ eigenvector matrix U' as follows:

$$U' = [U, \ h]R \tag{7}$$

where

$$h = \begin{cases} h/\|h\| & \text{if } \|h\| > \eta \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

As we can see from Eq. (7), $\boldsymbol{R}$ operates as a rotation of the eigenvectors; we will call $\boldsymbol{R}$ a rotation matrix. Note that if $\boldsymbol{h} = \boldsymbol{0}$, $\boldsymbol{R}$ degenerates into a $n \times k$ matrix; that is, the dimensions of the updated eigenspace remains the same as those of the previous eigenspace.

3.2   Evolving Clustering Method (ECM)

ECM is a fast one-pass algorithm for dynamic clustering of an input stream of data, where there is no predefined number of clusters. This algorithm is a distance-based clustering method where the cluster centers (called "prototypes") are determined online such that the maximum distance, *MaxDist*, between an input $\boldsymbol{x}_i$ and the closest prototype cannot be larger than a threshold value, *Dthr*. The ECM algorithm can be seen in [3].

3.3   The Proposed Method

As stated in [4], IPCA is utilized for reducing the dimensions of input data and constructing an appropriate feature space (i.e., eigenspace) based on an incoming data stream. In IPCA, depending on input data, the following two operations are carried out: eigen-axes rotation and dimensional augmentation of a feature space. On the other hand, ECM can evolve the prototypes which correspond to the representative points in the feature space constructed by IPCA. Hence, when the rotation and dimensional augmentation are carried out, all prototypes must be modified so as to keep consistency between the old and new eigenspaces.

When only the rotation is needed, each prototype $\boldsymbol{p}_j$ is easily updated by multiplying by the rotation matrix $\boldsymbol{R}$ as follows:

$$\boldsymbol{p}_j^{new} = \boldsymbol{R}^T \boldsymbol{p}_j, \text{ for } j = 1, \cdots, l. \tag{9}$$

where l is the number of prototypes. On the other hand, when the dimensional augmentation as well as the rotation is needed, the prototypes are not easy to be updated because the current prototypes have already lost information in the augmented complementary eigenspace. There are at least two methods to update prototypes. One is to approximate them as follows:

$$\boldsymbol{p}_j^{new} \simeq \boldsymbol{R}^T [\boldsymbol{p}_j^T, \ 0]^T, \text{ for } j = 1, \cdots, l \tag{10}$$

where $[\boldsymbol{p}_j^T, \ 0]^T$ is a $k + 1$ dimensional column vector which is given by adding a zero element to the current prototype $\boldsymbol{p}_j$. Another method is to define the following 2-fold prototypes $(\boldsymbol{p}_j, \boldsymbol{p}_j^*)$, where $\boldsymbol{p}_j^*$ is the original vector in the input space associated with the prototype $\boldsymbol{p}_j$; i.e., $\boldsymbol{p}_j = \boldsymbol{U}^T \boldsymbol{p}_j^*$ is hold for these prototypes. When the dimensional augmentation occurs, the new prototypes are updated using $\boldsymbol{p}_j^*$ instead of $\boldsymbol{p}_j$ as follows:

$$\boldsymbol{p}_j^{new} = \boldsymbol{R}^T \boldsymbol{U}^T \boldsymbol{p}_j^*, \text{ for } j = 1, \cdots, l \tag{11}$$

Note that the new prototypes obtained by Eq. (10) are only an approximation to those obtained by Eq. (11). However, if the approximation error could be ignored, the first method is a good choice in the terms of memory costs because it does not require keeping the $\boldsymbol{p}_j^*$ prototype. Let us assume that a small number of initial training data is given to form an initial eigenspace. We propose the following unsupervised one-pass incremental learning algorithm,

Step 0: Calculate the eigenvector matrix $\boldsymbol{U}$ and eigenvalue matrix $\boldsymbol{\Lambda}$ from the covariance matrix of initial training data. Calculate the projection of training data $\boldsymbol{x}_i$ into the eigenspace to obtain the feature vectors $\tilde{\boldsymbol{x}}_i$. Apply ECM to the feature vectors, and keep the selected two-fold prototypes $(\boldsymbol{p}_j, \boldsymbol{p}_j^*)$.

Step 1: Apply IPCA to a new training sample $\boldsymbol{x}_i$ and the current eigenspace model $\Omega = (\boldsymbol{x}, \boldsymbol{U}, \boldsymbol{\Lambda}, N)$ as follows:

1. Solve an intermediate eigenproblem in Eq. (6) to obtain a rotation matrix $\boldsymbol{R}$ and an eigenvalue matrix $\boldsymbol{\Lambda}'$.
2. Update the mean input vector $\boldsymbol{x}$ and eigenvector matrix $\boldsymbol{U}'$ based on Eqs. (4) and (7), respectively.
3. Increase the total number of training samples N by one.

Step 2: Add $\boldsymbol{x}_i$ to a training set S, and increase the size $|S|$ of the training set by one.

Step 3: If $|S|$ is larger than a certain value B, go to Step 4. Else, go back to Step 1.

Step 4: If the dimensional augmentation is not needed, update all the current prototypes $\boldsymbol{p}_j$ based on Eq. (9). Else, update them based on Eq. (11).

Step 5: For all training samples $\boldsymbol{x}_i$ in S, calculate the feature vectors $\tilde{\boldsymbol{x}}_i$ as follows:

$$\tilde{\boldsymbol{x}}_i = \boldsymbol{U}'^T \boldsymbol{x}_i, \text{ for } i = 1, \cdots, B. \tag{12}$$

Step 6: Initialize the training set S.

Step 7: Apply ECM to the feature vectors $\tilde{\boldsymbol{x}}_i$ ($i = 1, \cdots, B$), and keep the selected two-fold prototypes $(\boldsymbol{p}_j, \boldsymbol{p}_j^*)$.

Step 8: Go back to Step 1.

When a query sample is presented for classification purpose, the distances to all prototypes are calculated, and then the k nearest neighbor (k-NN) method is applied to determine the class.

4   Experiment Results and Discussions

4.1   Training

We perform various experiments to evaluate the proposed membership authentication technique using a benchmark MPEG-7 face dataset, which consists of 1,355 face images of 271 persons. (5 different face images per person are taken). We randomly selected a certain number of persons among 271 persons and assigned them as the membership group. The remaining persons are assigned as the non-membership group. To simulate the procedure of incremental membership
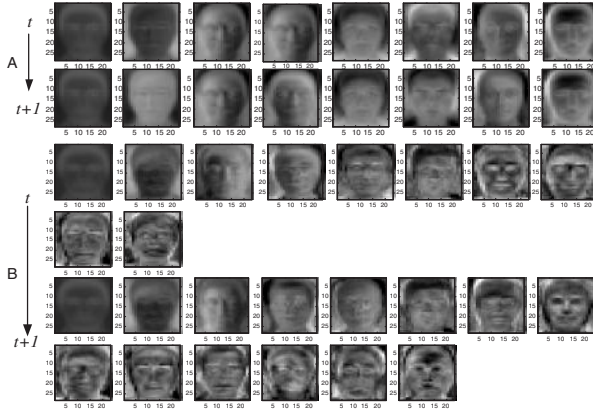
Fig. 1. Example of eigenspace variations

authentication, we used a data division as follows: at each step, 4 out of 5 face images of each person are used for training, and the remaining 1 face for test. The learning procedure is composed of an initial learning phase and numbers of incremental learning stages. For the convenience of performance evaluation, except for the initial training dataset where we used 5% of 271 persons (i.e., 20 images), all the other stage dataset are set as 5 persons for incremental learning at each stage, where both the percentage and the stage size may vary. Here, selecting a small percentage is just for a more distinct difference between PCA and IPCA on feature deriving.

We implemented the proposed one-pass incremental learning algorithm on the above simulated stream data. We observed that rotation and augmentation happen alternately and frequently during the incremental learning process. As rotation happening, eigenvectors are rotated. Fig. 1 (A) is an example of eigenspace rotation. As we can see, the eigenfaces are changed between two incremental learnings, but the number of eighenfaces are kept the same. Whereas as augmentation happening, since it is usually accompanied by the happening of rotation, the eigenfaces are not only rotated, but also increased in the number. Fig. 1 (B) is an example of eigenspace augmentation with rotation.

4.2   Comparison of Results

Fig.2 shows the time courses of the authentication accuracy in the case of 50 persons are randomly selected as members. Where we carried out a comparison among three methods, the proposed one-pass IMA, IMA with PCA and SVM ensemble [1], and the results were represented by three curves in the figure respectively. As can be seen, the performance of incremental membership authentication with IPCA is stable even if the incremental learning for new member/non-member proceeds. While with no incremental feature construction (using PCA), the system does not have such stability. As it is shown in the figure,
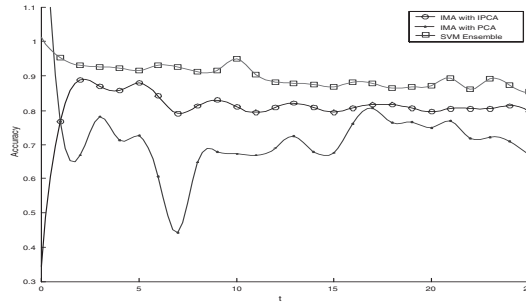
Fig. 2. Authentication comparison among IMA-IPCA, IMA-PCA and SVM ensemble

some fluctuations appear in the curve of IMA-PCA. However, the performance of IMA-IPCA is slightly lower than that of the SVM ensemble in which the learning is conducted in a batch mode by using the whole of the training dataset. We can see that the curves of IMA-IPCA and SVM ensemble in the figure become closer as the learning is extended over time. It indicates that the proposed incremental system is convergent, and it can perform classification with the same accuracy as the batch system do, as long as a incremental learning is carried on.

5   Conclusions

In this paper, we proposed an algorithm of one-pass incremental membership authentication method by face classification. The algorithm processes a face stream efficiently in one pass, and learns to make a decision as accurately as possible whenever the authentication is needed. We compared the proposed IPCA feature-derived incremental authentication with the PCA feature-derived authentication and the traditional batch-modelling authentication, the performance of the proposed algorithm is found to be converging to the performance of batch learning as the learning procedure proceeds, which imply us a very optimal incremental learning of face membership authentication.

References

[1] S. N. Pang, D. Kim, and S. Y. Bang, "Membership authentication in the dynamic group by face classification using SVM ensemble," Pattern Recognition Letters, Vol. 24, pp. 215-225, 2003. 155, 160
[2] S. N. Pang, D. Kim, and S. Y. Bang, "Fraud detection using support vector machine ensemble," Proc. ICORNIP2001, pp. 1344-1349, 2001. 156
[3] N. Kasabov, Evolving connectionist systems: Methods and applications in bioinformatics, brain study and intelligent machines, Springer-Verlag, 2002. 158
[4] P. Hall and R. Martin, "Incremental eigenanalysis for classification," British Machine Vision Conference, Vol. 1, pp. 286-295, 1998. 156, 157, 158

Face Recognition Using Asymmetric Faces

Srinivas Gutta[1] and Harry Wechsler[2]

[1] Philips Research Laboratories
Media Interaction Department
Prof. Holstlaan 4, 5656 AA Eindhoven, The Netherlands
{Srinivas.Gutta@philips.com}
[2] Department of Computer Science,
George Mason University, Fairfax, VA 22030, USA
{wechsler@cs.gmu.edu}

Abstract. Much of the research on face recognition has focused so far only on the identification from full frontal/profile facial images. This paper expands on our earlier results on recognizing faces from either their left or right half images as we now train the face recognition engine using full faces and test its recognition ability on asymmetric faces constructed from either their left or right half augmented by its mirror image. Our approach employs Ensemble of Radial Basis Functions (ERBF) as mixture of expert networks. The motivation for ERBF comes from their ability to cope with the inherent variability in the image formation and data acquisition process. Our experimental results, using k-fold Cross Validation (CV) on a database of 3000 images corresponding to 150 subjects, indicate that our synergetic approach, driven by asymmetric faces for representation and ERBF for classification, yields excellent results. The recognition rate was in excess of 90% and we observed asymmetric right (96%) is slightly better than the asymmetric left (92%) face.

1 Introduction

Faces are accessible 'windows' into the mechanisms that govern our emotional and social lives. The face is a unique feature of human beings. Even the faces of "identical twins" differ in some respects. Humans can detect and identify faces in a scene with little or no effort even if only partial views (due to occlusion) of the faces are available. This skill is quite robust, despite changes in the visual stimulus due to viewing conditions, and expression. Much of the research on face recognition has focused so far only on the identification from full frontal/profile facial images. Partial faces, due to occlusion, are all what is sometimes available, however, for scenarios characteristic of the *Face in a Crowd* challenge. This paper expands on our earlier results on the identification task, i.e., recognizing faces from either their left or right half images as we now train the face recognition engine using full faces and test its recognition ability on asymmetric faces constructed from either their left or right half augmented by its mirror image.

2   Face Recognition

Our earlier work on recognition from partial faces showed that the left half, right half and the full face yield similar performance when matched against similar types of face images [1, 2]. Liu et al. [3] have recently proposed to extend the use of facial asymmetry measures (which are a critical factor for evaluation of facial attractiveness [4] and expression [5]) to human identification.   One also notes that facial attractiveness for men is inversely related to recognition accuracy [6] and that asymmetrical faces are less attractive [4]. The explanation for those findings come from the simple observation that asymmetrical faces are more distinctive and this makes it easier to remember and recognize them. Martinez [7] working on the synthetic (1/6 to 1/3) occlusion problem divides the face into k local parts and then used a probabilistic method to find the best match. Different recognition rates for the left and right face images were reported, a finding not corroborated by our own studies [1, 2] conducted on a much larger database set.

   Liu et al. [3] argue that facial asymmetry measures (asymmetric faces that require a face midline defined in terms of two canthi and philtrum fiducial landmarks that are now manually annotated) contain discriminative information and provide useful synergy when combined with Fisherfaces and Eigen-face methods. The asymmetric faces, the Density Difference D-Face and the Edge Orientation Similarity S-Face, are derived using reflection with respect to the face midline. The method proposed in this paper is much simpler and its use is most suitable for the Face in a Crowd problem when one has access at most to half faces rather than full faces.   Our asymmetric faces are simpler to define and they are used for recognition rather than training the face recognition engine.

3   Ensemble of Radial Basis Function (ERBF) Networks

An RBF classifier has an architecture that is very similar to that of a traditional three-layer back-propagation network [8].   Connections between the input and middle layers have unit weights and, as a result, do not have to be trained. Nodes in the middle layer, called BF nodes, produce a localized response to the input using Gaussian kernels. Each hidden unit can be viewed as a localized receptive field (RF). The hidden layer is trained using k-means clustering. The most common basis function (BF) used are Gaussians, where the activation level y_i of the hidden unit i is given by:

$$y_i \; = \; \Phi_i \left(\|X - \mu_i\| \right) = \exp \left[- \sum_{k=1}^{D} \frac{\left(x_k - \mu_{ik} \right)^2}{2 h \sigma_{ik}^2 o} \right]$$

where h is a proportionality constant for the variance, x_k is the kth component of the input vector $X=[x_1, x_2, \ldots , x_D]$, and μ_{ik} and σ_{ik}^2 are the kth components of the mean and variance vectors, and o is the overlap factor, respectively, of basis function node i. The outputs of the hidden unit lie between 0 and 1, and could be interpreted

as fuzzy memberships; the closer the input to the center of the Gaussian, the larger the response of the node. The activation level Z_j of an output unit is given by:

$$Z_j = \sum_i w_{ij} y_i + w_{0j}$$

where Z_j is the output of the jth output node, y_i is the activation of the ith BF node, w_{ij} is the weight connecting the ith BF node to the jth output node, and w_{0j} is the bias or the threshold of the jth output node. The bias comes from the weights associated with a BF node that has a constant unit output regardless of the input. An unknown vector X is classified as belonging to the class associated with the output node j with the largest output Z_j.

The RBF input consists of n normalized face images pixels fed to the network as 1D vectors. The hidden (unsupervised) layer, implements an enhanced k-means clustering procedure, where both the number of Gaussian cluster nodes and their variance are dynamically set. The number of clusters varies, in steps of 5, from 1/5 of the number of training images to n, the total number of training images. The width of the Gaussian for each cluster, is set to the maximum of {*the distance between the center of the cluster and the member of the cluster that is farthest away - within class diameter, the distance between the center of the cluster and closest pattern from all other clusters*} multiplied by an overlap factor o, in our experiment equal to 2. The width is further dynamically refined using different proportionality constants h. The hidden layer yields the equivalent of a functional facial base, where each cluster node encodes some common characteristics across the face space. The output (supervised) layer maps face encodings ('expansions') along such a space to their corresponding class and finds the corresponding expansion ('weight') coefficients using pseudo-inverse techniques. In our case the number of nodes in the output layer correspond to the number of people we wish to identify.

For a connectionist architecture to be successful it has to cope with the variability available in the data acquisition process. One possible solution to the above problem is to implement the equivalent of query by consensus using ensembles of radial basis functions (ERBF). Ensembles are defined in terms of their specific topology (connections and RBF nodes) and the data they are trained on. Specifically, both original data and distortions caused by geometrical changes and blur are used to induce robustness to those very distortions via generalization [9].

The ERBF architecture is shown in Fig. 1. Each RBF component is further defined in terms of three RBF nodes, each of which specified in terms of the number of clusters and the overlap factors. The overlap factors o, defined earlier, for the RBF nodes RBF (11, 21, 31), RBF(12, 22, 32), and RBF(13, 23, 33) are set to 2, 2.5, and 3, respectively. The same RBF nodes were trained on original images, and on the same original images with either some Gaussian noise added or subject to some degree of geometrical ('rotation'), respectively. The intermediate nodes C_1, C_2, and C_3 act as buffers for the transfer of the normalized images to the various RBF components. Training is performed until 100% recognition accuracy is achieved for each RBF node. The nine output vectors generated by the RBF nodes are passed to a *judge* who would make a decision on whether the probe ('input') belongs to that

particular class or not. The specific decision used is - if the average of 5 of the 9 network outputs is greater than a threshold θ then that probe belongs to that class.

4   Data Collection

The number of unique individuals in our database corresponds to 150 subjects. During image acquisition each subject was asked to sit in front of the computer equipped with a Matrox frame grabber and a Philips CCD camera. The distance from the subject and the camera was approximately 3 feet.

Each subject was asked to first look at the camera for approximately 5 seconds and turn his/her head $\pm 5^0$. The frame grabber was set to acquire imagery at the rate of 5 fps. The images were acquired at a resolution of 640x480 pixels and encoded in 255 gray scale levels. The images are then passed to the face detection module [10]. Detected faces greater than the set threshold of 0.95 were stored. The faces are then normalized to account for geometrical and illumination changes using information about the eye location. The final face obtained at the end of detection and normalization is of a standard resolution of 64x72 pixels. Since we know the location of the eyes from the detection module, we create the partial-face by cutting the normalized facial image vertically at the point where the distance from one end of the image to the center is 1/2 the eye distance. Next a mirror image of the partial-face is constructed by flipping the partial-face image from left to right. Finally an asymmetric face image is obtained by combining the partial-face image with its mirror image. A sample set of face images and their corresponding partial faces and asymmetric face images are shown in Figs. 2, 3 and 4 respectively.
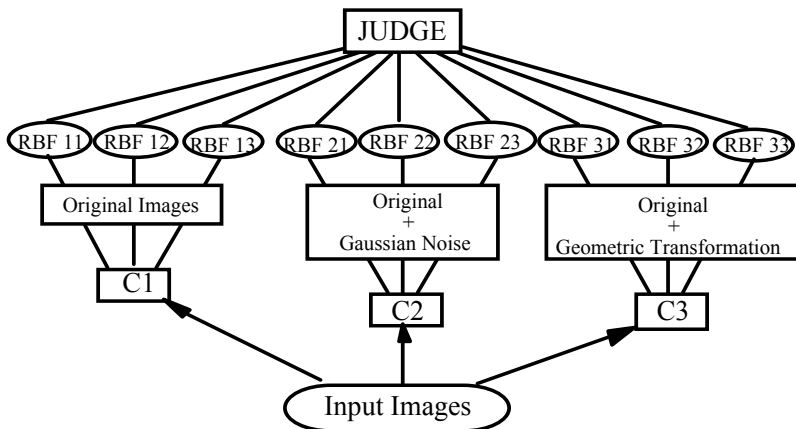
Fig. 1. ERBF Architecture

Fig. 2. Examples of detected frontal faces

Fig. 3. Examples of partial left and right half of the face

Fig. 4. Examples of asymmetric left and right faces

5   Experiments

Extensive experiments using cross-validation were conducted to assess the performance between using asymmetric left vs asymmetric right face images when trained on full-face images. In Section 5.1 we report on the experiments conducted for the recognition of subjects when trained on full-face images and tested on asymmetric left images, while in Section 5.2 we report the results when asymmetric right face images were used.

5.1 Face Recognition Using Asymmetric Left Face

First we report on experiments when only one instance of the subjects facial image was used for training ('Experiment 1') followed by the case in which multiple instances were used for training. The training and testing strategy used in both cases is similar to that of k - fold cross validation (CV) [11]. In k - fold cross validation, the cases are randomly divided into k mutually exclusive partitions of approximately equal size. The cases not found in one partition are used for training, and the resulting classifier is then tested on the partition left out of training. The average error rates over all k partitions are the CV error rate. Each partition in our case consists of images corresponding to 50 unique subjects. Each CV cycle consists of 20 iterations.

Table 1. Average CV results: train on full faces and test on asymmetric left face

Ave. CV Results (%)	RBF Network TP; FN; TN; FP	ERBF TP; FN; TN; FP
Exp1	74; 26; 83; 17	89; 11; 88; 12
Exp2	81; 19; 87; 13	90; 10; 94; 06
Exp3	87; 13; 93; 07	92; 08; 96; 04

Table 2. Average CV results: train on full faces and test on asymmetric right face

Ave. CV Results (%)	RBF Network TP; FN; TN; FP	ERBF TP; FN; TN; FP
Exp1	76; 24; 82; 18	89; 11; 86; 14
Exp2	84; 16; 86; 14	93; 07; 95; 05
Exp3	89; 11; 95; 05	96; 04; 98; 02

As an example, the first CV cycle on its first iteration would randomly pick one image corresponding to each of the 50 subjects ('gallery') in the first partition while testing on the asymmetric left face images of the subjects in the gallery plus the asymmetric left face images ('probes') corresponding to 100 subjects. For each cycle this process is repeated for a total of 20 times. Table 1 shows the average CV results over all cycles and all iterations when the threshold was set at 0.70 when a single RBF network or an ensemble was used. Similarly Table 1 also shows results when the number of images per individual has been increased to 5 ('Experiment 2') and 9 ('Experiment 3'), respectively.

5.2 Face Recognition Using Asymmetric Right Face

The specific training and testing procedure remains the same as that was used in section 5.1. The average CV results over all cycles and iterations when the number of images per individual is 1, 5 and 9 for a single RBF network and an ensemble are shown below in Table 2. The threshold for acceptance/rejection is the same as that was used earlier – 0.70.

6   Conclusions

In many situations, partial faces, due to occlusion or local illumination changes, are all what is sometimes available, for scenarios characteristic of the *Face in a Crowd* challenge. This paper expands on our earlier results on recognizing faces from either their left or right half images as we now train the face recognition engine using full faces and test its recognition ability on asymmetric faces constructed from either their left or right half augmented by its mirror image. We have validated our approach and shown the feasibility on a collection of 3,000 face images corresponding to 150 subjects with $\pm 5^0$ rotation. Cross Validation (CV) results yield an average accuracy rate of (a) 92% when asymmetric left face is used and (b) 96% when asymmetric

right face is used. Based on the experimental results, we believe that in situations where only a partial face image is available, asymmetric faces could be constructed and used for testing against existing face recognition systems that have already been trained on full faces without having to retrain such systems.

Acknowledgements

References

[1]   Gutta, S., Philomin, V., and Trajkovic, M.: An Investigation into the use of Partial Faces for Face Recognition, *5th International Conference on Face and Gesture Recognition (FGR)*. (2002) 33-38, Washington D.C., USA.

[2]   Gutta, S. and Wechsler, H.: Analysis of Partial-Faces for Face Recognition, *10th Int. Conf. on Computer Analysis of Images and Patterns (CAIP)*. (2003) Groningen, The Netherlands.

[3]   Liu, Y., Schmidt, K. L. Cohn, J. F. and Weaver, R. L.: Facial Asymmetry Quantification for Expression Invariant Human Identification, *16th Int. Conf. on Pattern Recognition,(ICPR)* (2002), Quebec-City, Canada.

[4]   Thornhill, R. and Gangestad, S. W.: Facial Attractiveness, *Trans. In Cognitive Sciences*, 3(12) (1999) 452-460.

[5]   Richardson, C. Bowers, K. D. Bauer, R. M. Heilman, K. M. and Leonard, C. M.: Digitizing the Moving face During Dynamic Displays of Emotion, *Neuropsychologia,* 38(7) (2000) 1028-1039.

[6]   O'Toole, A.: The Perception of Face Gender: The Role of Stimulus Structure in Recognition and Classification, *Memory and Cognition*, 26(1) (1998) 146-160.

[7]   Martinez, A. M.: Recognizing Imprecisely Localized, Partially Occluded and Expression Variant Faces from a Single Sample per Class, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(6) (2002) 748-763.

[8]   Lippmann, R.P., Ng, K.: A Comparative Study of the Practical Characteristic of Neural Networks and Pattern Classifiers, MIT Lincoln Labs. Tech. Report 894 (1991).

[9]   Gutta, S., Wechsler, H.: Face Recognition using Hybrid Classifiers, *Int. J. Pattern Recognition*. 30(4) (1997) 539-553.

[10]   Colmenarez, A., Frey, B., Huang, T. S.: Detection and Tracking of Faces and Facial Features, in *Proc. of International Conference on Image Processing*. (1999) 268-272, Kobe, Japan.

[11]   Weiss, S.M., Kulikowski, C.A.: Computer Systems That Learn, Morgan Kaufmann, San Francisco (1991).

Performance versus Computational Complexity Trade-Off in Face Verification

Thirimachos Bourlai, Kieron Messer, and Josef Kittler

University of Surrey, Guildford, Surrey GU2 7XH, United Kingdom
{t.bourlai,k.messer,j.kittler}@surrey.ac.uk

Abstract. We discuss the implementation of a face verification system on smart card and study the trade-off between performance and computational complexity. In order to establish the system limitations, a study was performed on BANCA and XM2VTS databases. The results of the experiments performed show that the choice of a fixed precision data type does not affect system performance very much but can speed up the verification process. Moreover, the use of less than 8 bits per pixel gray-scale image resolution does not necessarily result in a degradation of system performance. We show that, for both databases, image resolution can be reduced without degrading system performance. However, the actual optimal settings will depend on application scenario.

1   Introduction

The design of automatic personal identity authentication systems based on facial images is a very challenging task [11][12] with many promising applications in the field of security. Such biometric systems [5] provide an effective and inherently more reliable way to carry out personal verification of individuals, seeking access to physical or virtual locations. A higher level of security is achieved by exploiting user specific biometric characteristics that it would be difficult to be appropriated by impostors. A secure personal ID system should address government and business policy issues and individual privacy concerns. Smart cards combined with biometric technology [10][8][9] is considered to be a challenging and promising solution in secure personal ID systems[6].

In a conventional face verification system, a database is used to store all biometric templates and users information and the biometric data captured by a camera is transmitted to a central computer where all the image processing and decision making takes place. Although the idea of a centralised online biometric system design is acceptable in many applications, it raises both privacy and security issues. For that reasons, a smart card based face verification system is proposed, demonstrated in *figure 1*. By combining a pin code with on-card storage of the biometric template, we can meet the privacy requirements. Security is also improved since information and processes are protected within the system. Other advantages are improved system performance and system evolvement.

However, with the use of smart cards there are restrictions to be considered like the available CPU computational power, storage capacity and bandwidth[4].
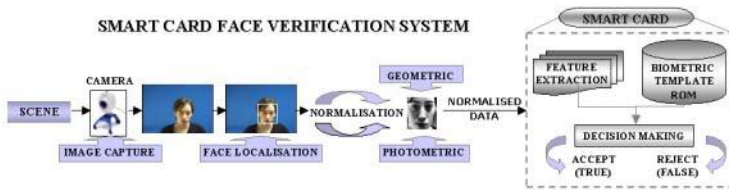
Fig. 1. Proposed smart card face verification system

Therefore, a challenge in the implementation of a smart card face verification system is to optimise the trade-off between the fundamental performance of face verification and the computational complexity of the experimental platform used (server or smart card). Issues to be considered are image quality, system delay, complexity, efficiency, error tolerance (accuracy), compatibility, scalability, memory management, data type dependence and finally, architecture of the system where the code is implemented. The evaluation of such a system is also dependent on the face database used to perform the experiments.

In this paper, we discuss the implementation of a smart card face verification system and we study the trade-off between performance and computational complexity. In order to establish the limitations of such a system, a study was performed, selecting the performance and the computational cost of using different data types on the smart card and on the server, the bit precision of the matching function parameters, the gray-scale resolution of probe images (from 8bpp currently, down to 1bpp) and the image resolution that dominates system speed and memory management.

The rest of the paper is organised as follows. In the next section, the basic face verification process will be covered. In Section 3, the experimental setup and results will be presented. Finally, in Section 4 some conclusions are made.

2 Face Verification System

The face verification method adopted for the implementation on a smart card is the client specific linear discriminant analysis technique, which combines face representation and decision making into a singe step, requiring a template of the size of the input image (see *figure 1*). The overall face verification system involves face detection, photometric normalisation and finally the verification test. All but the last processing step are carried out in the host system. The photometrically normalised image is then transmitted to the smart card where the user biometric template is stored and the verification score computed as well as the final decision taken. The original resolution of the image data in both is 720x576. The experiments were performed with a relatively low resolution face images, namely 55x51. This resolution was used initially as a reference for our study. After detecting the face and localising the eye centres, a simple rotation and scaling of the eye positions onto two fixed points was used for geometric

transformation. The photometric normalisation is achieved by a homomorphic filter and histogram equalisation. For the feature extraction stage, a *PCA* model is built to achieve a dimensionality reduction and then an *LDA* model is produced to get the overall client i specific linear discriminant transformation $\mathbf{a}_i$, which defines the client specific fisher face for testing the claimed identity. The decision making stage produces a score, which defines how close the probe of the claimed identity is to the class of impostors. The thresholds in this stage have been determined based on the equal false rejection (FR) and acceptance (FA) error rates (EER).

The *Client Specific Fisherface* (CSLDA) representation adopted for smart card face verification contrasts with the conventional LDA method, which involves multiple fisherfaces. Apart from its good performance, the method is advantageous in the case of open-set scenarios when new clients are enrolled without system retraining, since it requires only a matrix multiplication of the client mean vector. Moreover, the client enrolment is insulated from the enrolment of other clients. Therefore, it becomes possible to use other than centralised architecture for the face verification system and the smart card processing becomes a reality without any need to restrict the representation framework, and as a result the representation capacity of the system. Finally, the speed of probe testing is more than two orders of magnitude faster than that achieved by the PCA and LDA methods[2], as the *CSLDA* method involves only a single fisher face $\mathbf{a}_i$ per client stored on the smart card.

In the verification stage we use the metric (distance) d_c:

$$d_c = \left| \mathbf{a}_i^T \mathbf{z} - \mathbf{a}_i^T \mu_i \right| \tag{1}$$

where $\mathbf{z}$ is the probe image and μ_i is the client mean.
Now, if $d_c \leq t_c$, then we accept the claim, where t_c is a pre-computed threshold.

3   Experimental Setup and Results

For the purpose of this study, BANCA (open-set protocol)[1] and XM2VTS (closed-set protocol)[7] face databases were used in the experiments. We are dealing with single-modality experiments. From the sets containing the face images, training, evaluation and test set is built. The training set is used to construct client models; the evaluation set produces client and impostor access scores (used to compute a client-specific or global threshold that determines acceptance or rejection of a person); and the test set is selected to simulate realistic authentication tests where impostor's identity is unknown to the system. The threshold is set to satisfy certain performance levels on the evaluation set. Finally, the performance measures of the verification system are the FA and FR rates on the test set.

The smart card used in these experiments was provided by Sharp Laboratories, Oxford, UK. It boasts a 13.5MHz processor, 1Mbyte of EEPROM, 8KBytes of RAM, a cryptographic co-processor, does not have a floating point

co-processor and operates in both contact and contactless modes. In these experiments we used it in contact mode which has a data transfer rate of 115.2Kbits per second. The transmission rates of data between the server and card is fairly slow. Therefore, the amount of data being sent to and read from the smart card (e.g. a biometric template or a facial image) must be kept to a minimum. Finally, the amount of RAM available on the smart card is limited, which means all the data can not be kept in memory for the calculations and the ROM must be used as a cache.

In the *first experiment*, the relationship between computational cost of verification on the smart card and the use of different data types was examined. We measured the time required to perform certain on-card extended data operations. These experiments demonstrated that the use of integers instead of floating point numbers can speed up the overall verification performance by over a factor of 6. The reason is the non-availability of a floating point co-processor on the card and the use of a simulated floating point unit instead. Fixed point numbers (FPN) proved to be a risky choice when used exclusively and the choice of double precision on the card was absolutely prohibiting in terms of computational costs. In order to perform the same experiment on the server, a specific fixed point library was built. Interestingly, the use of the generated n-bit precision data type instead of integers resulted in only 12% loss in the overall computational cost. Ideally, the use of integers in both the server and the smart card would be the best solution in terms of speed but not in terms of precision. However, the use of fixed n-bit precision data type on the server and the use of integers on the smart card is expected to significantly increase the overall speed when on-card verification is performed.

In the *second experiment*, we investigated the trade-off between performance and n-bit precision for the verification function parameters when using fixed point arithmetic for authentication. These parameters are the client specific LDA transformation matrix $\mathbf{a}$, the client mean vector μ_i and the global mean $\mu = \sum_{j=1}^{N} \mathbf{z}_i$, where N is the size of the training set and $\mathbf{z}_i$ are the training images. The basic idea behind that was to change the precision of the CSLDA transformation that is actually sent on the smart card for on-card verification based on the distance metric given in equation (1). In order to evaluate the system performance the following formula was computed:

$$PPD = \sum_{i=1}^{2} \frac{\left| \frac{FA\_Gi_{init} - FA\_Gi_{new}}{FA\_Gi_{init}} \right| + \left| \frac{FR\_Gi_{init} - FR\_Gi_{new}}{FR\_Gi_{init}} \right|}{4} * 100\% \qquad (2)$$

where *PPD* stands for *Percentage of Precision Degradation* and G_1, G_2 stand for the two groups we are using for training and testing alternatively. Within the formula, each percentage represents the absolute percentage difference between the original value of the (false acceptance)/(false rejection) rate and the new (false acceptance)/(false rejection) rate after applying the n-bit precision transformation. However, Half Total Error Rate (HTER) was also used as the basic metric of performance. In *table 1* we can see the best selected n-bit precision in terms of *PPD* when applying this to all BANCA protocols.

Table 1. Best cases by using n-bit precision in BANCA

Protocol	*n-bit* precision	HTER init	HTER new	PPD
MC	12,13	0.07932	0.07932	0
MD	13	0.10737	0.10737	0
MA	13	0.09775	0.09775	0
UD	10	0.1915	0.19166	0.68
UA	13	0.26634	0.26682	0.2232
P	13	0.21927	0.2196	0.2246
G	10	0.07927	0.07879	1.03756
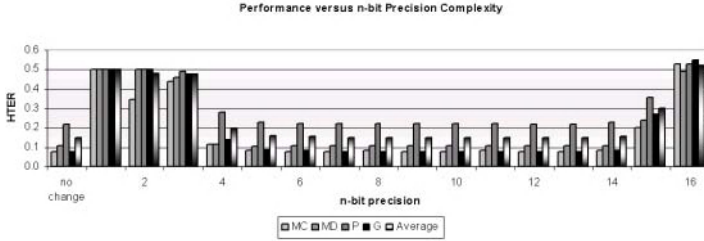
Fig. 2. Graphical representation of the effects of n-bit precision on the performance in MC, MD, P, G and average BANCA protocols
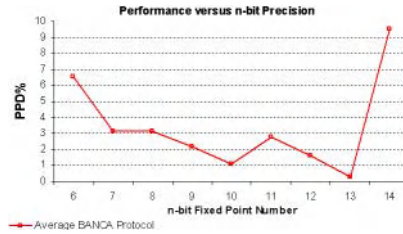
Fig. 3. Average BANCA protocol performance when using n-bit fixed point numbers ($n \in [6,..,14]$)

Moreover, in *figure 2* we can see the overall results in terms of performance and n-bit precision in some selected BANCA protocols and the average results in all protocols. An interesting observation is that the use of a n-bit precision data type above 13 can result in an increased loss of performance due to quantisation errors. By trying to optimise the results for all cases and to find the best value of n for all BANCA protocols in terms of the minimum *average PPD*, we conclude that by using *13-bit precision* we achieve only a 0.32% loss in overall system performance. The overall average results are graphically represented in *figure 3*.

In the *third experiment*, we investigated the effect on performance by altering the gray-scale pixel resolution of the (55x51) normalised probe images in the

Table 2. BANCA protocol performance results using different gray-scale pixel resolution

n-bit	MC	MD	MA	UD	UA	P	G	Average Protocol PPD
8bpp	0.07932	0.1073	0.09775	0.1915	0.2663	0.2192	0.0792	0.1487
7bpp	0.07932	0.1073	0.09935	0.1911	0.2665	0.2194	0.0792	0.1489
6bpp	0.08028	0.1078	0.09935	0.1915	0.2653	0.2204	0.0792	0.1491
5bpp	0.07884	0.1062	0.09759	0.1889	0.2639	0.2211	0.0785	0.1478
4bpp	0.07788	0.1041	0.09647	0.1915	0.2642	0.2205	0.0772	0.1474
3bpp	0.07628	0.1083	0.09727	0.1903	0.2663	0.2195	0.0808	0.1484
2bpp	0.07003	0.1169	0.10224	0.2139	0.2838	0.2322	0.0864	0.1579
1bpp	0.0915	0.2355	0.14033	0.3153	0.3685	0.3082	0.1637	0.2319

training set. Since an 8 bit camera is used, the initial performance of our system was measured by using 8 bits per pixel for each face image. Then, the gray-scale resolution was reduced by a factor of 1bpp each time before building the PCA and LDA model. All seven cases of BANCA protocols were tested and the results in all protocols are given in *table 2*. One interesting observation is that the use of 4-bit gray-scale pixel resolution yields an improved overall performance by 0.95% compared to the initial reference images. This can be explained by the fact that pixel resolution normalisation introduces quantisation errors that can affect the computation of the error rates and thresholds. Since we know that in verification systems there is always a trade-off between FA and FR rates, there are cases where these rates can change in such a way that, on average, the overall verification performance is positively affected (that is, for example, the FA rate degrades while FR improves even faster than FA degrades).

In the *last experiment*, we investigated the trade-off between performance and image resolution that can be exploited to minimise the computational costs in both BANCA and XM2VTS databases. In this study, the initial raw face images were geometrically and photometrically normalised to a predefined size. As a reference, image resolution 55x51 was selected. In *figure 4* the experimental results are presented. In the top row, we can see the detailed results in both databases, clearly for all protocols and configurations. In XM2VTS (*figure 4a*), better overall results are achieved and particularly configuration II seems to achieve a more desirable demonstration of system performance. This is probably because the training set in configuration II is larger than that in configuration I. In the case of BANCA (*figure 4b*), the overall trend of the results does not change, although the overall results are worse than those of XM2VTS. In the bottom row, a representation of the average results obtained on each database are shown (see *table 3*). In (*figure 4c*) the average results on XM2VTS are presented. Note that going from 55x51 to 20x18 we achieve an overall 0.65% improvement in performance while at the same time speed is increased 87,6%. However, the best average performance is achieved by using 30x28 resolution where we achieve 5.8% improvement in performance while at the same time the speed is increased 71.1%. In comparison, in *figure 4d* the average results on BANCA are presented.
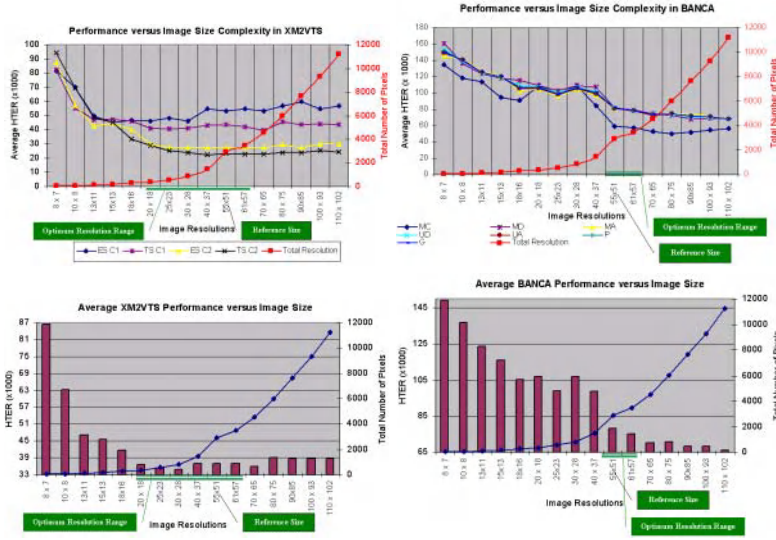
Fig. 4. The effect on performance in terms of *HTER* by using different image resolution for probe images in the evaluation/test set (in both configurations) of XM2VTS (a) and in all protocols in BANCA (b). The same effect is presented when we average the results in XM2VTS (c) and in BANCA (d)

However, in this case, in order to improve the system performance at 17.8%, we have to tolerate that the speed will be halved. This major difference in the two databases is due to their different protocol configuration; Basically, in XM2VTS we have more controlled conditions and fully frontal faces, whereas in BANCA different condition scenarios are presented. It seems that closed-set protocols perform better and a redesign of the verification system when new clients are added in an open-set protocol will result in better results.

4   Discussion and Conclusions

Optimisation of a smart card face verification system design in terms of performance and computational complexity is a very complex process, with many key factors to consider. The specification includes gray-level and spatial image resolution, the overall system speed and performance requirements. The number of bits to be transmitted from the server to the smart card may have to be further restricted when fusion methods need to be incorporated onto the smart card [3] and therefore an increased number of biometric templates have to be stored on the card.

Experiments showed that the use of fixed point arithmetic can speed up the template matching on the card. However, the ideal solution is to use integers both on the card and the server. Using less than 8bpp grey-scale image resolution for the normalised face images does not necessarily result in a degradation of the

Table 3. Overall average results in both BANCA and XM2VTS

Image Resolution R	Total no of Pixels	BANCA Average HTER	XM2VTS Average HTER
8 x 7	56	149.33	86.49
10 x 8	80	136.67	63.09
13x11	143	123.57	47.14
15x13	195	116.02	45.46
18x16	288	105.31	41.68
20 x 18	360	106.92	36.64
25x23	575	98.96	35.44
30 x 28	840	106.92	34.75
40 x 37	1480	98.68	36.95
55x51	2907	78.24	36.88
61x57	3477	75.33	36.87
70 x 65	4550	70.48	35.84
80 x 75	6000	70.75	38.93
90x85	7650	68.47	38.73
100 x 93	9300	68.38	38.70
110 x 102	11220	66.40	38.64

system performance. This allows for fewer bytes of data to be sent to the smart card. There is a trade-off between performance and image resolution range that dominates system speed and memory management. In both databases we can improve the system performance when decreasing the gray-scale resolution, but not when decreasing image size. Different results were achieved on the BANCA in contrast to XM2VTS database suggesting that the use of different operating scenarios for system evaluation may call for different optimum operating point.

Acknowledgements

References

[1] E. Bailly-Baillire, S. Bengio, F. Bimbot, M. Hamouz, J. Kittler, J. Mariethoz, J. Matas, K. Messer, V. Popovici, F. Poree, B. Ruiz, and J.-Ph Thiran, 'The banca database and evaluation protocol', *AVBRA*, (2003). 171

[2] P. N. Belhumeur, J. Hespanha, and D. J. Kriegman, 'Eigenfaces vs. fisherfaces: Recognition using class specific linear projection', *IEEE PAMI*, **19**, 45–58, (1996). 171

[3] J. Czyz, S. Bengio, C. Marchel, and L. Vanderdorpe, 'Scalability analysis of audio-visual person identity verification', *AVBRA*, 752–760, (June 2003). 175

[4] Jacek Czyz and Luc Vandendorpe, 'Evaluation of lda-based face verification with respects to available computational resources', in *PRIS*, (April 2002). 169

[5] J. L. Dugelay, J. C. Junqua, C. Kotropoulos, R. Kuhn, F. Perronnin, and I. Pitas, 'Recent advances in biometric person authentication', *ICASSP (special session on biometrics), Orlando, Florida*, (May 2002). 169

[6] Copyright Smart Card Alliance Inc., 'Smart card and biometrics in privacy-sensitive secure personal identification systems', *A Smart Card Alliance white paper*, `http://www.datakey.com/resource/whitePapers/Smart_Card_Biometric_paper.pdf`, (May 2002). 169

[7] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre, 'Xm2vtsdb: The extended m2vts database', *AVBRA*, 72–77, (March 1999). 171

[8] M. Osborne and N. K. Ratha, 'A jc-bioapi compliant smart card with biometrics for secure access control', *AVBRA*, 903–910, (June 2003). 169

[9] W. Rankl and W. Effing, 'Smart card handbook', *John Wiley & Sons*, (2000). 169

[10] R. Sanchez-Reillo and C. Sanchez-Avila, 'Fingerprint verification using smart cards for access control systems', *IEEE AESM*, **17**(9), 12–15, (2002). 169

[11] M. Turk and A. Pentland, 'Eigenfaces for recognition', *Cognitive Neuroscience IEEE PAMI*, **3**(1), 71–86, (1991). 169

[12] W. Zhao, R. Chellappa, A. Rosenfeld, and P. Phillips, 'Face recognition: A literature survey', *UMD CfAR Technical Report CAR-TR-948*, (2000). 169

An Application of Fractal Image-Set Coding in Facial Recognition

Hossein Ebrahimpour, Vinod Chandran, and Sridha Sridharan

Queensland University of Technology, Brisbane, Australia
hossein@ieee.org
{v.chandran,s.sridharan}@qut.edu.au

Abstract. Faces are complex patterns that often differ in only subtle ways. Face recognition algorithms have difficulty in coping with differences in lighting, cameras, pose, expression, etc. We propose a novel approach for facial recognition based on a new feature extraction method called fractal image-set encoding. This feature extraction method is a specialized fractal image coding technique that makes fractal codes more suitable for object and face recognition. A fractal code of a grayscale image can be divided in two parts – geometrical parameters and luminance parameters. We show that fractal codes for an image are not unique and that we can change the set of fractal parameters without significant change in the quality of the reconstructed image. Fractal image-set coding keeps geometrical parameters the same for all images in the database. Differences between images are captured in the nongeometrical or luminance parameters - which are faster to compute. Results on a subset of the XM2VTS database are presented.

1 Introduction

A compact representation of fractal encoding and decoding process can be provided by using these notations [1] : Let $\Im^m$ denote the space of $m \times m$ digital grayscale images, that is, each element of $\Im^m$ is a $m \times m$ matrix of grayscale values. The get-block operator $\Gamma_{n,m}^k : \Im^N \to \Im^k$, where $k \leq N$, is the operator that extract the $k \times k$ block with lower corner at n, m from the original $N \times N$ image. The put-block operator $(\Gamma_{n,m}^k)^* : \Im^k \to \Im^N$ inserts a $k \times k$ image block into a $N \times N$ zero image, at the location with lower left corner at n, m . A $N \times N$ image $x_f \in \Im^N$ can be shown as

$$x_f = \sum_{i=1}^{M} (x_f)_i = \sum_{i=1}^{M} (\Gamma_{n_i,m_i}^{r_i})^* (R_i) \tag{1}$$

that $\{R_1, \ldots, R_M\}$ are a collection of range cell images that partition x_f . Each R_i has dimension $r_i \times r_i$ with lower corner located at n_i, m_i in x_f. If the range cells R_i are the result of fractal image encoding of the image x_f, then for each range cell R_i there is a domain cell D_i and an affine transformation W_i such that

$$R_i = W_i(D_i) = G_i(D_i) + H_i \tag{2}$$

Denote the dimension of D_i by d_i, and denote the lower left coordinates of D_i by k_i, l_i . $G_i = \Im_i^d \to \Im_i^r$ is the operator that shrinks (assuming $d_i > r_i$), translates $(k_i, l_i) \to (n_i, m_i)$ and applies a contrast factor s_i, while H_i is a constant $r_i \times r_i$ matrix that represents the brightness offset. We can write $D_i = \Gamma_{k_i,l_i}^{d_i}(x_f)$ Thus, (1) can be rewritten as the following approximation:

$$x_f = \sum_{i=1}^{M} (\Gamma_{n_i,m_i}^{r_i})^* \{ G_i(\Gamma_{k_i,l_i}^{d_i}(x_f)) + H_i \}$$

$$x_f = \underbrace{\sum_{i=1}^{M} (\Gamma_{n_i,m_i}^{r_i})^* \{ G_i(\Gamma_{k_i,l_i}^{d_i}(x_f)) \}}_{A(x_f)} + \underbrace{\sum_{i=1}^{M} (\Gamma_{n_i,m_i}^{r_i})^* (H_i)}_{B} \qquad (3)$$

Then if we write the Get-Block operator $(\Gamma_{n,m}^{k})^*$ and Put-Block operator $\Gamma_{k_i,l_i}^{d_i}$ and transformations G_i in their matrix form we can simplify equation (3) as

$$x_f = A \times x_f + B \qquad (4)$$

In this equation A, B are fractal parameters of image x_f.

2   Fractal Image-Set Coding

In this section we will use the compact representation (4) to show some interesting properties of fractal image encoding and introducing a method for extracting fractal codes for a set of face images with the same geometrical parameters which we will call *Fractal Image-set Coding*. The fundamental principle of fractal image encoding is to represent an image by a set of affine transformations. Images are represented in this framework by viewing them as vectors. This encoding is not simple because there is no known algorithm for constructing the transforms with the smallest possible distance between the image to be encoded and the corresponding fixed point of the transformations. Banach's fixed point theorem guarantees that, within a complete metric space, the unique fixed point of a contractive transformation may be recovered by iterated application thereof to an arbitrary initial element of that space. The Banach's fixed point theorem gives us an idea of how the decoding process works:

Let $T : \Im^n \to \Im^n$ be a contractive transformation and $(\Im^n, d)$ a metric space with metric d then the sequence of $\{X_k\}$ constructed by $X_{k+1} = T(X_k)$ converge for any arbitrary initial image $X_0 \in \Im^n$ to the unique fixed point $X_f \in \Im^n$ of the transformation T. The contraction condition in this theorem is defined by this definition: Transformation $T : \Im^n \to \Im^n$ is called contractive if there exists a constant $0 < s < 1$, such that

$$\forall x, y \in \Im^n, d(T(x), T(y)) \le s.d(x, y) \qquad (5)$$

This condition is a sufficient condition for existence of a unique fixed point for fractal transformations. Let us show the fractal transformation in compact
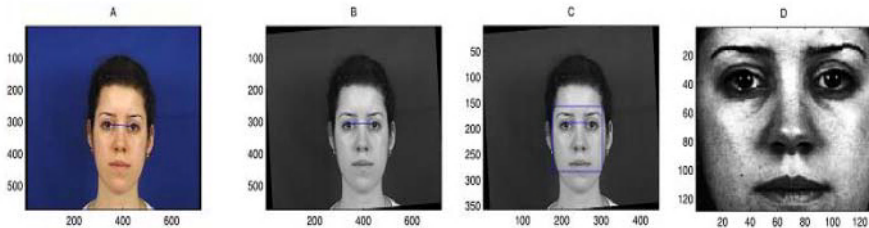
Fig. 1. An example of preprocessing with an image in the data-set.(A) The original image, (B) grayscale image with orientation normalized, (C) Nominated image with face area marked, (D) normalized and histogram equalized face image

form (4) so the fractal image coding of an image x_f can be defined by finding A, B to satisfy this condition, while A, B define a contractive transformation: $x_f = A \times x_f + B$ This condition shows that the fractal code for an image x_f is not unique because we can have infinite pairs of A, B to satisfy that condition, and have the same fixed point x_f. In many of them, A and B define a contractive transformation $T(x)$ with $|s| < 1$.

Different fractal coding algorithms, use different A and B for an image that makes the fractal face recognition process more complex. The aim of Fractal Image-set coding is to find fractal parameters for several images with the same geometrical part for all of them. In this case, the information in the luminance part of fractal codes of these images is more comparable. This method is also more efficient and faster than existing methods because there is no need to search for the best matching domain block for any range block which is the most computationally expensive part of the traditional fractal coding process.

In our algorithm, a sample image is nominated for using to find the geometrical parameters. This image can be an arbitrary image of database, an image out of the database or the average image of all or part of the database. The *Fractal image-set coding* algorithm can be described as follow:

Step 0 (preprocessing) -For any face image data-set F use eye locations and histogram equalization to form a normalized face image data-set F_{nor}. Any face image in this data-set is a 128×128, histogram equalized 256-grayscale image, with the position of left and right eye at (32,32) and (96,32) respectively as shown in Figure 1.

Step 1 - Calculate the fractal codes for the sample image $\overline{x}$ (that can be the average image of data-set) using traditional fractal image coding algorithms [2]. These fractal codes contain the luminance information and geometrical position information for range blocks $\{R_1, R_2, \ldots, R_n\}$, the domain block $\{D_1, D_2, \ldots, D_m\}$ corresponding to each range block and the geometrical transformations like rotation and resizing to match the domain block with the range block.

Step 2 - For any image x_i of the data-set, use the same geometrical parameters (range and domain blocks positions and geometrical transformations) that
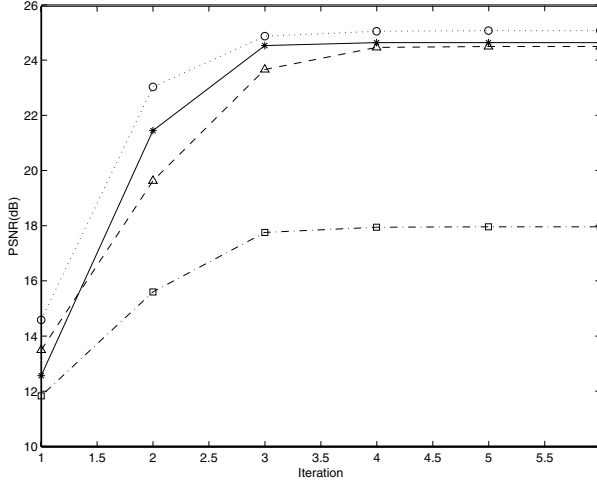
Fig. 2. The PSNR versus the number of decoding steps for 128×128 gray-scale, normalized, encoded images of the XM2VTS database. The dash-dot line, solid line, dashed line and dotted line correspond to images 002_1_1, 000_1_1, $003\_1\_1$and 005_4_1 images, respectively

used for coding the sample image $\overline{x}$. Let (x_{R_i}, y_{R_i}), l_{R_i} be the geometrical position and the block size of Range block R_i and (x_{D_j}, y_{D_j}), l_{D_j} the geometrical position and the size of domain block D_j which is the best matched domain block for R_i.

Step 3 - For any image range block R_i in image x use the domain block at the same position (x_{D_j}, y_{D_j}) and the same size l_{D_j} and calculate the luminance parameters s and o to minimize the error e [2]: $e = \sum_{i=1}^{n} (s.d_i + o - r_i)^2$.

Step 4 - Save the geometrical parameters as well as luminance parameters as fractal codes of image x.

The PSNR versus iteration characteristic is drawn in figure 2 for four images of XM2VTS database. It is clearly shown that the fixed point of each fractal image codes can be reached after only 5 or 6 iterations.

Each image can be represented as a point in image space, R^N, where N is the number of pixels. For the purpose of illustrating convergence and distances in feature space, we will use a two-dimensional image space, $X = [x, y]$. Fractal code parameters can then be represented using matrix A and vector B in the transformation $F(X) = A \times X + B$. We choose different initial images and different fractal parameters to show how distances in image space can be used for classification. When the fractal code image x_f is applied iteratively to an initial image, say x_{01} the image converges to x_f after several iterations. We want to find that image in the database which is closest to x_f. If a Euclidean distance based on grayscale differences between corresponding pixels is used, the distance between a database image and the query image is not very reliable
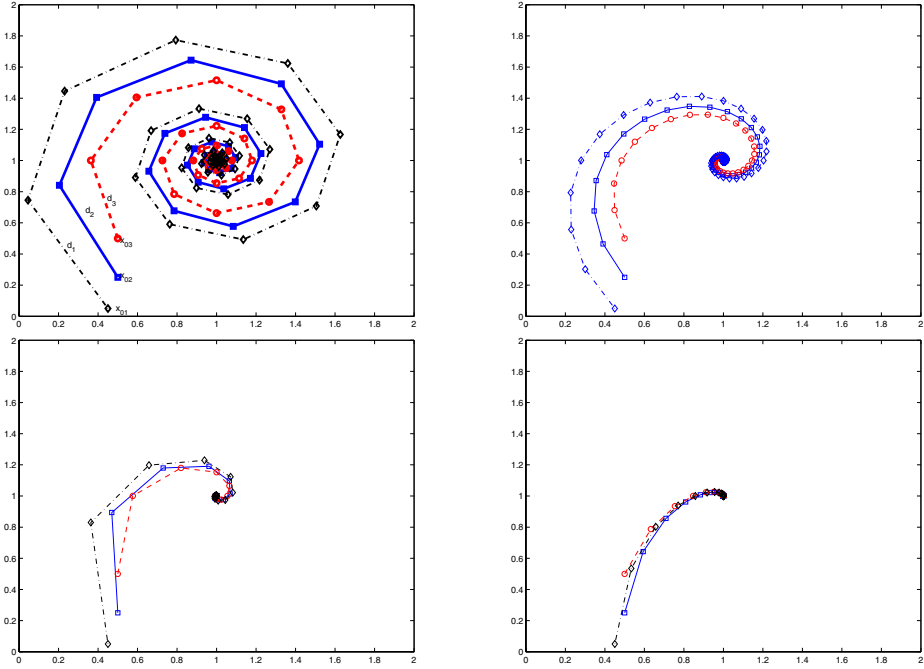
Fig. 3. Top left: Convergence trajectories for three different initial images when the same fractal code is applied iteratively. Note that the initial image (x_{03}) closest to the fixed point shows the least distance between successive iterations ($d_3 < d_2 < d_1$). The fractal parameters are $A = 0.9 \times \rho_{45}$ and $B = (I - 0.9 \times \rho_{45}) \times x_f$. Bottom Left: $A = 0.9 \times \rho_{15}$ and $B = (I - 0.9 \times \rho_{15}) \times x_f$. Top right: $A = 0.6 \times \rho_{45}$ and $B = (I - 0.6 \times \rho_{45}) \times x_f$. Bottom right: $A = 0.6 \times \rho_{15}$ and $B = (I - 0.6 \times \rho_{15}) \times x_f$

because it can change considerably with noise and with small changes to the query image. For example, a small misalignment between the images can cause large differences between grayscale values of corresponding pixels. Therefore, a more robust distance measure is required. One such distance is that between two successive iterations of an image when the fractal code for x_f is applied to it. Figure 3(top left) shows the trajectories in image space as images x_{01}, x_{02} and x_{03} converge towards x_f for the simplified two-dimensional case. It can be observed that image x_{03} is closest to x_f and d_3 is also the shortest of d_1, d_2 and d_3. This relationship holds regardless of fractal parameters A and B. Fractal parameter A can be decomposed into a rotation matrix ρ_θ and a scale factor s. Figure 3(bottom left) shows convergence trajectories for the same images when $A = 0.9 \times \rho_{15}$ and $B = (I - 0.9 \times \rho_{15}) \times x_f$. Figures 3(top left) and 3(bottom left) correspond to a scale factor $s = 0.9$ but different rotation matrices. Figures 3(top right) and 3(bottom right) are the same plots with the same rotation matrices as 3(top left) and 3(bottom left), respectively, but with the scale factor changed to $s = 0.6$. Note the faster convergence for the lower value of s. We can use

Table 1. Accuracy versus the number of persons in the database

Number of persons	5	10	15	20	25	30	35	38
%Accuracy	100.00	100.00	93.33	85.00	76.00	76.67	71.43	71.09

distance d between an image and its first iterate when the fractal code of x_f is applied to it as a measure of the distance of this image to x_f. All images in the database are subjected to this transformation and distances are compared. The image with the least distance is used to identify the person (there may be more than one image of the same person in the database used for training).

3   Using Fractal Image-Set Coding for Face Recognition

The fractal codes extracted by this method for a face data-set F have the advantage that all the codes have the same geometrical parameters and therefore the luminance parameters are more comparable than the traditional fractal codes. For face recognition applications, we can divide the image database to two image-sets, a training set and a test set.The sample image $\overline{x}$ can be the average image of the test set or the training set or the entire database. These cases may be suitable to face recognition from a closed set or open set. The results have been found not to change much with the choice of the image set from which the geometrical parameters are extracted.

4   Experimental Results

We selected the first 38 individuals from the XM2VTS database and 4 face images per individual. The first image was used as a test image while the other images were added to the training set. Eye location information available from the database is used to normalize and align the images to a 128×128 pixel grid. The eye coordinates are now fixed and 64 pixels apart. The average image $\overline{x}$ over the entire data-set is calculated and used to extract the shared geometrical fractal parameters.

The performance of our system is illustrated through some test cases in figures 4, 5 and 6. Recognition accuracy (Table 1) was 100% for up to 14 persons and falls to about 70% for 38 persons.

The fractal code of a query or test image is applied to all the images in the training set for one iteration. The distance, d, between each of these transformed images and the correspondent initial (target) image is used as a measure of distance between the test image and the target image. A similarity score, e^{-d}, which is bounded between 0 and 1, indicates the closeness of the match between the target image and the test image. The target that has the highest similarity score is the recognized identity. The next 5 best matches reveal the effectiveness of the method as explained in the figure captions.
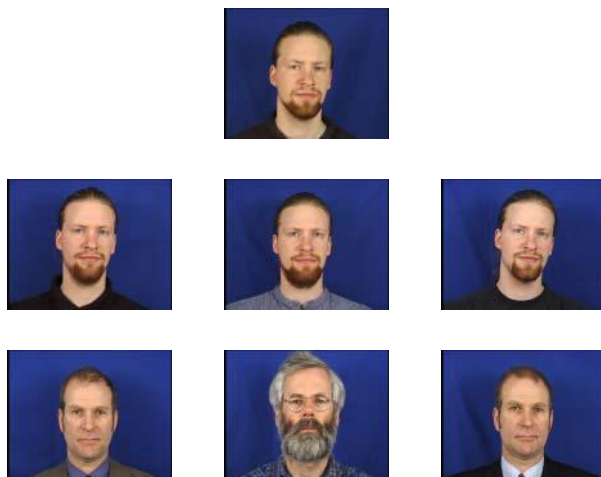
Fig. 4. An example showing a query image on top followed by the six closest images in the database. The best match is on the top-left, followed by others left to right in row-first order. All the closest matched images in this case share either a beard or similar hair with the person in the query image. Also the query image and all the 6 closest matched images are light-skinned individuals
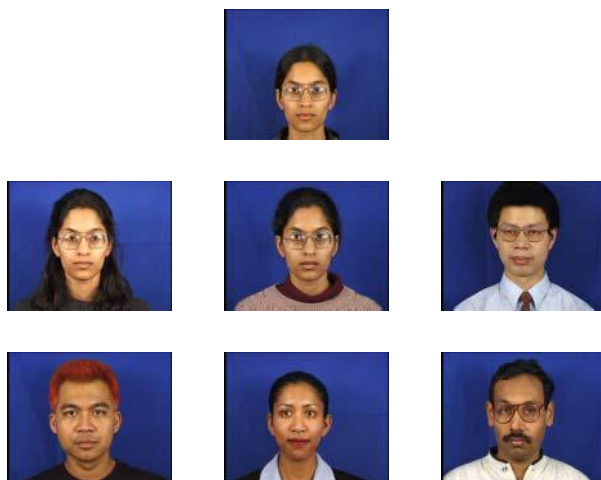
Fig. 5. Yet another correctly identified case. Note again that 5 out of the 6 close matches are images of persons with similar skin tone. The other image in this case is of a person with similar spectacles

The biggest advantage that this method offers is that effects of changes in parts of a face are confined by the geometrical parameters used and by the number of iterations. Since the parameters are common to all codes, we can choose

Fig. 6. A test case that failed. The second closest match is of the correct individual and so is the fourth closest. The closest matched face is of a person with some resemblance to the correct individual who has a similar gaze in this image. The training images of the correct individual (second and fourth) are quite different in eye closure and gaze

to emphasize or deemphasize certain regions, thereby achieving robustness to the presence of spectacles or expression changes. This may not be possible easily in other non-fractal methods where a change in one part of a face affects all features (such as in the eigenface approach without part segmentation) or fractal methods where geometrical fractal code parameters vary from image to image. Experiments show that the recognition accuracy is not compromised in the process.

5   Summary and Conclusions

A new method of using fractal codes for facial recognition is presented. It is shown that fractal codes are not unique and that certain parameters can be held constant to capture image information in the other parameters. Fractal codes are calculated keeping geometrical fractal parameters constant for all images. These parameters are calculated from a set of images. The proposed method is faster than traditional fractal coding methods which require time to search and find the best domain for any range block. It also lends itself to preprocessing steps that provide robustness to changes in parts of a face and produces codes that are more directly comparable. Results on the XM2VTS database are used to demonstrate the performance and capabilities of the method.

Acknowledgements

References

[1] G. Davis,"Implicit Image Models in Fractal Image Compression",*SPIE Conference on Wavelet Applications in Signal and Image Processing IV*, 1996   178
[2] Y. Fisher,"Fractal Image Compression: Theory and Application," *Springer-Verlag Inc.,* 1995.   180, 181

Corrupted Face Image Authentication Based on Noise Model

Ho-Choul Jung[1], Bon-Woo Hwang[2], Sang-Woong Lee[1], and Seong-Whan Lee[1]

[1] Center for Artificial Vision Research
Department of Computer Science and Engineering
Korea University
Anam-dong, Seongbuk-gu, Seoul 136-701, Korea
{hcjung,sangwlee,swlee}@image.korea.ac.kr
[2] VirtualMedia, Inc.
#316 Korea Techno Complex
126-16 Anam-dong, Seongbuk-gu, Seoul 136-701, Korea
bwhwang@virtualmedia.co.kr

Abstract. In this paper, we propose a method for authenticating corrupted face images based on noise model. The proposed method first generates corrupted images by controlling noise parameters in the training phase. The corrupted image and noise parameters are represented by a linear combination of prototypes of the corrupted images and the noise parameters. With the corrupted image, we can estimate noise parameters of the corrupted image in the testing phase. Then, we can make a synthesized face image from the original face image with the estimated noise parameters and verify it with the corrupted face image. Our experimental results show that the proposed method can estimate noise parameters accurately and improve the performance of face authentication.

1 Introduction

Recently, face authentication market is expanding and many fields such as smart card authentication system and biometric passport system, etc, are requiring face authentication technique for security reasons. Especially, face image authentication is to verify an original face photograph in database or chip in identification card, passport or smart card and a face photograph scanned from them. In the authentication process, we are faced on some problems such as scratch, blur and discoloration appearing in private photographs (Fig. 1). Additionally, most of approaches for face authentication require that there are at least two training images per person in order to obtain good performance. Unfortunately, in real-world tasks such a requirement can not always be satisfied.

In particular, several researchers recently have studied for dealing with corrupted images. Sanderson et al. proposed a method to extract robust features in difficult image conditions [5]. This method is based on the robust feature extraction against Gaussian white noise and Gaussian illumination changes. However, this method has no consideration of important features required for authentication in images corrupted by scratch, blur and discoloration.

Fig. 1. The example of corrupted face images scanned from identification card

Also, previous researches have been focused on the training algorithms using multiple training images per person [1]. In many applications, though, only one sample per person is available to the face authentication system [2]. Because of these problems, face recognition using virtual models has been studied. Beymer et al. proposed pose invariant face recognition method based on virtual views [4]. This method performs face recognition by synthesizing various virtual views from one example view. This method has a good performance against various pose but has a difficulty with virtual view generation against occluded region.

In this paper, in order to solve the above two problems: the corruption of face images and the number of training image per person, we propose an face image authentication method based on noise model.

This paper is organized as follows: In Section 2 and 3, we define noise model and describe the method for noise parameter estimation and face image authentication, respectively. Experimental results of face authentication are given in Section 4. Finally, we present conclusive remarks in Section 5.

2   Noise Model

2.1   Analysis of Noise in Corrupted Images

In this paper, we assume that corrupted images are generated from the change of contrast and brightness and Gaussian blur. First, we define an image of which the contrast and brightness are changed as follows:

$$I_{CB}(x, y) = c \times I_{org}(x, y) + b \tag{1}$$

where I_{CB} is the image corrupted by the change of contrast and brightness, I_{org} is the original image, c is the contrast parameter, b is the brightness parameter. Second, we define corrupted images using Gaussian blur like (2).

$$I_G(x, y) = I_{org}(x, y) \cdot G_{blur}(x, y) \tag{2}$$

where G_{blur} is the Gaussian blur filter, G is the function of Gaussian blur and $\cdot$ is the image convolution operator.

$$G_{blur}(x, y) = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}} \tag{3}$$

where σ in (3) is the standard deviation of Gaussian function as the parameter of Gaussian blur.

2.2 Definition of Noise Model

In this section, we will formally specify noise model. Let us corrupted image, I^c be as follows:

$$I^c = I_{CBG} = I_{CB} \cdot G_{blur} \tag{4}$$

Then, more formally the noise model is defined as the set of corrupted images I^c and noise parameter P.

$$N_i = \begin{pmatrix} I_i^c \\ P_i \end{pmatrix}, \quad (i = 1, \cdots, m) \tag{5}$$

where $I^c = (x_1, \cdots, x_k)^T$, $P = (p_1, \cdots, p_l)^T$, $x_1, \cdots, x_k$ are intensity of pixels in the corrupted image, k is the number of pixels in the corrupted image, p is the noise parameter value, l is the number of used noise parameters and m is the number of corrupted image. In this paper, we define $l = 3$, $p_1 = c$, $p_2 = b$, $p_3 = \sigma$ and $P = (c, b, \sigma)^T$ since we consider the change of contrast and brightness and Gaussian blur. Thus, N is represented as follows:

$$N = \bar{N} + \sum_{i=1}^{m-1} \alpha_i n_i(j), \quad (j = 1, \cdots, k, k+1, \cdots, k+l) \tag{6}$$

where $\bar{N}$ is the mean of $N_i(i = 1, \cdots, m)$. By PCA, a basis transformation is performed to an orthogonal coordinate system formed by eigenvector n_i of the covariance matrices on data set of m corrupted images and noise parameters.

3 Face Image Authentication

3.1 Overview

In order to authenticate corrupted face images, our method includes the following two phases. In the training phase, we first generate corrupted images by controlling the parameters of the contrast and brightness and the parameter of Gaussian blur from an original face image. Then, we get basis vectors of corrupted images and noise parameters. In the testing phase, the face authentication procedure of corrupted face image is as follows:

- Step 1. Compute the linear coefficients of the sub-matrix of eigenvectors corresponding to corrupted images by least-square minimization method (LSM).
- Step 2. Estimate noise parameters of the sub-matrix of eigenvectors corresponding to the noise parameters by using the obtained linear coefficients.
- Step 3. Synthesize a corrupted face image by applying the estimated noise parameters to original face image.
- Step 4. Perform the face authentication between the test image and the synthesized face image in Step 3.

3.2   Noise Parameter Estimation

With noise model, only approximation of the required parameters is possibly obtained. Our goal is to estimate noise parameters by finding an optimal solution in such an overdetermined condition. At first, we want to find α which will satisfy (7).

$$\tilde{N}(j) = \sum_{i=1}^{m-1} \alpha_i n_i(j), \quad (j = 1, \cdots, k) \tag{7}$$

where j is the intensity of pixels in the corrupted image, k is the number of pixels in the corrupted image and the difference image is defined as $\tilde{N} = N - \bar{N}$. Generally, there may not exist a choice of α that perfectly fits the $\tilde{N}$. So, the problem is to choose $\alpha^\star$ so as to minimize the error.

For this purpose, we first define an energy function $E(\alpha)$ as the sum of square errors which are the difference between the given corrupted image and the represented one by linear combination of eigenvectors, and set a condition to minimize the energy function. The problem, (8) is to find α which minimizes the error function, $E(\alpha)$, which is given as:

$$\alpha^\star = \arg\min_\alpha E(\alpha) \tag{8}$$

with the error function,

$$E(\alpha) = \sum_{j=1}^{k} \left(\tilde{N}(j) - \sum_{i=1}^{m-1} \alpha_i n_i(j) \right)^2 \tag{9}$$

According to (8) and (9), we then find the coefficient values that minimize the energy function using the least-square minimization [3]. Equation (7) is equivalent to the following:

$$\begin{pmatrix} n_1(1) & \cdots & n_{m-1}(1) \\ \vdots & \ddots & \vdots \\ n_1(k) & \cdots & n_{m-1}(k) \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_{m-1} \end{pmatrix} = \begin{pmatrix} \tilde{N}(1) \\ \vdots \\ \tilde{N}(k) \end{pmatrix} \tag{10}$$

We rewrite (10) as:
$$\mathbf{I}\alpha = \tilde{\mathbf{I}} \tag{11}$$

where

$$\mathbf{I} = \begin{pmatrix} n_1(1) & \cdots & n_{m-1}(1) \\ \vdots & \ddots & \vdots \\ n_1(k) & \cdots & n_{m-1}(k) \end{pmatrix},$$

$$\alpha = (\alpha_1, \cdots, \alpha_{m-1})^T, \quad \tilde{\mathbf{I}} = (\tilde{N}(1), \cdots, \tilde{N}(k))^T \tag{12}$$

(a)

(b)
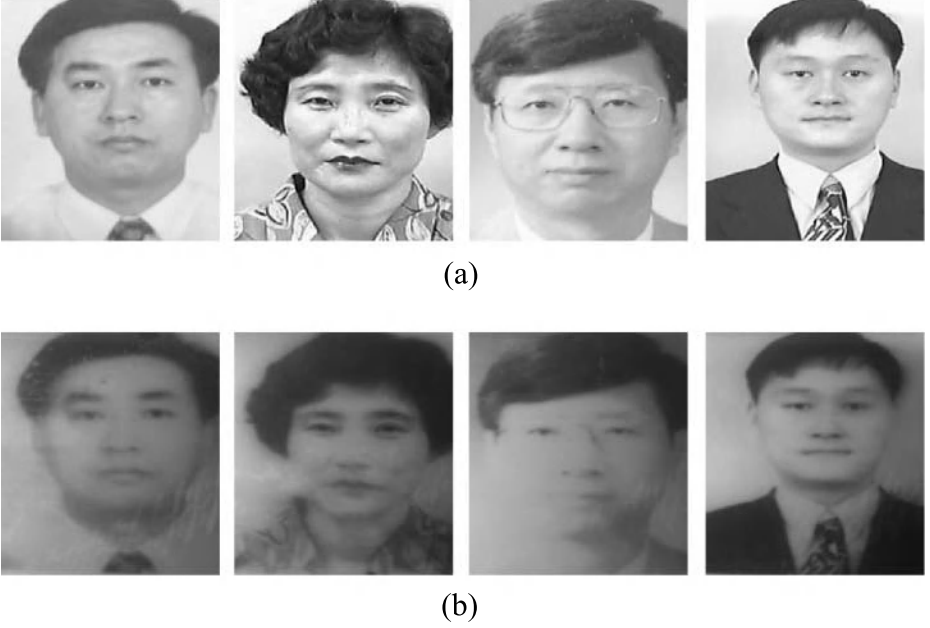
Fig. 2. The example of face images in identification card. (a) Original face images. (b) Face images scanned from identification card

The least-square solution to an inconsistent $\mathbf{I}\alpha^\star = \tilde{\mathbf{I}}$ of k equation in $m - 1$ unknowns satisfies $\mathbf{I}^T\mathbf{I}\alpha^\star = \mathbf{I}^T\tilde{\mathbf{I}}$. If the columns of $\mathbf{I}$ are linearly independent, then $\mathbf{I}^T\tilde{\mathbf{I}}$ has an inverse and

$$\alpha^\star = (\mathbf{I}^T\mathbf{I})^{-1}\mathbf{I}^T\tilde{\mathbf{I}} \tag{13}$$

The projection of $\tilde{\mathbf{I}}$ onto the column space is therefore $\hat{\mathbf{I}} = \mathbf{I}\alpha^\star$. By using (6) and (13), we obtain

$$N(j) = \bar{N}(j) + \sum_{i=1}^{m-1} \alpha_i^\star n_i(j), \quad (j = 1, \cdots, k) \tag{14}$$

For estimating noise parameters, the linear coefficients are applied to the submatrix of eigenvectors corresponding to noise parameters. Thus, the estimation of noise parameters can be defined as (15).

$$P = \bar{N}(s) + \sum_{i=1}^{m-1} \alpha_i n_i(s), \quad (s = k+1, \cdots, k+l) \tag{15}$$
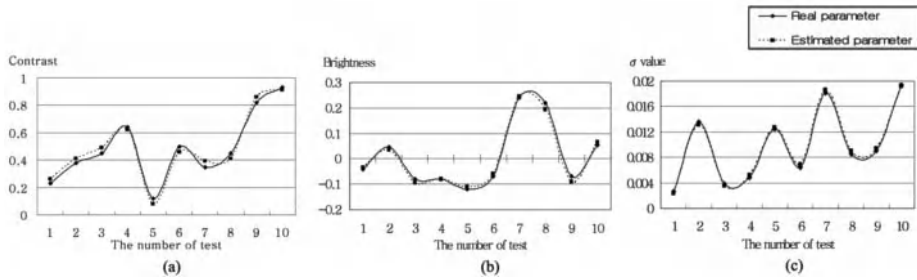
Fig. 3. The comparison of the real parameters and the estimated ones. (a) Contrast parameter. (b) Brightness parameter. (c) Gaussian blur parameter

3.3   Authentication

Given the test image and the original image, face image authentication is to decide if the test image is identical to the original image. In this paper, an synthesized face image is obtained by applying the estimated noise parameter, P to the original face image like (1) and (2). Then, we used the normalized correlation method for authenticating between the synthesized image and the test image [7].

4   Experimental Results and Analysis

For testing the proposed method, we used the Korean Face Database(KFDB) introduced in [6] and face data scanned from identification card (Fig.2). We used 100 persons which have one frontal image in KFDB and generated 100 virtual corrupted images by the method introduced in section 2.1 for testing. We also tested our algorithm against 137 face images scanned from identification card. The resolution of the images was 320 by 200 pixels and the color images were converted to 8-bit gray level images. Also, these 137 face images were scanned with 300dpi.

In the experiment, we performed the following two experiments. First, we compared the difference between the real parameter values and the estimated parameter values against virtual corrupted images from KFDB. We estimated the noise parameters using virtual corrupted face images not including training data. Fig. 3 represents the real parameter values and the parameter values estimated by the proposed method. The experimental result shows that the proposed method can estimate noise parameters accurately. In this case, however, the estimation error of noise parameters for corrupted face images depends on the number and the value of noise parameter. Therefore, it is very important to use suitable parameters to make corrupted images.

Second, we performed an face authentication against face images scanned from identification card. Fig. 4 shows that the EER is improved from 8.05% down to 1.32%. As a result of this experiment, we showed that the proposed method is more robust for face authentication of corrupted face images than before.
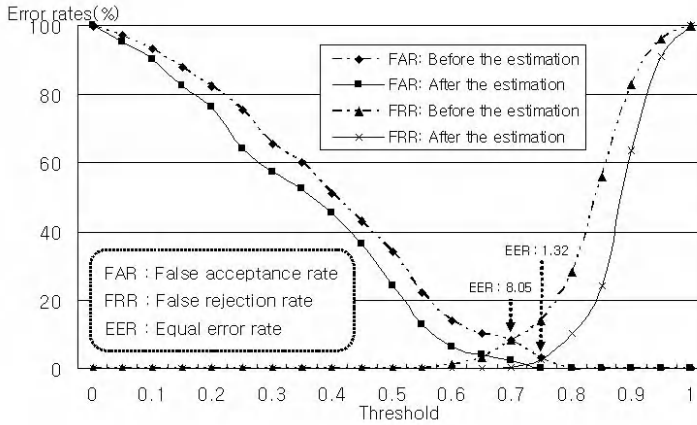
Fig. 4. ROC curve for threshold value used 137 images scanned from identification card

5   Conclusions

In this paper, we have proposed a method for authenticating corrupted face images based on noise model. In contrary to previous studies, our method deals with the corrupted face image based on noise model and uses only one image per person for training. The experimental results show that noise parameter estimation of proposed method is quite accuracy and is very useful for face authentication by solving the noise problem of corrupted face images. Also, the proposed method has a good performance against corrupted face images scanned from identification card.

Our future work is to develop the method for estimating partial noise parameters and generating various corrupted images for accurate face authentication. We expect that the proposed method will be useful for the practical applications for face authentication such as biometric passport system and smart card identification system.

References

[1] A. M. Martinez: Recognizing Imprecisely Localized, Partially Occluded, and Expression Variant Faces from a Single Sample per Class. IEEE Trans. on PAMI **24** (2002) 748–763   188

[2] A. M. Martinez and A. C. Kak: PCA Versus LDA. IEEE Trans. on PAMI **23** (2001) 229–233   188

[3] B.-W. Hwang and S.-W. Lee: Reconstruction of Partially Damaged Face Images Based on a Morphable Face Model. IEEE Trans. on PAMI. **25** (2003) 365–372 190

[4] D. Beymer and T. Poggio: Face Recognition from One Example View. Proc. of Int. Conf. on Computer Vision. (1995) 500–507   188

[5] C. Sanderson and S. Bengio: Robust Features for Frontal Face Authentication in Difficult Image Condition. Proc. of Int. Conf. on AVBPA. (2003) 495–504   187

[6] B.-W. Hwang, M.-C. Roh, H. Byun and S.-W. Lee: Performance Evaluation of Face Recognition Algorithms on the Asian Face Database, KFDB. Proc. of Int. Conf. on AVBPA. (2003) 557–565   192

[7] M. Sadeghi, J. Kittler and K. Messer: A Comparative Study of Automatic Face Verification Algorithms on the BANCA Database. Proc. of Int. Conf. on AVBPA. (2003) 35–43   192

Eigenspace-Based Face Hashing

David C. L. Ngo[1], Andrew B. J. Teoh[1], and Alwyn Goh[2]

[1] Multimedia University, Jalan Ayer Keroh Lama
75450 Melaka, Malaysia
{david.ngo,bjteoh}mmu.edu.my
[2] Corentix Laboratories, 32 Jalan Tempua 5
47100 Puchong, Malaysia
alwyn_goh@yahoo.co.uk

Abstract. We present a novel approach to generating cryptographic keys from biometrics. In our approach, the PCA coefficients of a face image are discretised using a bit-extraction method to *n* bits. We compare performance results obtained with and without the discretisation procedure applied to several PCA-based methods (including PCA, PCA with weighing coefficients, PCA on Wavelet Subband, and LDA) on a combined face image database. Results show that the discretisation step consistently increases the performance.

1 Biometric Key Cryptography

Biometric ergonomics and cryptographic security are highly complementary, hence the motivation for their integrated application. Following the presentation in [1], known methods for generating cryptographic keys from biometric measurements can be characterized as having two stages: feature extraction and key computation. In the feature extraction stage, certain features of raw input from a biometric-measuring device are examined and used to compute a bit string called a feature descriptor. The key computation stage develops a cryptographic key from the feature descriptor and stored cryptographic data in the device. If two descriptors are sufficiently similar, then the same cryptographic key will be generated from them.

Several techniques fitting this two-stage structure have been proposed for generating cryptographic keys from biometrics. Davida et al. [2] describe a second-stage strategy using majority decoding and error-correcting codes and how it could be combined with first-stage approaches for generating feature descriptors from iris scans [3]. Soutar et al. [4] describe a different approach for generating cryptographic keys from fingerprints using optical computing techniques. Monrose et al. [1] describe a mechanism for developing a cryptographic key, called a hardened password, from a text password and the keystroke dynamics of the user while typing the password.

The focus in this paper is on the robust generation of bit-strings from face data; we focus here on the first stage of the key generation process. In principle, our techniques could be combined with any of the second-stage techniques to generate cryptographic keys. We have previously outlined bit-extraction from face data in [5],

and this paper extends the basic methodology to account for several eigenspace-based algorithms.

2   Biometric Hashing Scheme

A procedure, consisting of two sub-stages: image energy compaction and information bit extraction, similar to lossy JPEG compression algorithm [6] is used to generate *n* bit feature descriptors that are tolerant to distortions in the input images.

2.1 Image Energy Compaction

Transform coding techniques exploit the property that for typical images a large amount of energy is concentrated in a small fraction of the transform coefficients. Principal Component Analysis (PCA) [7] is a data dependent transform which will maximize the image energy in the first coefficients. Our approach for generating feature vectors from biometric readings is based on PCA and takes advantage of the characteristics of the data, and the optimal energy compaction properties of the transform. Here we consider four PCA-based methods for image energy compaction: Principal Component Analysis (PCA) [7], Linear Discriminant Analysis (LDA) [8], PCA on Wavelet Subband (WT+PCA) [9], and PCA with discriminatory energy coefficients (PCA(DEC)) [10].
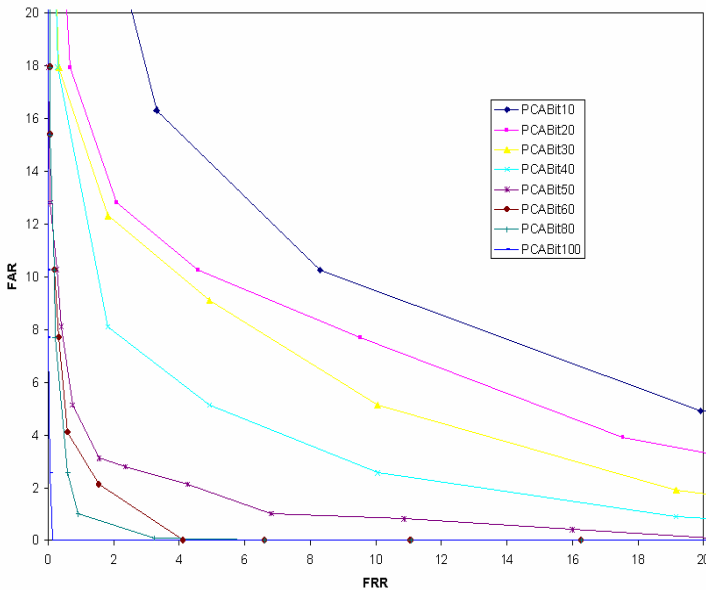
Fig. 1. ROC curves for PCA+BH (20, 40, 60, 80 and 100 bits)

Table 1. Equal error rates for PCA (40 eigenvectors) and PCA+BH (100 bits), PCA(DEC) (40 eigenvectors) and PCA(DEC)+BH (100 bits), WT+PCA (40 eigenvectors) and WT+PCA+BH (100 bits), and LDA (39 eigenvectors) and LDA+BH (39 bits)

Technique	EER	FAR=0%	FAR	FRR
PCA	6.58	38.46	6.95	6.21
PCA+BH	0.07	7.69	0.14	0.00
PCA(DEC)	3.38	17.95	4.20	2.56
PCA(DEC)+BH	0.51	1.01	1.01	0.00
WT+PCA	5.47	19.23	5.82	5.13
WT+PCA+BH	0.12	2.56	0.14	0.11
LDA	2.83	25.64	3.10	2.56
LDA+BH	0.00	0.00	0.00	0.00

2.2 Information Bit Extraction

The transform coefficients can be quantized and coded in any way. Once the PCA projection was completed, we extract n bits by projecting n key-dependent orthogonalized random vectors onto the PCA coefficients. Let α be a vector in $\Box^n$. Let k be a user-specified key.

1. Use k to compute n ($\leq n_\lambda$ the number of eigenvectors required to represent the face space) random vectors in n-space with entries uniformly distributed in the interval [-1, 1].

$$\{\beta_1, \ldots, \beta_n\} \tag{1}$$

2. Apply the Gram-Schmidt process to transform the basis $\{\beta_1, \ldots, \beta_n\}$ into an orthonormal set of vectors $\{\chi_1, \ldots, \chi_n\}$

3. Compute $(\langle \alpha, \chi_1 \rangle, \langle \alpha, \chi_2 \rangle, \ldots, \langle \alpha, \chi_n \rangle)$ using the Euclidean inner product on $\Box^n$.

4. Compute n bits b_i.

$$b_i = \begin{cases} 0 & if \ \langle \alpha, \chi_i \rangle \leq \mu \\ 1 & if \ \langle \alpha, \chi_i \rangle > \mu \end{cases} \tag{2}$$

3   Experimental Results

The bit-extraction method has been tested in terms of their receiver operating characteristic (ROC) and the equal error rate (EER) on a face data set of 10 different images of 40 distinct subjects, drawn from the ORL and Essex Face Databases [11, 12]. The data set was subdivided into a training set, made up of 5 images per class (200 images), and a test set, made up of 5 images per class (200 images). As the adopted face image database contains multiple variations, we first perform geometrical alignment and lighting adjustment before feature extraction.

Due to space limitations, Fig. 1 presents the ROC curves of the enhanced PCA. It should be noted, however, that similar results were obtained in the other approaches

as well. The performance of the system increases with the number of bits extracted. The best EER ($\approx 0\%$) is obtained when 100 bits are extracted from each image.

Table 1 compares the EER's achieved by the proposed methods with the EER's achieved by the raw methods. From the inspection of the table, it is seen that:

1. All the EER are reduced. The improvements for PCA+BH, PCA(DEC)+BH, WT+PCA+BH, and LDA+BH are 98.94%, 84.91%, 97.81%, and 100%, respectively. The later extension provides notably better performance in our tests. The best results are obtained with LDA+BH where the EER is reduced from 2.83% to 0.00%.
2. PCA(DEC)+BH is the worst extended method (EER = 0.51%), and LDA the best raw method (EER = 2.83%). The worst extended method is better than the best raw method.
3. The axis weighting procedure leads to significant performance improvements, which is similar to the finding in [10]. The EER of PCA with discriminatory energy coefficients is 3.38% whereas the EER of PCA is 6.58%. It is seen that the incorporation of the axis weighting procedure improves the EER considerably by 48.63%.
4. WT+PCA gives better results than PCA and this is similar to the finding from a previous study [9].
5. The LDA approach shows better performance than the PCA approach. A similar effect is reported by Swets and Weng [8] where LDA shows improved performance over PCA.

In this experiment, Daubechies-1 [13] is adopted for image decomposition. In fact, in order to select a suitable wavelet, the recognition rates are computed by applying different wavelets on face image decomposition. Daubechies-1 gives the best results in the wavelets studied, and therefore is adopted for image decomposition in our system.

4   Conclusions

The main focus in this paper is on the robust generation of bit-strings from face data; we focus here on the first stage of the key generation process. In principle, the proposed techniques could be combined with any of the second-stage techniques to generate cryptographic keys. The experiment results collected are very encouraging. Dramatic performance improvements are obtained when the bit-extraction procedure is applied to the PCA-based methods.

References

[1]   Monrose, F., Reiter, M.K., Wetzel, S.: Password Hardening Based on Keystroke Dynamics. In Proceedings of the 6th ACM Conference on Computer and Communications Security. Singapore, 73–82. (1999)

[2]   Davida, G.I., Frankel, Y., Matt, B.J.: On Enabling Secure Applications through Offline Biometric Identification. In Proceedings of the 1998 IEEE Symposium on Security and Privacy. Oakland, California, 148–157. (1998)

[3]   Daugman, J.: High Confidence Visual Recognition of Persons by a Test of Statistical Independence. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15, 1148-1161. (1993)

[4]   Soutar, C., Roberge, D., Stoianov, A., Gilroy, R., Kumar, B.V.: Biometric Encryption. In: Nichols, R.K. (ed.): ICSA Guide to Cryptography, McGraw-Hill, New York, 649-675. (1999)

[5]   Goh, A., Ngo, D.C.L.: Computation of Cryptographic Keys from Face Biometrics. In: Lioy, A., Mazzocchi, D. (eds.): Communications and Multimedia Security - Advanced Techniques for Network and Data Protection. Lecture Notes in Computer Science, Vol. 2828. Springer-Verlag, Berlin Heidelberg New York, 1-13. (2003)

[6]   Wallace, G.: The JPEG Still Picture Compression Standard. Communications of the ACM, Vol. 34, No. 4, 30-44. (1991)

[7]   Turk, M., Pentland, A.: Eigenfaces for Recognition. Journal of Cognitive Neuroscience, Vol. 3, No. 1, 71-86. (1991)

[8]   Swets, D.L., Weng, J.J.: Using Discriminant Eigenfeatures for Image Retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 8, 831-836. (1996)

[9]   Feng, G.C., Yuen, P.C., Dai, D.Q.: Human Face Recognition Using PCA on Wavelet Subband. Journal of Electronic Imaging, Vol. 9, No. 2, 226-233. (2000)

[10] Yambor, W., Draper, B., Beveridge, R.: Analyzing PCA-based Face Recognition Algorithms: Eigenvector Selection and Distance Measures. In Proceedings of the 2nd Workshop on Empirical Evaluation in Computer Vision. Dublin, Ireland. (2000)

[11] Database of Faces, AT&T Laboratories, Cambridge University, UK. Available: `http://www.uk.research.att.com/facedatabase.html`

[12] Face Database, Vision Group, University of Essex, UK. Available: `http://hp1.essex.ac.uk/`

[13] Daubechies, I.: The Wavelet Transform, Time-Frequency Localization and Signal Analysis. IEEE Transactions on Information Theory, Vol. 36, No. 5, 961-1005. (1990)

Point to Point Calibration Method of Structured Light for Facial Data Reconstruction

Dongjoe Shin and Jaihie Kim

Department of Electrical and Electronic Engineering,Yonsei University
Biometrics Engineering Research Center, Seoul, Korea
{tunegod,jhkim}@yonsei.ac.kr

Abstract. Since the calibrating point pairs of a structured light are not easy to obtain, most previous work on calibration is related to the uncalibrating method. This paper proposed a new method for determining a set of 3D to 2D point pairs for the offline calibration of structured light system focused on 3D facial data acquisition for the recognition. The set of point pairs is simply determined based on epipolar geometry between a camera and structured light plane, and a *structured light calibrating pattern*. The structured light calibrating process is classified into two stages: the 3D point data acquisition stage and the corresponding 2D data acquisition stage. After point pairs are prepared, the Levenberg-Marquardt (LM) Algorithm is applied. Euclidian reconstruction can be achieved simply using a triangulation, and experimental results from simulation are presented.

1 Introduction

Reconstructing an accurate shape from an image is one of the major subjects in computer vision [1]. It can be considered as inferring the 3D structure of the scene from multiple view geometry. A stereo camera is a well-known approach for extracting 3D data from several 2D images. It is called a passive method since reconstruction only depends on the projective geometry between multiple images and a world structure [2]. On the other hand, an active method intensionally changes the camera parameters or projects special energy to the object for the shape reconstruction. In a structured light system, an object is illuminated from a structured light source and this scene is imaged by a camera located at a distance from a light source. The main advantage of this kind of active vision system is it can alleviate the corresponding problem [1]. Besides, the recent coded light system makes one-shot reconstruction possible. In any case, Euclidian 3D reconstruction is required to know the calibrating information of all cameras and corresponding pairs of a feature to be reconstructed. Therefore, a projector also should be calibrated, because a structured light can be considered as a substituted case for a camera with a pattern projector.

The camera calibration is normally carried out using a calibration rig, whose 3D coordinate data must be known precisely and corresponding 2D data in image plane are easily detectable. This kind of calibration is called *point to*

point calibration, because it is performed from a set of 3D to 2D point pairs [3]. In this case, the main problem is how to determine such point pairs and how to optimize camera parameters from these data. While camera calibration has been well studied, the structured light calibration is not, since it is hard to choose such pairs for the calibration, e.g. the known 2D point in the projector plane can not always illuminate the known 3D position.

Fofi et al. suggested uncalibrated reconstruction which assumes a projector and a camera calibration matrix based on a fundamental matrix and recover scene up to projective reconstruction [4]. More constraints are then added from the projected pattern in order to achieve Euclidian reconstruction. They found additional information from the the grids. Hence, the accuracy of reconstruction results depends on how precisely detect the projected grids. If refined grids are applied for more data points acquisition, then exact detection of the properties from the grids can not be expected. D. Q. Huynh et al. proposed a method to estimate 2D homography which transforms each projected stripe to the image plane in a 2D projective space in order to build a 4-by-3 image to world transformation of each stripe pattern [5]. However, there is much computational burden when many stripes are projected for higher resolution. Therefore, Huynh's idea is not expected for rapid face data acquisition.

In this paper, we propose a hard calibration method of a structured light for Euclidian shape reconstruction. The proposed method is more computationally efficient than Huynh's, because it does not have to estimate all homographies of each projected stripe in order to figure out the image to world transformation. Furthermore, since the proposed system uses two different patterns, i.e. a calibrating grids and the coded pattern, any kind of the coded pattern instead of simple grids is available.

2   Projective Geometry

In this section, simple projective geometry which plays an important role in this paper and outline of notation are described. Homogeneous coordinates are useful representations of a data in the n-tuple projective space denoted by $\mathbb{P}^n$. It is denoted by a bold lower case letter with a "$\sim$", e. g. homogeneous coordinate of a 3D point is $\tilde{\mathbf{x}} = [x\ y\ z\ 1]^t$. On the other hand, a simple vector is written in a bold lower case letter, e.g. $\mathbf{x}$. Homograpy, called collineation or projective transform, is a transform matrix between projective spaces [6]. It is simply denoted in an upper case letter like $H_{3\times3}$, which stands for 3-by-3 projective transformation.

Epipolar Geometry. Epipolar geometry is an intrinsic projective geometry between two views. It is not only independent of scene structure, but also represents correspondence geometry in two-view-geometry [7]. It accounts for where the corresponding point is in the second image plane, given an image point in the first view. An epipole is a point of intersection of the line joining the camera centers with the image plane. A plane containing a base line and real world point determine an epipolar plane. An epipolar line is the intersection of an epipolar

plane with an image plane. Generally, epipolar geometry is used as a constraint for the possible location of corresponding points in another view because the corresponding point should be lying on the epipolar line. This constraint of two view geometry is applied to the projector calibration.

Cross Ratio. The cross ratio is a well-known invariant characteristic in projective geometry. It is a ratio from four collinear point configurations. If the cross ratio of a given set of four points is defined, then its corresponding points on the line crossing the pencil of rays preserve the same cross ratio [8]. The cross ratio can be used for obtaining two geometrical constraints: a test of spatial collinearity and a test of coplanarity [4]. In this paper this invariant characteristic is used when computing world 3D data points for the calibration of a projector. It will be presented in Sect. 3.

3   System Calibration

System calibration basically consists of camera calibration and projector calibration. In this paper, the camera was calibrated by using Levenberg-Marquardt (LM) Algorithm and a Direct Linear Transformation (DLT) solution of a given set of 3D to 2D data was used as an initial guess of LM algorithm [9]. A projector calibrating pattern, a simple colored grid, was projected onto the calibration rig in order to obtain a set of 3D to 2D point data for projector calibration.

3.1   Camera Calibration

The LM Algorithm gives nonlinearly optimized solution of a camera matrix. This algorithm can be seen as a modified version of Gauss-Newton's method, (1).

$$\mathbf{c}_{cam}{}^{(k+1)} = \mathbf{c}_{cam}{}^{(k)} - (J(\mathbf{c}_{cam})^T J(\mathbf{c}_{cam}) + \mu_k I)^{-1} J(\mathbf{c}_{cam})^T \mathbf{r}(\mathbf{c}_{cam}) \qquad (1)$$

$J(\cdot)$ is the Jacobian matrix and $\mathbf{r}(\cdot)$ represents a vector whose elements are nonlinear cost function of camera parameter. Since it has a regularized term,μ, in each iteration denoted by k, the inverse of the Hessian matrix is more stable. Moreover, second derivatives (Hessian matrix) makes optimization more efficient. The cost function of camera matrix is defined as follows, (2). Geometrical distance is used for defining error.

$$r_i(\mathbf{c}_{cam}) = d(\tilde{\mathbf{x}}_i^c, C_{cam}\tilde{\mathbf{x}}_i^w) \qquad (2)$$

$r_i(\mathbf{c}_{cam})$ is i-th element of cost function vector, $\tilde{\mathbf{x}}_i^c \mapsto \tilde{\mathbf{x}}_i^w$ represents i-th point pair and $\tilde{\mathbf{x}}_i^c$ is the point in the camera image plane, and $\tilde{\mathbf{x}}_i^w$ is the point in the world frame, respectably. C_{cam} is the camera matrix to be optimized. The linear solution of a given data set is used for an initial guess, which makes early convergence possible.
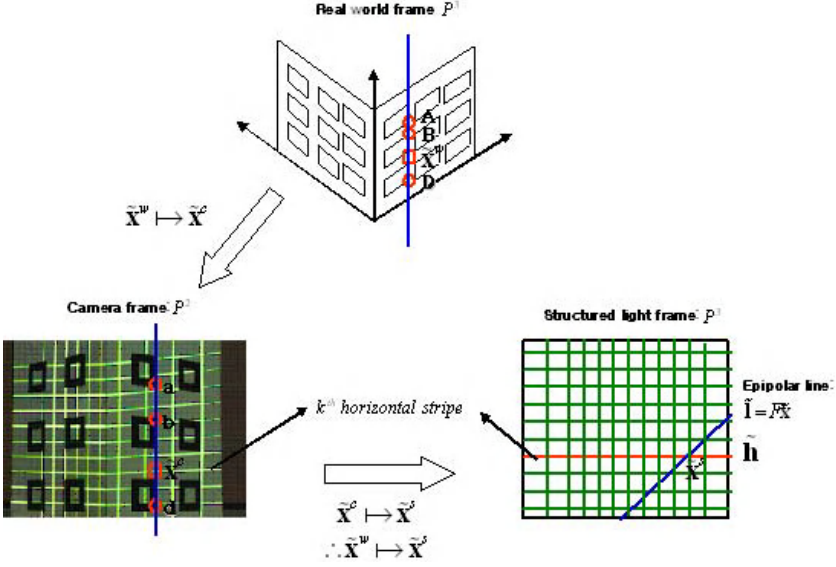
Fig. 1. Structured light calibration: $\tilde{\mathbf{x}}^w \mapsto \tilde{\mathbf{x}}^c$ correspondence is determinded based on the cross ratio, and $\tilde{\mathbf{x}}^c \mapsto \tilde{\mathbf{x}}^s$ is derived from the intersection of epipolar line $\tilde{\mathbf{l}}$ with horizontal line $\tilde{\mathbf{h}}$. Every possible $\tilde{\mathbf{x}}^w \mapsto \tilde{\mathbf{x}}^c$ pair data used when structured light calibration

3.2   Structured Light Calibration

The structured light calibration is considered as a special case of camera calibration, because a structured light seems to be a replacement of a camera in the stereo camera system. The problem is 3D to 2D point pairs for its calibration is not easily obtainable. Although we can select a 2D position in the projector plane, this 2D point is not always expected to fall onto the known 3D position [5]. To overcome this problem, a project Structured Light Calibration (SLC) Pattern is projected onto the camera calibration rig and we exploit the camera image plane which provides the intermediate 2D projective space between 3D world projective space and 2D structured light plane. If $\tilde{\mathbf{x}}^w \in$ World $\mathbb{P}^3$, $\tilde{\mathbf{x}}^c \in$ Camera $\mathbb{P}^2$, and $\tilde{\mathbf{x}}^s \in$ Structued light $\mathbb{P}^2$ then their correspondence can be defined as (3).

$$\tilde{\mathbf{x}}^w \mapsto \tilde{\mathbf{x}}^c \mapsto \tilde{\mathbf{x}}^s \tag{3}$$

The 3D to 2D data pair acquisition scheme for structured light calibration is as follows. First, project SLC Pattern onto the camera calibration rig. Estimate a fundamental matrix (F shown at Fig. 1) between a camera and structured light plane via the crossing points of SLC pattern. Secondly, determine every 4-point-set lying on a collinear line selected by the corner points on the same vertical line of the calibration rig, i.e. $\{(A, a), (B, b), (D, d)\}$. The last point is

Table 1. The estimated camera intrinsic parameters

	α_x	α_y	x_0	y_0	skew	residual error
DLT	3383.5	3325.7	503.46	383.61	-0.28407	1.2555
LM	3377.1	3318.7	503.46	382.6	-0.39617	1.2551

then selected by an intersection point of horizontal line of SLC pattern with the collinear line, i.e. $\tilde{\mathbf{x}}^c$. The corresponding 3D point to the last point is obtained based on the cross ratio of the 4 points, (4). It is called a 3D point acquisition stage.

$$\text{Cross Ratio}\{a, b, \tilde{\mathbf{x}}^c, d\} = \text{Cross Ratio}\{A, B, \tilde{\mathbf{x}}^w, D\} \tag{4}$$

Finally, the point $\tilde{\mathbf{x}}^c$ is transferred to the structured light plane using a fundamental matrix and horizontal Id of SLC pattern, (5). This process is called a corresponding 2D point acquisition stage.

$$\tilde{\mathbf{x}}^s = F\tilde{\mathbf{x}}^c \times \tilde{\mathbf{h}} \tag{5}$$

Where $\tilde{\mathbf{h}}$ represents a homogeneous coordinates of a k-th horizontal line in the structured light projective space, $\mathbb{P}^2$. After preparing all point pairs, LM algorithm gives the nonlinearly optimized parameters of a structured light calibration matrix.

4 Experimental Results

The proposed algorithm has been tested and its reconstruction results are included here. A sharp PG-M25X DLP projector with 1024 by 768 resolutions is used for illuminating a color coded pattern to an object, and JAI co.'s CV-M77, which has 1/3" color CCD is used as a color camera. A simulation code is written in Visual C++ 6.0 and run under a Pentium IV-2.4 [GHz] environment. In order to compare reconstruction results, Head and face 3D scanner, a product of Cyberware co., is used. An object is placed around 1 [m] from a system, and a captured image size is 1024 by 768, with 24-bit color resolution.

Calibration Results. Table 1 and Table 2 show the estimated calibration results of a camera and structured light, respectively. The intrinsic parameter is obtained from the RQ decomposition of the calibration matrix. In this paper, we assume a finite projective camera model [9]. The LM algorithm converges to optimal camera parameters after 5 steps, and a projector matrix is obtained after 12 iterations. Since a cumulative error makes an unstable data pair during projector calibration, more iteration was required to converge. In order to quantify the estimated calibration matrix, a residual error was measured.

Table 2. The estimated structured light intrinsic parameters

	α_x	α_y	x_0	y_0	skew	residual error
DLT	2529.3	2457.8	66.784	139.27	-92.734	3.9593
LM	2474.4	2393.4	101	91.17	-97.823	3.8819

(a)

(b)
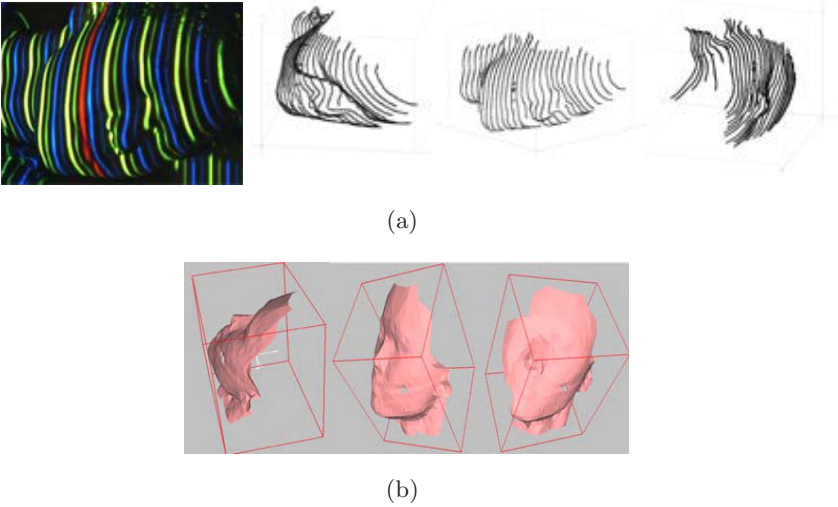
Fig. 2. Face reconstruction results: (a) Input image with the color coded light pattern and Extracted range data (13,968 points) (b) 3D rendering results

Reconstruction Results. Figure 2 shows the result of the performance of face reconstruction. Fig. 2(a) is an input image for face reconstruction and the reconstructed range data. In our experiments color coded pattern is applied for one-shot reconstruction. Around 14,000 data are acquired within 400 [msec]. After building a triangle patches and light effect is added, Fig. 2(b) is generated. Reconstruction points are sparse between lines but a lot of points are obtained within the lines. Therefore, sub-sampling is performed within line data to make an appropriate triangle patch.

5   Conclusion

A point to point calibration method is reduced to the computation of the transformation which models the relation between the 3D object points and their 2D observable corresponding point. This paper shows the possibility of the point to point calibration of a structured light. The 3D input device of the 3D or 2.5D face recognition system requires accurate reconstruction and fast triangulation with higher resolution. Therefore, point to point calibration is inspected to re-

duce reconstruction error, and using different coded patterns from calibrating pattern can make fast triangulation with higher resolution.

Experimental results show around a 2% of depth error for the polyhedral object, but its performance decreases some around the highly curved object. It is induced by the decoding process of the coded pattern and cumulative error when deciding a point pairs of the structured light calibration. Besides, the position of the camera and projector is considered important factor affecting performance.

Nevertheless, the proposed method performs Euclidian face reconstruction with fast triangulation, consistent resolution, and reasonable accuracy. Further works will deal with the head pose estimation between 3D range data from the proposed device and the more accurate 3D face DB.

Acknowledgements

References

[1] E. Mouaddib, J. Batlle, and J. Salvi: Recent progress in structured light in order to solve thecorrespondence problem in stereo vision. Proceedings of IEEE international conference on Robotics and Automation, Vol. 1, pp.130∼136, April 1997 200

[2] M. A. Sid-Ahmed, M. T. Boraie: Dual camera calibration for 3-D machine vision metrology, IEEE Transactions on Instrumentation and Measurement, Vol. 39, No. 3, pp. 512∼516, June 1990 200

[3] R. S. Lu, T. F. Li: Calibration of a 3D vision system using pattern projection, Sensors and Actuators A: Physical, Vol. 104, No. 1, pp. 94∼102, March 2003 201

[4] David Fofi, Joaquim Salvi, El Mustapha Mouaddib: Uncalibrated reconstruction: an adaptation to structured light vision, Pattern Recognition, Vol. 36, No. 7, pp. 1631∼1644, July 2003 201, 202

[5] D. Q. Huynh, R. A. Owens, P. E. Hartmann: Calibration a structured light stripe system: a Novel approach, International Journal of Computer Vision, Vol. 33, No. 1, pp. 73∼86, 1999 201, 203

[6] R. Hartley and A. Zisserman: Multiple view geometry in computer vision, Cambridge University Press, 2000 201

[7] J. Ponce, Y. Genc: Epipolar geometry and linear subspace method: a new approach to weak calibration, International Journal of Computer Vision, Vol. 28, No. 3, pp. 223∼243, 1998 201

[8] C. H. Chen and A. C. Kak: Modeling and calibration of a structured light scanner for 3D robot vision, Proceedings of IEEE International Conference on Robotics and Automation, Vol. 4, pp. 907∼815, March 1987 202

[9] H. Backoglu, M. S. Kamel: A three step camera calibration method, IEEE Transactions on Instrumentation and Measurement, Vol. 46, No. 5, pp. 1165∼1172, October 1997 202

Biometrics Person Authentication Using Projection-Based Face Recognition System in Verification Scenario

Hyeonjoon Moon

Center for Emotion Recognition and Robot Vision (CERRV)
School of Computer Engineering, Sejong University
Seoul, 143-747, Korea
`hmoon@sejong.ac.kr`

Abstract. There are tremendous need for personal verification and identification in internet security, electronic commerce and access control in recent years. Also, as the demands for security in many applications such as data protection and financial transaction become an increasingly relevant issues, the importance of biometrics technology is rapidly increasing. We explored face recognition system for person authentication applications by explicitly state the design decisions by introducing a generic modular PCA face recognition system. We designed implementations of each module, and evaluate the performance variations based on virtual galleries and probe sets. We perform various experiments and report results using equal error rates (EER) for verification scenario. In our experiment, we report performance results on 100 randomly generated image sets (galleries) of the same size.

1 Introduction

As the demands for security in many applications such as data protection and financial transaction become an increasingly relevant issues, the importance of biometric technology is rapidly increasing. In addition to research based on improving the performance of personal authentication and evaluation technologies, standardizations are emerging to provide a common interface and to permit an effective comparison and evaluation of different biometric technologies. We explored core technologies for uni-modal biometrics based on face recognition system and provide performance evaluation technology for higher reliability of core biometric algorithms.

In this paper, we present a modular projection-based face recognition system which includes normalization, PCA projection [5], and recognition modules [3]. Each module consists of a series of basic steps, where the purpose of each step is fixed. The selection of which algorithm is in each step is a design decision. Based on the modular design for projection-based algorithms, we evaluate different implementations. We report verification performance score for each category of probes using equal error rate (EER).
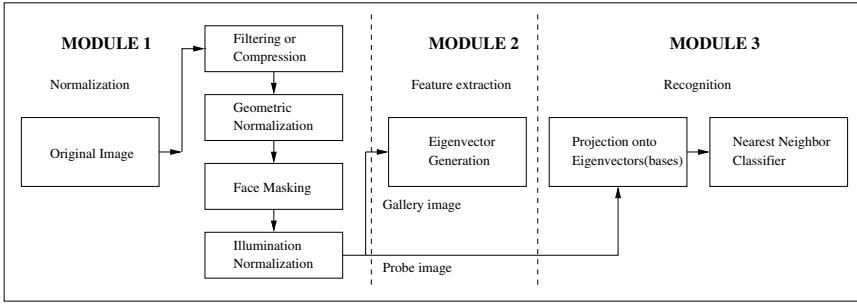
Fig. 1. Block Diagram of Projection-based Face Recognition System

In biometrics person identification and verification applications, two critical questions are often ignored [2]. First, how does performance vary with different galleries and probe sets. Second, when is a difference in performance between two algorithms statistically significant. We systematically designed the modular based face recognition system and explored this question by analyzing randomly generated 100 galleries of the same size. We then calculate performance on each of the galleries against **fb** and duplicate probes. Because we have 100 scores for each probe category, we can examine the range of scores, and the overlap in scores among different implementations of the PCA-based face recognition system.

2   Projection-Based Face Recognition System

Our face recognition system consists of three modules and each module is composed of a sequence of steps (see Figure 1). The goal of normalization is to transform facial images into a standard format that removes variations that can affect recognition performance. The second module performs the PCA decomposition on the training set. This produces the eigenvectors (eigenfaces) and eigenvalues. The third module identifies the face from a normalized image, and consists of two steps. The first step projects the image onto the eigen representation. The critical parameter in this step is the subset of eigenvectors that represent the face. The second step recognizes faces using a nearest neighbor classifier. The critical design decision in this step is the similarity measure in the classifier.

3   Test Design

3.1   Test Sets, Galleries, and Probe Sets

We use FERET database for our experiment which provides facial images for both development and testing of face recognition algorithms. For details of the FERET database, refer to Phillips et al. [4]. To obtain a robust comparison of algorithms, it is necessary to calculate performance on a large number of

Table 1. Size of galleries and probe sets for different probe categories

Probe category	duplicate I	duplicate II	**FB**	**fc**
Gallery size	1196	864	1196	1196
Probe set size	722	234	1195	194

galleries and probe sets. This allowed a detailed analysis of performance on multiple galleries and probe sets.

To allow for a robust and detailed analysis, we report verification scores for four categories of probes. The first probe category was the **FB** probes. For each set of images, there were two frontal images. One of the images was randomly placed in the gallery, and the other image was placed in the **FB** probe set. The second probe category contained all duplicate frontal images in the FERET database for the gallery images. We refer to this category as the duplicate I probes. The third category was the **fc** (images taken the same day, but with a different camera and lighting). The fourth consisted of duplicates where there is at least one year between the acquisition of the probe image and corresponding gallery image. We refer to this category as the duplicate II probes. The size of the galleries and probe sets for the four probe categories are presented in Table 1. The **FB**, **fc**, and duplicate I galleries are the same. The duplicate II gallery is a subset of the other galleries. None of the faces in the gallery images wore glasses.

3.2   Verification Model

In our verification model, a person in image p claims to be the person in image g. The system either accepts or rejects the claim. (If p and g are images of the same person then we write $p \sim g$, otherwise, $p \not\sim g$.) Performance of the system is characterized by two performance statistics. The first is the probability of accepting a correct identity; formally, the probability of the algorithm reporting $p \sim g$ when $p \sim g$ is correct. This is referred to as the verification probability, denoted by P_V (also referred to as the hit rate in the signal detection literature). The second is the probability of incorrectly verifying a claim formally, the probability of the algorithm reporting $p \sim g$ when $p \not\sim g$. This is called the false-alarm rate and is denoted by P_F.

Verifying the identity of a single person is equivalent to a detection problem where the gallery $G = \{g\}$. The detection problem consists of finding the probes in $p \in P$ such that $p \sim g$.

For a given gallery image g_i and probe p_k, the decision of whether an identity was confirmed or denied was generated from $s_i(k)$. The decisions were made by a *Neyman-Pearson* observer. A Neyman-Pearson observer confirms a claim if $s_i(k) \leq c$ and rejects it if $s_i(k) > c$. By the Neyman-Pearson theorem, this decision rule maximized the verification rate for a given false alarm rate α. Changing c generated a new P_V and P_F. By varying c from it's minimum to maximum value, we obtained all combinations of P_V and P_F. A plot of all

combinations of P_V and P_F is a receiver operating characteristic (ROC) (also known as the relative operating characteristic) [1]. The input to the scoring algorithm was $s_i(k)$; thresholding similarity scores, and computing P_V, P_F, and the ROCs was performed by the scoring algorithm.

The above method computed a ROC for an individual. However, we need performance over a population of people. To calculate a ROC over a population, we performed a round robin evaluation procedure for a gallery G. The gallery contained one image per person.

The first step generated a set of partitions of the probe set. For a given $g_i \in G$, the probe set P is divided into two disjoint sets D_i and F_i. The set D_i consisted of all probes p such that $p \sim g_i$ and F_i consisted of all probes such that $p \nsim g_i$.

The second step computed the verification and false alarm rates for each gallery image g_i for a given cut-off value c, denoted by $P_V^{c,i}$ and $P_F^{c,i}$, respectively. The verification rate was computed by

$$P_V^{c,i} = \begin{cases} 0 & \text{if } |D_i| = 0 \\ \frac{|s_i(k) \leq c \text{ given } p_k \in D_i|}{|D_i|} & \text{otherwise,} \end{cases}$$

where $|s_i(k) \leq c \text{ given } p \in D_i|$ was the number of probes in D_i such that $s_i(k) \leq c$. The false alarm rate is computed by

$$P_F^{c,i} = \begin{cases} 0 & \text{if } |F_i| = 0 \\ \frac{|s_i(k) \leq c \text{ given } p_k \in F_i|}{|F_i|} & \text{otherwise.} \end{cases}$$

The third step computed the overall verification and false alarm rates, which was a weighted average of $P_V^{c,i}$ and $P_F^{c,i}$. The overall verification and false-alarm rates are denoted by P_V^c and P_F^c, and was computed by

$$P_V^c = \frac{1}{|G|} \sum_{i=1}^{|G|} \frac{|D_i|}{\frac{1}{|G|} \sum_i |D_i|} P_V^{c,i} = \frac{1}{\sum_i |D_i|} \sum_{i=1}^{|G|} |s_i(k) \leq c \text{ given } p_k \in D_i| \cdot P_V^{c,i}$$

and

$$P_F^c = \frac{1}{|G|} \sum_{i=1}^{|G|} \frac{|F_i|}{\frac{1}{|G|} \sum_i |F_i|} P_F^{c,i} = \frac{1}{\sum_i |F_i|} \sum_{i=1}^{|G|} |s_i(k) \leq c \text{ given } p_k \in F_i| \cdot P_F^{c,i}.$$

The verification ROC was computed by varying c from $-\infty$ to $+\infty$.

In reporting verification scores, we state the size of the gallery G which was the number of images in the gallery set G and the number of images in the probe set P. All galleries contained one image per person, and probe sets could contain more than one image per person. Probe sets did not necessarily contain an image of everyone in the associated gallery. For each probe p, there existed a gallery image g such that $p \sim g$.

For a given algorithm, the choice of a suitable hit and false alarm rate pair depends on a particular application. However, for performance evaluation and comparison among algorithms, the *equal error rate* (EER) is often quoted. The equal error rate occurs at the threshold c where the incorrect rejection and false alarm rates are equal; that is $1 - P_V^c = P_F^c$.
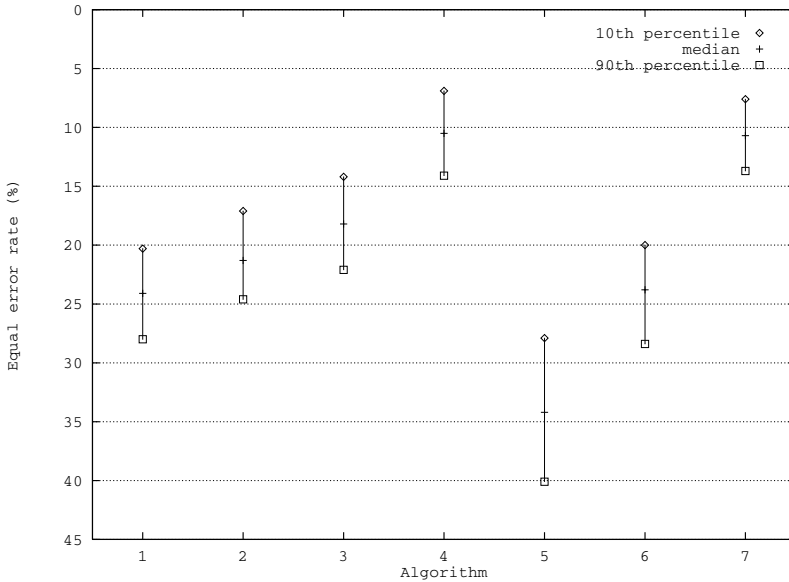
Fig. 2. The range of EER (%) for duplicate I probes using seven different nearest neighbor classifiers. Note that the value of y-axis is in reverse order. The nearest neighbor classifiers presented are: (1) L_1, (2) L_2, (3) Angle, (4) Mahalanobis, (5) L_1+Mahalanobis, (6) L_2+Mahalanobis, and (7) Angle+Mahalanobis

4   Experiment

4.1   Variations in Galleries and Probe Set

The performance comparison among the different probe sets cannot be directly compared since the number of probes in each category of probes are different. The natural question is, when is the difference in performance between two classifiers significant?

To address this question, we randomly generated 100 galleries of 200 individuals, with one frontal image per person. The galleries were generated without replacement from the **FB** gallery of 1196 individuals in experiment. Then we scored each of the galleries against the **FB** and duplicate I probes for each of the seven classifiers in experiment. (There were not enough **fc** and duplicate II probes to compute performances for these categories.) For each randomly generated gallery, the corresponding **FB** probe set consisted of the second frontal image for all images in that gallery; the duplicate I probe set consisted of all duplicate images in the database for each image in the gallery.

The faces are represented by their projection onto the first 200 eigenvectors and the classifier uses the L_1 norm. For an initial look at the range in performance, we examine this baseline algorithm. There are similar variations for the six remaining distances. For each classifier and probe category, we had 100 different scores. Our experiment shows the histogram of EER (%) using the baseline

Table 2. Comparison of verification performance scores for Baseline, Proposed I ($\mu = 0.0$ and $\sigma = 1.0$, LPF, first low order eigenvector removed, angle+Mahalanobis distance), and Proposed II ($\mu = 0.0$ and $\sigma = 1.0$, Wavelet [0.5bpp], first low order eigenvector removed, L1+Mahalanobis distance) algorithm. Performance scores are equal error rate (EER)

	Probe category			
Algorithm	duplicate I	duplicate II	**FB** probe	**fc** probe
Baseline	0.24	0.30	0.07	0.13
Proposed I	0.11	0.21	0.07	0.15
Proposed II	0.20	0.22	0.07	0.10

algorithm for duplicate I probe. In this case, the performance ranges from 18.8 to 33.2 for EER which clearly shows a large range in performance of the 100 galleries.

In Figure 2, we reported a truncated range of EER (%) for the seven different nearest neighbor classifiers for duplicate I probe. For each classifier, score is marked with; the median by $\times$, the 10th percentile by $+$, and 90th percentile by $*$. We plotted these values because they are robust statistics. We selected the 10th and 90th percentile because they mark a robust range of scores and outliers are ignored. From these results, we get a robust estimate of the overall performance of each classifier.

4.2   Discussion

The main goal of our experiment was to get a rough estimate of when the difference in performance is significant. From Figure 2, the range in verification score is approximately ± 0.06 about the median. This suggests a reasonable threshold for measuring significant difference in performance for the classifiers is ~ 0.12.

The top performing nearest neighbor classifiers were the Mahalanobis and angle+Mahalanobis. These two classifiers produces better performance than the other methods as shown in Figure 2. In this experiment, the L_1+Mahalanobis received the lowest verification performance scores. This suggest that for duplicate I scores that the angle+Mahalanobis or Mahalanobis distance should be used. Based on the results of this experiment, performance of smaller galleries can predict relative performance on larger galleries.

5   Conclusion

We proposed a biometrics person authentication model based on face recognition system and evaluation procedure. We introduced a modular design for PCA-based face recognition systems and systematically measure the impact of these variations on performance. Also, We generated 100 galleries and calculate performance against **fb** and duplicate probes. Then, we examined the range and

the overlap in scores among different implementations which shows the performance within a category of probes can vary greatly.

From the results throughout the series of experiments, we present two models for PCA-based face recognition system. In proposed models, our design decision includes processing steps with better performance in each module. The choice of steps used in Proposed I system includes: (1) illumination normalization ($\mu = 0.0$ and $\sigma = 1.0$), (2) Low-pass filtering (LPF), (3) remove first low order eigenvector, and (4) angle+Mahalanobis distance. The choice of steps used in Proposed II system includes: (1) illumination normalization ($\mu = 0.0$ and $\sigma = 1.0$), (2) wavelet compression [0.5 bpp], (3) remove first low order eigenvector, and (4) L_1+Mahalanobis distance. Proposed I system addresses the effects of LPF with angle+Mahalanobis distance while Proposed II system represents wavelet compression with L_1+Mahalanobis distance.

In Table 2, the verification performance for duplicate I probe is improved from 0.24 to 0.11 for Proposed I method, and duplicate II probe improved from 0.30 to 0.21 for Proposed I method. The verification performance score for **FB** probe shows same results for all three methods, and **fc** probe improved from 0.13 to 0.10 for Proposed II method.

Based on these results, the proposed algorithms show reasonably better performance for duplicate I, duplicate II (for Proposed I method) and **fc** probes (for Proposed II method) than the baseline algorithm in verification scenario. For **FB** probes, verification results show almost identical performance scores for each method used.

Acknowledgement

References

[1] J. P. Egan. *Signal Detection Theory and ROC Analysis.* Academic Press, 1975. 210

[2] H. Moon and J. Kim. Biometrics identification and verification using projection-based face recognition system. volume LNCS 2908. Springer, 2003. 208

[3] H. Moon and P. J. Phillips. Computational and performance aspectsm of projection-based face recognition algorithms. *Perception*, 30:303–321, 2001. 207

[4] P. J. Phillips, H. Moon, S. Rizvi, and P. Rauss. The feret evaluation methodology for face-recognition algorithms. *IEEE Trans. PAMI*, 22:1090–1104, 2000. 208

[5] M. Turk and A. Pentland. Eigenfaces for recognition. *J. Cognitive Neuroscience*, 3(1):71–86, 1991. 207

Extraction of Glasses in Human Face Images

Yi Xiao[1] and Hong Yan[1,2]

[1]School of Electrical & Information Engineering
University of Sydney, NSW 2006, Australia
`yix@ee.usyd.edu.au`
[2]Department of Computer Engineering & Information Technology
City University of Hong Kong, Kowlon, Hong Kong
`h.yan@cityu.edu.hk`

Abstract. In this paper, a face model is established based on the Delaunay triangulation for extraction of eyeglasses in human face images. A face shape is decomposed into a few chains represented by the non-shortest edge shared triangles. The width and length of a chain are quantitatively measured by the triangles in the chain. An eye region consists of small sized triangles and triangle chains with short length and narrow width. And an eye region and its glasses frame are linked the triangle chains called surrounding area. Eyes and the glasses are located by measuring the symmetry and similarity of any two regions and their surrounding areas. Experiments show that the glass boundary in various shapes can be traced automatically and accurately and small broken edges can be linked without obviously distortion.

1 Introduction

In human face image analysis systems, glasses significantly affect the performance of face detection, face recognition, and facial expression understanding. Location of glasses in a face image is a difficult task because of variations in glass frame shape, size and color and the different lighting conditions. Jing and Mariani [1] have described a deformable contour based method for glass detection, which combines edge features and geometrical features. The contours representing glass frames may be deviated by nose, eyebrow, eyes and wrinkles. To overcome this drawback, Jing, Mariani and Wu [2] have presented a method using Bayes rules that incorporate the features around each pixel. Bayes rule based method may be stuck at local optima of the objective function. Wu, C. et al [3] have further applied a Morkov chain Monte Carlo method to locate the glasses by searching for the global optimum of the objective function. For the above algorithms, presumed number of points is used to define the glasses contour and the points can only converge to local or global optimum, exact frame shapes may not be presented. And the algorithms are sensitive to rotation, tilt and frame thickness. Wu, H. et al [4] have described a method to detect glasses frames using the 3D features. A 3D Hough transform is carried on to obtain a plane in which 3D features are concentrated. The group of 3D features

belonging to the frame of the glasses is detected based on some geometry constraints. This approach does not require any prior knowledge about face pose, eye positions, or the shape of the glasses but the images are from a trinocular stereo vision system.

This paper proposes a Delaunay triangulation [5] based method to locate glasses from a front-up face image. The aim is to develop an automated method that can trace the exact shape of glasses without prior knowledge about face pose, eye positions, or the shape of the glasses. In our method, a Delaunay triangulation model is established on a binarized face image. As the Delaunay triangulation inherently incorporate neighborhood and adjacency information, it is efficient for point clustering [6]. The dense points in a facial feature region can be clustered by measuring different sized Delaunay triangles. The triangulation model also provides information about the spatial relation between clusters. Glass frames are the immediate neighbors of eye regions. They can be identified by specific Delaunay triangles surrounding the eye regions.

2 The Proposed Method

2.1 Face Image Modelling

We first enhance the edges of the face image using the image intensity gradient so that the boundaries of the facial features are extracted reliably after binarization. Here we call an 8-neighbor connected foreground pixel area as an object. Objects with their pixel number less than n_{th} are removed as noise. As a result, the layout of eyes and glass frames emerge and points in an eye region are surrounded and well separated from glass frames (we only discuss the case that the eyes covered by glasses). Due to the variations of illumination and color, areas representing eyes or glass frames may be broken. Figure 1 shows an example of a face image and its binarization result.

To create a face image model, the Delaunay triangulation is established on the foreground pixels of the objects. Delaunay triangles are classified into two types. A *foreground triangle* is a Delaunay triangle in an object. A *background triangle* is a Delaunay triangle outside any objects. As the Delaunay edges within an object have lengths short than $\sqrt{2}D$, where D is the sample grid for the digital image. The edge lengths outside any objects are longer than $\sqrt{2}D$. Clear thresholds can be found to differentiate the two types of triangles.

$$F = \{T(v_i, v_j, v_k) \,|\, \max\{L_{ij}, L_{ik}, L_{jk}\} < th\}$$
$$B = \{T(v_i, v_j, v_k) \,|\, \max\{L_{ij}, L_{ik}, L_{jk}\} \geq th\}$$

Where F is the set of foreground triangles. B is a set of background triangles. $T(v_i, v_j, v_k)$ is a Delaunay triangle with vertices v_i, v_j and v_k. L_{ij} is the length of the Dealunay edge with vertices v_i and v_j. $th = \sqrt{2}D$. Figure 2 shows the Delaunay triangulation model of the face image in Figure 1 (a).

Fig 1. A face image and its result after edge enhancement, binarization and noise removal
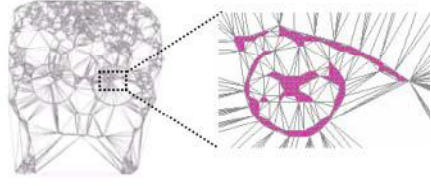
Fig 2. A Delaunay triangulation face model. Foreground (filled) triangles occupy in objects, background triangles (unfilled) are outside any objects

2.2 Segmentation of Eye Regions

To segment eye regions, we decompose a face model into a number of triangle chains. *A triangle chain* is a set of non-shortest edge shared background triangles, of the form T, L, … L, T, where T and L stand for a terminal triangle and a linking triangle respectively. *A terminal triangle* is a background triangle sharing one non-shortest edge with its adjacent triangle in a chain. *A linking triangle* is a background triangle sharing its two non-shortest edges with its adjacent triangles in a chain. Figure 3. illustrates the triangle chains in a face image model for Figure 1.

Fig 3. The triangle chains in the face model of Figure 1. Each triangle chain is assigned with the same color
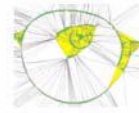
Fig 4. A dense point area (The eye region)

A chain is expressed mathematically as: $B = \{t^0, t^1, .. t^{n-1}\}$. Where $t^j = \{v_1^j, v_2^j, v_3^j\}$ is the $(j+1)$th triangle in a chain of non-shortest edge shared triangles with vertices v_1^j, v_2^j, v_3^j, $j = 0, 1, 2, .., n-1$. $v_k^j = \{x_k^j, y_k^j\}$, $k = 1, 2, 3$ with x_k^j, y_k^j being rectangle coordinate of v_k^j. t^0 and t^{n-1} are terminal triangles and n is the number of triangles in B. B is quantitively measured by its width $w(B)$ and length $l(B)$. They are defined as follows:

$$w(B) = \max_{\substack{0 < j < \\ n-1}} \{ \max_{k=1,2} \{ |v_1^j - v_k^j| \mid v_1^j, v_k^j \in t^j \} \}$$

$$l(B) = n$$

Where, $w(B)$ is the maximum edge-length of triangles in B.

In a face model, if $w(B) < w_{th}$ && $l(B) < l_{th}$, the background triangles are marked as "quasi-foreground triangles". The edge-shared foreground triangles and quasi-foreground triangles form a dense point area (See Figure 4). A Delaunay edge on the boundaries of a dense point region (represented by polygon C_e) is shared by a foreground triangle/quasi-foreground triangle and a background triangle. The surrounding of C_e is defined as a group of such background triangles that each of them has one edge on C_e. We call these triangles as *surrounding triangles*. The region of a surrounding is approximated by polygon C_f. The vertices of C_f are the vertices of surrounding triangles not on C_e. Figure 5 shows the surroundings in a face model. A $C_e(i)/C_f(i)$ is a possible eye/eye frame region.

Fig 5. (a) Some surroundings (filled triangles) in a face model; (b) A C_e and its C_f

2.3 Eye/Glasses Frame Regions Identifications

Assume an eye is in the glasses frame, we measure the size and the relative position of C_e and C_f to filter out the impossible eye regions.

A possible eye/glasses frame region satisfies following conditions:

1) $W_f < 8W_e$ && $H_f < 9H_e$;
2) $x_{f,R} > x_{e,R}$ && $x_{f,L} < x_{e,L}$;
3) $y_{f,B} < y_{e,B}$ && $y_{f,T} > y_{e,T}$.

W_f and W_e are the widths of the rectangles bounding C_f and C_e respectively; H_f and H_e are the heights of the rectangles bounding C_f and C_e respectively; $y_{e,B}$ and $y_{e,T}$ are the y coordinates of the bottom and top points of the rectangle bounding C_e respectively; $x_{e,L}$ and $x_{e,R}$ are the x coordinate of the left and right most points of the rectangle bounding C_e respectively; $y_{f,B}$ and $y_{f,T}$ are the y coordinates of the bottom and top points of the rectangle bounding C_f respectively; $x_{f,L}$ and $x_{f,R}$ are the x coordinates of the left and right most points of the rectangle bounding C_f respectively.

2.4 Eye/Frame Region Location

The eye regions and glass frame regions can be identified by measuring the similarity and symmetry of two possible eye/frame regions $C_e(i)/C_f(i)$ and $C_e(j)/C_f(j)$. To measure the similarity of $C_e(i)/C_f(i)$ and $C_e(j)/C_f(j)$, *width ratio* (r_W) and *height ratio* (r_H) are employed, and the symmetry is measured by the difference Δx of the center of $C_e(i)$ and $C_e(j)$ and the center of $C_f(i)$ and $C_f(j)$.

$$r_{eW} = \frac{W_e(i)-W_e(j)}{W_e(i)+W_e(j)}, r_{eH} = \frac{H_e(i)-H_e(j)}{H_e(i)+H_e(j)},$$

$$r_{fW} = \frac{W_f(i)-W_f(j)}{W_f(i)+W_f(j)}, r_{fH} = \frac{H_f(i)-H_f(j)}{H_f(i)+H_f(j)}$$

$$\Delta x = |x_{e,L}(i)+x_{e,R}(i)+x_{e,L}(j)+x_{e,R}(j)-x_{f,L}(i)-x_{f,R}(i)-x_{f,L}(j)-x_{f,R}(j)|$$

Let $S = r_{eW}r_{eH}r_{fW}r_{fH}\Delta x$. The pair of $C_e(i)/C_f(i)$ and $C_e(j)/C_f(j)$ with minimum S is regarded as the eye/frame regions.

3   Experiment

We have conducted experiments using the database of the XM2VTSDB [7]. The database was acquired within the M2VTS project. The XM2VTSDB contains 295 subjects with different racial types, ages and skin colors. From the 295 images (average size: 162×242), 103 frontal images with glasses were selected.

The procedure for eye/glasses frame location can be summarized as: Edge enhancement, binarization, noise removal, Delaunay triangulation construction, eye region clustering and eye/glasses frame location.

The choice of the parameters depends on the resolution and the size of the images. Experiments show that the performance of the glasses extraction depends mainly on the eye region clustering. Improper setting of the values of w_{th} and l_{th} may cause the splitting or merging of eye regions, which leads the incorrect location.
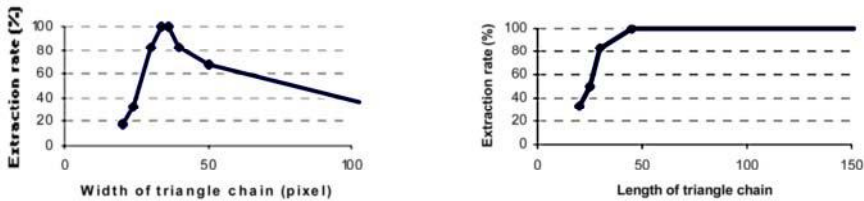
Fig 6. Extraction rates versus w_{th} or l_{th}

Figure 6. summarizes the effects of w_{th} and l_{th} on the extraction accuracy. We can see from the diagram that the eye region segmentation is sensitive to the threshold of w_{th}. When the threshold value of w_{th} or l_{th} is small, triangle chains in eye regions may not be pruned, causing the splitting of the eye regions. When the threshold value of w_{th} is large, triangle chains around eye regions may be included in the eye regions, causing the merging of the eye regions.

Figure 7. shows extracted face boundaries for several tested images. The shapes of glasses vary individually. Some of them are thin and round, and some of them are broad or flat. In Figures 7 (a, b, d, e), the edge of frame is broken. The missing edges are not too long, and in this case the gap can be filled by directly connecting the nearby boundary segments. In Figure 7 (f) the gap is too large, and a direct link will obviously deviate the frame boundary, causing distortion at the broken position. Due to the effects of illumination, not all edges of the frames in Figures 7 (c, e) are extracted during the binarization. Consequently, slight distortion exists in the results. In Figure 8 (a), eyes are outside glasses frames. And in Figure 8 (b), the color of upper part of the glasses frames is too light. For both cases in Figure 8, the proposed method is unsuccessful for the extraction.

(a)   (b)   (c)

(d)   (e)   (f)
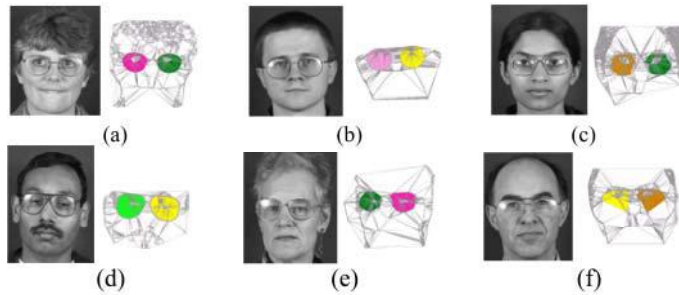
Fig 7. Succeeded extractions ($l_{th} = 45$ and $w_{th} = 36(pixel)$)

Fig 8. Problem cases: the glasses do not cover eyes or the glass frames have very low contrast

The glasses in 88 (out of 103) images were successfully traced. The failure was due to the blur of images, the missing of frame edges and the divergence of eyes from glasses.

The processing is pixel-oriented; the total processing time depends mainly on eye region segmentation and identification. The size and number of edges in the image is the main factor that affects computation time. The average processing time is 4s for an image in XM2VTSDB. The experiments were computed on a Pentium-III processor with a clock frequency of 800 MHz.

A comparison of the application and limitation between the proposed method and that reported in [2] and [3] is tabled below.

Table 1. The comparison of the methods

Method	Application	Limitation
Jing et al [2]	Front-up	Eyes not in glasses frames, automatic
Wu, C. et al [3]	Front-up	Eyes not in glasses frames, tilt
Proposed method	Front-up	Eyes not in glasses frames

4   Conclusion

In a Delaunay triangulation face model, a face shape can be decomposed into a number of triangle chains that can be quantitatively described. The eye regions can be segmented by clustering the small sized triangles and triangle chains with short length and narrow width into one group. Eyes and glasses frame are identified by measuring the symmetry, the size and relative position of any two of these groups and their surrounding areas.

Comparing to the existing methods, the proposed approach needs no prior knowledge about face pose and eye positions, and it works well in dealing with arbitrary frame shapes. The method can also link broken edges and tolerate rotation, tilt and frame thickness.

Acknowledgement

References

[1]   Jing, Z., Mariani, R., Wu, J., Glasses detection and extraction by deformable contour, in Proceedings of the 15th International Conference on Pattern Recognition, (2000), Vol. II, p.4B.

[2]   Jing, Z., Mariani, R., Wu, J., Glasses Detection and Extraction by Bayes Rules and Deformable Contour, The 4th world multiconference on Systemics, Cybernetics and Informatics (SCI), Orlando, Florida, (2000).

[3]   Wu, C., Liu, C., Shum, H. Y., Xu, Y. Q. Zhang, Z., Automatic Eyeglasses Removal from Face Images, ACCV2002: The 5th Asian Conference on Computer Vision, (2002), Australia.

[4]   Wu, H., Yoshikawa, G., Shioyama, T., Lao, T., Kawade, T., Glasses frame detection with 3D Hough transform, Proceedings. 16th International Conference on Pattern Recognition, (2002) Vol 2, pp 346-349.

[5]   A. Okabe, B. Boots and K. Sugihara, Spatial Tessellations-Concepts and Applications of Voronoi Diagrams, Second Edition, Wiley Series in probability and statistics, 2000.

[6]   Xiao, Y. and H. Yan, Text region extraction in a document image based on the Delaunay tessellation, Pattern Recognition, Vol. 36, No. 3, 2003, pp. 799-809.

[7]   http://www.ee.surry.ac.uk/Research/VSSP/xm2vtsdb.

Ant Colony Optimization for Feature Selection in Face Recognition

Zhong Yan and Chunwei Yuan

Department of Biomedical Engineering, Southeast University
Nanjing, 210096, China
`{seu_yanz,cwy}@seu.edu.cn`

Abstract. To render the face recognition work more efficiently, ACOSVM, a face recognition system combining Ant Colony Optimization (ACO) with Support Vector Machine (SVM), is presented, which employs SVM classifier with the optimal features selected by ACO. The Principal Component Analysis method (PCA) is used to extract eigenfaces from images at the preprocessing stage, and then ACO for selection of the optimal subset features using cross-validation is described, which can be considered as wrapper approach in the feature selection algorithms. The experiments indicate that the proposed face recognition system with selected features is more practical and efficient when compared with others. And the results also suggest that it may find wide applications in the pattern recognition.

1   Introduction

With the development of new information technology and media, more effective and friendly methods for human computer interaction are being developed, among which computer vision systems for people monitoring will play an increasingly important role in our lives. Examples include face detection, face tracking, face recognition, action recognition and gender classification. These emerging fields greatly enrich the implication of computer application and improve the capability of computer system. Over the past decade years numerous approaches for face recognition have been proposed in the computer vision community, successful face recognition can boost the performance of various systems such as identity authentication, access control, and surveillance; surveys can be found in [1,2]. Among all these approaches, Eigenface based on Principal Component Analysis (PCA), also known as Karhunen-Loeve transform, has demonstrated success.

In the nineties an algorithm called Ant System (AS)[3] was proposed as a novel heuristic approach for discrete optimization problems. This algorithm is derived from the forgiving behavior of real ants in nature. People find that real ants have abilities to find the shortest path from the food source to the nest without any visual cues by collective behavior through pheromone (a chemical substance). Since AS was advanced it has inspired researchers lots of interest. They design some improved algorithms and apply to discrete applications [4,5,6]. Later, all versions of AS were consolidated into Ant Colony Optimization (ACO) [7] which provided a unitary view of the ongoing research.

In this paper, we propose an appearance-based face recognition method based on Support Vector Machine (SVM) and the intelligence of ant colonies, in which ACO is utilized to select feature subset encoding mostly facial information and SVM is used as classifier.

The rest of this paper is organized as follows. In section 2, SVM is briefly described, followed by the details of using ACO to select optimal features in section 3; the face recognition based on SVM and ACO is outlined in section 4; section 5 reports the experiment results and section 6 concludes the paper and points to future work.

2   Support Vector Machine

Support Vector Machine [8,9] is a relatively novel and promising classification algorithm for pattern recognition. It has not only simpler structure, but also better performance, especially better generation ability. One distinct characteristic of SVM is that it aims to find the hyperplane such that the expected recognition error for the unseen test samples is minimized. Based on the structural risk minimization inductive principle, the SVM adopts a systematic approach to find a linear function with lowest VC dimension. The key factor in SVM is to use kernels to construct nonlinear decision boundary. Different kernels are tried in the following experiments.

For solving c - class (c >2) problems with SVMs there exist two basic strategies: 1) one-against-all, in which q SVMs are trained and each of the SVM separates a single class from all remaining classes; 2) one-against-one, in which c^2 SVMs are trained and each SVM separates a pair of class. The comparison between them, discussed by Chih-Wei Hsu and Chih-Jen Lin [10], indicates that one-against-one is more suitable for practical use than other methods. In this paper SVM with one-against-one strategy in the *LibSVM* [11] is employed.

3   Ant Colony Optimization for Feature Selection

Feature selection algorithms presented in the literature can be classified into two categories based on whether features are done independently of the learning algorithm used to construct the classifier. If feature selection depends on learning algorithm, the approach is referred to as a wrapper model. Otherwise, it is said to a filter model. The approach for feature subset selection proposed in this paper is an instance of the wrapper approach. Before ACO is applied, eigen-features of facial images are extracted by PCA. Every one of them can be regarded as a binary bit. A 1 in a bit position of bit string indicates that an eigen-feature is selected, 0, else. Therefore the optimal eigen-features are represented as bit string, and all images are projected into the subspace constructed by the optimal eigen-features with '1' bits, and form coefficients that will be as input data for the SVM classifier.

The main steps of the proposed method are as follows:

1. Some parameters, such as the population of ant colony and a predetermined number of generations, are initialized;

2. All ant agents start to construct solutions simultaneously;

3. Every ant agent k should visit all of eigen-features and build solutions completely, keeping all the visited eigen-features as binary bits into bit string $Solution_k$;

Compared with a generated random, ant agent k located bit position i, decides whether eigen-feature i is selected or not. If this random is greater than or equal to an exploitable probability parameter *Exploit*, the feature i is chose from the larger between the two following probability parameters:

$$P_1 = [\tau_{i,1}]^\alpha [d_i + 1]^\beta \qquad P_0 = [\tau_{i,0}]^\alpha [d_i]^\beta \tag{1}$$

where $\tau_{i,1}$, $\tau_{i,0}$ represent the pheromone intensities of selected and no-selected eigen-features respectively; $d_i = \sum_{j=1}^{i-1} Solution_{k,j}$, the sum of selected features in which $Solution_{k,j}$ is equal to 1 if the feature j is selected, otherwise, 0; α, β factors respectively governing the influence of pheromone intensity and the number of selected features on this feature's selection. However, if this random is less than *Exploit*, whether the feature selected or not is determined by another random 0 or 1;

5. After all ant agents complete building solutions, the selected features are used to train SVMs. The accuracy rates of SVMs tested by the test image set are our objective function. At this stage the global optimal solution from the beginning of a trial is reserved, and the pheromone intensities of its features are enhanced through the formulae (3) and (4);

6. The pheromone intensity of each eigen-feature becomes:

$$\tau_{i,0} = (1-\rho)\tau_{i,0} + \Delta\tau_{i,0} \qquad \tau_{i,1} = (1-\rho)\tau_{i,1} + \Delta\tau_{i,1} \tag{2}$$

where $\Delta\tau_{i,1} = \sum_{k=1}^{N} \Delta\tau_{i,1}^k$, $\Delta\tau_{i,0} = \sum_{k=1}^{N} \Delta\tau_{i,0}^k$ are defined as the sum of the pheromone intensities of selected and no-selected eigen-features laid by all ant agents, respectively; N the population of ant colony; ρ, represents evaporation rate in order to avoid unlimited accumulation of trails; $\Delta\tau_{i,1}^k, \Delta\tau_{i,0}^k$ are the quantities laid on the feature i by the ant k and are defined as

$$\Delta\tau_{i,1}^k = \begin{cases} ARate_k / Q & \text{if ant } k \text{ choose this feature } i \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

$$\Delta\tau_{i,0}^k = \begin{cases} ARate_k / Q & \text{if ant } k \text{ don't choose this feature } i \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where Q is a constant; $ARate_k$, the accuracy rates of ant k;

7. Determining whether the termination condition is satisfied. If true, the system exits and reports the results of this trial. Otherwise the system reiterates from the second step. In our method the maximum of generations less than 100, and a mechanism that

the global optimal solution should be changed within 30 epochs, form our termination condition. The last and best optimal solution is our target.

From the procedure mentioned above, it can be generalized that the algorithm works by reinforcing portions of solutions that belongs to good solutions, and by selecting eigen-features as few as possible, which can be seen from formula (1).

4   Face Recognition System

A face recognition system based on SVM and ACO, called ACOSVM, is described. Firstly, to reduce the dimensionality of the face images, the PCA coefficients of all images are calculated at this preprocessing stage without some complex image operations such as face alignment or hair exclusion. It can be seen that it is more simple but practical. Secondly, all the experiments are carried out with 5-fold cross validation (k=5). In other words, all cases are use only once as testing and (k-1) times as training. Before feature selection, the eigenvectors are sorted in descending order of eigenvalues, and the first one hundred features are selected, which to be used or not are decided by ACO. After ACO's feature selection, the SVMs are trained and tested for face recognition. Reserving the best optimal features with highest accuracy rate, the next validation is continued. Finally, the average of the optimal best False Acceptance Rates (FAR) is the FAR of the face recognition, and the highest accuracy rate of the five accuracy rates is the best accuracy rate of this system.

5   Experimental Results

To assess the proposed method, experiments are performed using the ORL face database [12]. The ORL face database includes 10 different images of 40 distinct subjects. For some of the subjects, the images are taken at different times, varying lighting slightly, facial expressions (open/closed eyes, smiling/non-smiling) and facial details (glasses/no-glasses). All the images are taken against a dark homogeneous background and the subjects are in up-right, frontal position. The size of each image is 92x112, 8-bit grayscale levels. For the efficiency and convenience of computation, every image in this database is sized to 50 by 50, and the contrast of an intensity image is enhanced using histogram equalization, and the gray value is normalized to lie within [0,1]. After that, the PCA is executed, and the feature selection and test is repeated. Some parameters in ACO considered here are: N =100, α =1 , β =-1 , ρ =0.1, Q =10.0, *Exploit* =0.9, the initial pheromone intensity of each features is equal to 10. 0. We have made no serious attempt to optimize the setting of these parameters. Such an optimization should be tried in future research. We compare ACOSVM classifiers with SVM classifiers. The experimental results are given in Table 1.

Note that the SVMs with the second, third, forth order polynomial kernels are denoted by SVM (Poly q=2,3,4) respectively in Table 1. It can be concluded from the table that ACOSVMs obtained lower FAR while compared with other SVMs without

selected features, and some of their best recognition accuracy rate rich 100%. Therefore the system guarantees good generalization. The obtained results are also compared with other approaches in literatures on the ORL database. It is summarized in the Table 2. From the Table 2, it can be also seen that the recognition rates of our approaches are superior to other approaches mentioned above on the ORL database.

6   Conclusion

We have proposed a face recognition system based on Ant Colony Optimization and Support Vector Machine. Comparative experiments show that ACOSVM has a much better performance than others without feature selection. By selecting the feature subset with ACO, the computational cost is decreased, and the ability of face recognition is also improved. From the image process and experimental results it shows that the ACOSVM is more efficient and practical for its application. But the main drawback will be the computational cost by using SVM with one-against-one strategy when the subjects are too large. Regardless of this disadvantage the proposed method for feature selection is very promising. In future, comparison between ACO and other optimization methods in face recognition, and more experimental results will be investigated. Of course, the application domain of the proposed method is not limited in the problem of face recognition. It can be applied to many problems such as face detection, facial expression recognition, fingerprint classification and other pattern recognitions.

Table 1. Comparative results on the ORL face database

Approaches	FAR	Best recognition rate
SVM (Linear)	0.0325	0.9750
SVM (Poly q = 2)	0.030	0.9875
SVM (Poly q = 3)	0.030	0.9875
SVM (Poly q=4)	0.030	0.9875
SVM (Radial Basis Function)	0.030	0.9875
ACOSVM (Linear)	0.010	1.0000
ACOSVM (Poly q = 2)	0.0125	1.0000
ACOSVM (Poly q = 3)	0.0125	0.9875
ACOSVM (Poly q=4)	0.0125	1.0000
ACOSVM (Radial Basis Function)	0.0125	0.9875

Table 2. Comparison with other face recognition methods on the ORL face database

Approaches	Recognition rate
Kernel PCA [13]	0.975
KPCA+GAs [14]	0.9837
SVM [15]	0.98
SOM+CN [16]	0.962

Acknowledgements

Reference

[1]   Chellappa, R.,Wilson, C.L. and Sirohey, S. Human and machine recognition of faces: a survey, in: Proceedings of the IEEE, 83(5) (1995) 705 –741.

[2]   W. A. Barrett, A survey of face recognition algorithms and testing results, in: Conference Record of the Thirty-First Asilomar Conference on Signals, Systems & Computers, 1997, pp. 301-305.

[3]   M. Dorigo, V. Maniezzo, and A. Colorni, The ant system: optimization by a colony of cooperating agents, IEEE Transactions on Systems Man , and Cybernetics, Part B, 26(1) (1996): 9 - 41.

[4]   M. Dorigo, G. Di Caro, and L. M. Gambardella, Ant algorithms for discrete optimization, Artificial Life, 5(2) (1999): 137 – 172.

[5]   E.Bonabeau, M.Dorigo and G.Theraulaz, Inspiration for optimization from social insect behaviour, Nature, 406 (2000) 39 - 42.

[6]   Zhong Yan and Chunwei Yuan. Ant colony optimization for navigating complex labyrinths, in: Proceedings of the 9th international Conference, RSFDGrC 2003, Chongqing, China, May, 2003, pp. 445 – 448.

[7]   M. Dorigo and G. Di Caro, Ant colony optimization: a new meta-heuristic, in: Proceedings of the 1999 Congress on Evolutionary Computation, 1999, pp. 1470 - 1477.

[8]   Christopher J. C. Burges, A Tutorial on support vector machines for pattern recognition, Data Mining and Knowledge Discovery, 2 (1998), 121 - 167.

[9]   Lei Zhang, Fuzong Lin and Bo Zhang, Support vector machine learning for image retrieval, in: Proceedings of 2001 International Conference on Image Processing, 2001, pp. 721 - 724.

[10]   C.-W. Hsu and C.-J. Lin, A comparison of methods for multi-class support vector machines, in: Proceedings of the IEEE Transactions on Neural Networks, 13 (2002) 415 - 425.

[11]   Chih-Chung Chang and Chih-Jen Lin, LIBSVM: a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[12]   F. Samaria and A. Harter, Parameterisation of a stochastic model for human face identification, 2nd IEEE Workshop on Applications of Computer Vision, December 1994, Sarasota (Florida).

[13]   Kwang In Kim, Keechul Jung and Hang Joon Kim, Face recognition using kernel principal component analysis, IEEE Signal Processing Letters, 9 (2) (2002): 40 – 42.

[14]   Kim, K.I., Kim, J. and Jung, K., Recognition of facial images using support vector machines, in: Proceedings of the 11th IEEE Signal Processing Workshop on Statistical Signal Processing, 2001, pp. 468 – 471.

[15]   Zhang Yank and Liu Chongqing, Face recognition using kernel principal component analysis and genetic algorithms, in: Proceedings of the 2002 12th IEEE Workshop on Neural Networks for Signal Processing, 2002, pp. 337 –343.

[16]   Lawrence, S., Giles, C.L., Ah Chung Tsoi and Back, A.D., Face recognition: a convolutional neural-network approach, IEEE Transactions on Neural Networks, 8 (1) (1997): 98 – 113.

Face Recognition from Color Images in Presence of Dynamic Orientations and Illumination Conditions

Xiabo Yu and George Baciu

GAMA Lab, Department of ComputingThe Hong Kong Polytechnic University
{csxbyu,csgeorge}@comp.polyu.edu.hk

Abstract. Current face recognition systems work well in certain controlled conditions with frontal images. However, they do not take some unpredictable factors into consideration. For example, in dynamic environments with complex backgrounds and textures, or when the illumination or contrast is relatively low, most methods fail to perform accurate feature recognition. In this paper, we propose a robust face recognition system based on color reference model, efficient light bias method, face morphing and independent component analysis. This system performs face recognition against complex dynamic backgrounds and from variable angles of view. Our experimental results show recognition rates of over 92% using an optimized ICA algorithm.

1   Introduction

The human face is one of the most common features that can be used to identify and authenticate people quickly and non-invasively. In this respect, computer vision-based face recognition systems are effective with mediating and aiding both security and commercial authentication applications.

   Many face recognition methods have been proposed, such as eigenfaces [1], holons [2], and local feature analysis based on Principle Component Analysis (PCA) [3]. The PCA recognition ratio is satisfactory when tested on controlled face databases such as FERET and AR-face. In 1998, Bartlett et al. proposed a face recognition method that used Independent Component Analysis (ICA) [4][5]. ICA has the advantage that it properly separates high-order dependencies that would still exist in the joint distribution of the PCA coefficients. The main goal of the ICA method is to recognize people with different expressions and to recognize different views of the same person taken at different times.

   In practice, the locations of facial features may not be available or it may not be possible to obtain a direct front view of a face. In order to recognize a person dynamically, we need to first distinguish the human face and facial features in an image, then normalize and adjust the test image so that it has the same orientation as a candidate face in the database. The last step is to use an efficient face recognition method to classify and identify faces.

   In this paper, we proposed an illumination bias correction method, and discuss the problem of extracting facial feature information in the presence of non-ideal sample images taken from arbitrary orientations and background conditions. We also present

new experimental evidence for the application of ICA to face identification in the presence of noise and non-direct face views.

2   Face Detection

Many face detection approaches have been proposed in the research literature. Among these there are knowledge-based methods [6], feature invariant approaches [7], template matching methods [8] and appearance-based methods [9]. Many technical articles have discussed the adoption of a face color model as an efficient way to detect possible face areas [10].

2.1 Face Color Model

Skin color does not vary greatly from one person to another. Projecting the face colors of different persons onto a color space, they should congregate in blocks that form a color cluster. We propose a face color model that is adaptable to a given database or a specific reference subset of faces. This method can eliminate the influence of skin color diversity as for example, amongst Asians, Hispanics or Caucasians. Although, the number of candidate face images that are not images of real faces may increase, such images will be removed after the face feature extraction phase.

Modeling skin color requires choosing an appropriate color space. We have adopted the YC_bC_r space, which separates colors into chrominance and luminance, greatly reducing the effect of background lighting on a face.

Skin-tone color depends on lighting conditions. The colored light will make the projection of skin-tone color deviate from its original position which leads to incorrect matching. Generally speaking, the integrated affection of environment light on different parts of image is coherent and uniform when there is more than one light source. This global affection of light could be detected if there are real white and black objects in images because their original color components (RGB) values should be equal. Images of human faces will usually contain a certain proportion of "real white" [11] and "real black" pixels in locations such as eyes, and the luminance values of these pixels always lie respectively in the top and bottom. We define the pixels with the top 3 percent luminance values as the reference white and pixels with bottom 3 percent luminance values as reference white. By changing the RGB values of the real white regions to (255,255,255) and real black regions to (0,0,0) and using this as reference to normalize RGB values of pixels of the whole image, we can partially eliminate the color bias.

In Fig.1, the top left picture is the original picture and the bottom left picture is corresponding projection of all its pixels on the C_bC_r plane. The right two pictures are after light bias. Apparently, the top right picture looks more natural and its corresponding projection shows two distinct clusters of the skin-tone color and the background. The most important thing is that the background cluster (enclosed by red circle) is moved out of the possible face color region (the red and yellow areas on the C_bC_r plane). This makes the extraction of faces easier.
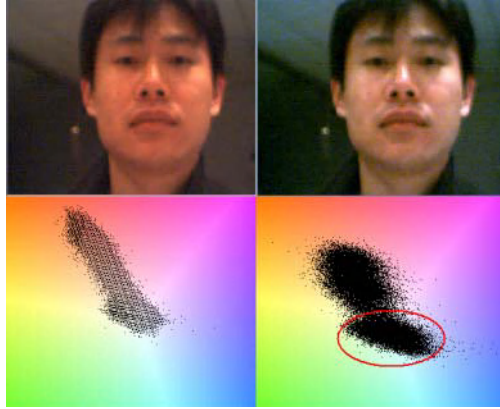
Fig. 1. Light bias using reference white and reference black

Fig. 2. Determining face images

After removing of light bias, we sample the face database to gather a set of skin colors pixels which are projected onto the C_bC_r chrominance plane. The chrominance plane is partitioned into $N \times N$ grids. The grid $P(i, j)$ is a face color grid, if $P(i, j) > P_{num}$, (0<=i ,j<=N). Here, $P(i,j)$ is the number of face color pixels in grid(i,j), P_{num} is a fixed threshold. Pixels of the input image are also projected onto the C_bC_r plane. If one pixel falls into a face color grid, it is regarded as a face color pixel. The input image is then divided into face color areas and non-face color areas. Fig.2 shows the process of determining candidate face images from a color image.

2.2 Extraction of Facial Features

The eyes and mouth have unique characteristics of color and texture. Hsu [12] suggested that if we combine and emphasize these characteristics, facial features can be located and extracted from the input images. A candidate face image is considered to be a face image if both eyes and mouth can be found from it. A very efficient facial feature detection method using morphological filter is proposed by Hsu in 2002. In order to achieve better detection result, an improved facial feature detection method is discussed in this paper Based on Hsu's method. The result of eye and mouth extraction is shown in Fig. 2.

As previously mentioned, the YC_bC_r space separates color into the chrominance and luminance. Eyes have high C_b and low C_r values. Eyes also have a unique

luminance texture in that dark and bright pixels are mixed together. This means that we can extract eyes correctly by abstracting using them using a morphological filter with an appropriate structuring element. Comparing to Hsu's method, we make three improvements:

1. Along the contour of face usually contains same bright and darkness mixing texture as eyes, which can be falsely detected as possible eye areas after morphological filtering. In order to avoid this error, a face mask is used to separate the face area from its background which is usually darker than the face contour.
2. Instead of using a semi-sphere structure element, a cylinder structure element is applied when doing morphological filter. The latter is more efficient in emphasizing the texture of eye and can avoid the effect of small area on the face with similar texture of eyes.
3. In Hsu's method, three elements ($C_b^2, -C_r^2, C_b/C_r$) are used to construct the eye chrominance model. In fact, the third element C_b/C_r is most important and can supply enough information of existence of eye while the other two elements may reduce the contrast between eyes and the other parts of face. So only normalized C_b/C_r is involved in our method.

The color of the mouth region contains a stronger red component and a weaker blue component than other facial regions Therefore, its chrominance component C_r is greater than C_b. The mouth chrominance model proposed in Hsu's method is applied and also, like in eye detection, a face mask is used before the mouth detection and a cylinder structure element is used in morphological filter.

3   Face Morphing

Face rotation can adversely affect the recognition rate. In order to reduce such influences, our method uses image-based morphing before the recognition stage which is much faster than a full 3D reconstruction and re-projection.

Three kinds of rotation in 3D space are defined: 1. tilt, 2. horizontal rotation, and 3. vertical rotation with Z, Y and X acting as the respective axes of rotation. Rotation effects can be reduced via warping and rotating the test 2D face image, using facial features as a reference in order to make the position of each part of the face correspond as closely as possible to those of the normalized images saved in the face database. Fig.3 illustrates the process of face morphing. A, B and D are the coordinates for the eyes and mouth, and $AD/BD = AC/BC$. First, the image is rotated to make CD vertical. Next, using C as the center, BC is rotated in the XOY plane to make it horizontal. Finally, the left/right face is either expanded or shrunk to make $AC = BC$. This morphing operation simulates in the 2D image-plane the rotation of a face in the 3D environment (shown in Fig. 4).

Fig. 3. The result of eye and mouth extraction
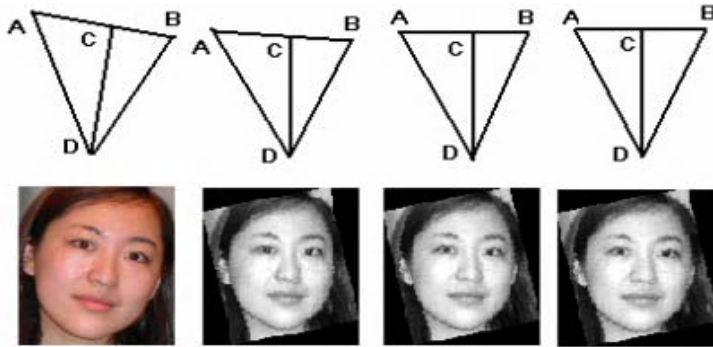
Fig. 4. Face morphing

4   Face Recognition

ICA is a statistical method that attempts to find underlying factors or components from multivariate statistical data [13]. There are a number of algorithms for efficiently computing the IC's [14]. We choose the info-max algorithm proposed by Bell and Sejnowski [15] which has been successful in separating randomly mixed auditory signals, electroencephalogram signals, and functional magnetic resonance imaging signals.

4.1 Experiments on Robustness of Face Recognition Using ICA

In this section we conduct studies on the effectiveness of the ICA method and seek improvements for the face recognition rate. We focus on six parameters: *horizontal translation, vertical translation*, *tilt*, *scale*, *rotation,* and *down-sampling* [16]. An experimental face image database including 57 images (one normalized frontal image for one person) is built in this test. Table 1 summarizes the limits of independent parameter values. Within these limits, face recognition using ICA is sufficiently robust to achieve a high recognition rate of over 92%.

Table 1. Experimental parameter limits

	Limits
Horizontal translation	$\pm$ 3% of Width
Vertical translation	$\pm$ 5% of Height
Tilt	$\pm$ 5 degrees
Scale	$\pm$ 5%
Rotation	<15%
Down-sampling	<40%

Face morphing can decrease the effect of *rotation* error. Morphed images may, however, introduce distortions when rotation angles are large, i.e. greater than 45deg. The efficiency of the face morphing may be further improved by determining the rotation angle that has a significant effect on the ICA decomposition. We recall that the ICA requires a set of images that forms a basis set. Then, the problem extends to finding a basis set that has the least impact on rotation. This, however, is an open problem.

4.2 Optimized Face Recognition Using ICA

In experiment, we find that when the rotation angle grows (from 25 degrees to 45 degrees), the recognition ratio decreases greatly (from 82.1% to 25.1%, from 89.3% to 50.7%). These ratios are lower because of the large distance between (1) V_1 the vector resulting from projecting the test image onto ICs space, and (2) V_2 the vector representing the corresponding image in the training set.

As the rotation angle increases, the average distance between V_1 and V_2 will also increase. When the rotation angle exceeds the limiting points for face morphing, the probability that V_1 falls into the sub-space of a class which it should belong to will be much smaller. Thus the original IC set cannot classify correctly even after face morphing. This can result in a low recognition ratio. This also implies that there are more independent components between the test images and the training image set and more information is needed in the training set. Due to the asymmetry of left and right/down and up faces, we choose five images for each person as the new training set for ICA: front, 45° and -45° horizontal rotation, 45° and -45° vertical rotation.

As we tested the new method on images (13 persons and 12 images of each person) with only horizontal or vertical rotations, we discovered that the recognition ratio is raised to 94.9%. The recognition ratio for images (13 persons and 12 images of each person) with both horizontal and vertical rotations is 85.7%. These results indicate that morphing can satisfy the requirements for detecting non-frontal face views in the presence of noise without recourse to 3D face reconstruction which would otherwise require a complex pixel depth map computation.

5   Conclusions

We have presented a dynamic face recognition model that can significantly improve on available systems. This system first automatically locates the position of the

human face and facial features. It then corrects the rotation errors using pseudo-morphing in 2D, a preprocessing method selected from a variety of face morphing methods. The recognition ratio is further increased using an optimized ICA whose proposed training set includes five multi-angles Results show that this method is robust and effective in the presence of large rotation angles arbitrary orientations.

References

[1]   M.Turk and A.Pentland, Eigenfaces for recognition, J.Conitive Meurosi., Vol. 1. (1991) 71-86

[2]   G.Cottrell, J.Metcalfe: Face, gender and emotion recognition using holons, In D.Touretzky, San Mateo (eds): Advances in Neutral Information Processing Systems, Vol. 21. CA: Morgan Kaufmann (1991) 974-989

[3]   P.S.Penv, J.J.Atick: Local feature analysis: A general statistical theory for object representation, Network: Comput. Neutral Syst., Vol. 7, No. 3. (1996) 477-500

[4]   M.Bartlett, H.Lades, T.Sejnowski: Independent component representation for face recognition, Proceeding of the SPIE Symposium on Electronic Image: Human Vision and Electronic Imaging. SPIE Press (1998) 32-99

[5]   M.Bartlett, J.Movellan, J.sejnowski: Face recognition by in dependent component Analysis, IEEE Transaction on Neutral Network, Vol. 13, No. 6. (2002) 1450-1464

[6]   G.Yang and T,S.Huang, Human Face Detection in ComplexBackground, IEEE Transaction on Pattern Recognition, Vol. 27, No. 1. (1994) 53-63

[7]   T.K.Leung, M.C.Burl, P.perona: Finding faces in cluttered scenes using labeled graph matching, Proc. 15th IEEE int'l Conf. Computer Vision. (1995) 637-644

[8]   I.Craw, D.Tock, A.Bennet: Finding face features, Proc. Second European Conf. Computer Vision. (1992) 92-96

[9]   K.-K.Sung, T.Poggio: Example-based learning for view-based human face detection, Technical Report Al Memo. Massachusetts Inst. Of Technology AI Lab (1994) 1521

[10] R.feraud, O.J.Bernier, J.-E.Viallet, M.collobert: A fast and accurate face detection based on neutral network, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 23, No. (2001) 42-53

[11] C.A.Poynton: A technical introduction to digital video, John Wiley & Sons (1996)

[12] R.L Hsu, M.Abdel-Mottaleb, A.K.Jain: Face detection in color Images, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 24, No. 5. (2002) 696-706

[13] A. Hyvärinen, J. Karhunen, E. Oja: Independent Component Analysis. John Wiley & Sons, Inc. (2001)

[14] M.Girolami: Advances in independent component analysis, Berlin, Germany, Springer-verlag (2000)

[15] A.J.Bell, J.Segnowski: An information-maximization approach to blind separation and blind deconvolution, Neutral Comput., Vol 7, No. 6. (1995)

[16] A. Lemieux, M. Parizeau: Experiments on Eigenfaces Robustness. IEEE Proceedings of Pattern Recognition. Vol 1, 421-424

Dynamic Local Feature Analysis for Face Recognition

Johnny Ng and Humphrey Cheung

Titanium Technology Research Centre
10/F, Tianjin Building, 167 Connaught Road West, Hong Kong, PR China
`{Johnny.ng,Humphrey.cheung}@titanium-tech.com`

Abstract. This paper introduces an innovative method, *Dynamic Local Feature Analysis* (*DLFA*), for human face recognition. In our proposed method, the face shape and the facial texture information are combined together by using the Local Feature Analysis (LFA) technique. The shape information is obtained by using our proposed adaptive edge detecting method that can reduce the effect on different lighting conditions, while the texture information provides the details of the normalized facial feature on the image. Finally, both the shape and texture information is combined together by means of LFA for dimension reduction. As a result, a high recognition rate is achieved no matter the face is enrolled under different or bad lighting conditions.

1 Introduction

Face recognition has become one of most important biometrics authentication technologies during the past few years. There are at least two reasons that can explain why face recognition has received extensive attention: 1) Face recognition has many applications, such as the biometrics system, the content-based video processing system, and law enforcement system. A strong need for a robust automatic face recognition system is obvious due to the widespread use of photo-ID for personal identification and security. 2) Although there are some extremely reliable methods of biometric personal identification existed such as fingerprint scans and iris scans, face recognition can still be effective because it does not require the cooperation or any other special knowledge of the participant. Besides that, building an automatic face recognition system normally is cheaper than building a personal identification system based on fingerprint scans or iris scans.

The human face is a highly variable objects, it is very difficult to develop a fast and robust face recognition system. Since the recognition rate may be affected by the presence of glasses, facial hairs, facial expression, lighting conditions, etc. In order to reduce the above problems, we combined the texture and the shape information by using the Local Feature Analysis (LFA) technique to develop a robust face recognition algorithm. The shape information is obtained by using our proposed adaptive edge detecting method that can reduce the effect on different lighting conditions, while the texture information provides the details of the normalized facial feature on the image. The organization of this paper is as follows. In Section 2, a literature review is presented to summaries the recent works by other researchers. In Section 3, we present our proposed Dynamic Local Feature Analysis method, which is based on the LFA formalism combined with the texture space and the edge

information. In Section 4, the experimental result of our proposed method will be presented to support the significance of this research. Finally, a conclusion is given in Section 5.

2   Literature Review

In 1995, Chellappa [1] summarized the existing techniques for human face recognition. Later, Zhao, Chellappa and Rosenfeld provide a more details information about human face recognition in a technical report [2]. From both reports, we can find that most of the current face recognition algorithms can be categorized into two classes, image template based or geometry feature-based. The template-based methods [3] compute the correlation between a face and one or more model templates to estimate the face identity. Statistical tools such as Support Vector Machines (SVM) [5], Linear Discriminant Analysis (LDA) [4], Principal Component Analysis (PCA) [6], Kernel methods [7], and Neural Networks [10] have been used to construct a suitable set of face templates. Wiskott *et al*. developed an elastic Bunch graph matching algorithm for face recognition in [8]. However, it may not achieve a high recognition rate if the enrollment process and the verification process are not done in the same lighting conditions. Thus, Penev *et. al* [9] proposed the LFA by modifying the PCA in order to solve the above problems. Local Feature Analysis (LFA) is a derivative of the eigenface method. Local Feature Analysis utilizes specific features for identification instead of the entire representation of the face. The system selects specific areas of the face, such as the eyes or mouth, to use as the defining features for recognition.

3   Dynamic Local Feature Analysis

In this section, a novel face recognition technique – Dynamic Local Feature Analysis (DLFA) is illustrated. Our approach can be divided into two main steps. The first step is preprocessing. The goal of this step is to get rid of high intensity noise, transform the input face image into a binary one by adaptive edge detection and then extract the texture of the face. The second step employs the local feature analysis to combine both edge of face shape and the texture.

(a)   (b)   (c)   (d)
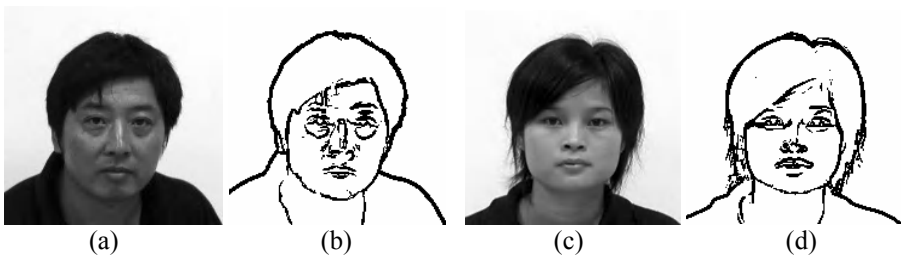
Fig. 1. Some of adaptive edge analysis result. (a) and (c): original facial images; (b) and (d): binary edge

3.1 Preprocessing

In general, a face contains four main organs, i.e. eyebrows, eye, nose, and mouth; it is very important in the face recognition system. In order to reduce the noise and some dark features around these four organs, an opening operation is employed on the input face image to achieve this propose. This operation can prevent the facial features from breaking into pieces, sharps and bright noises such as reflections on eyes.

In order to properly generate a binary edge image from an input face image, a adaptive edge analysis is proposed in our approach. The advantage of using edge as one of the image feature is that they can provide robustness to illumination change and simplicity of presentation. The input face image is firstly processed with morphological edge detection and then converts the resulted gray scale edge image into binary format. A fixed threshold does not work well for converting a gray scale edge image to a binary edge image. This is due to the fact that the contrast in edge image may vary significantly. In our approach, we use a dynamic threshold in each sub-block of edge image to obtain the corresponding threshold binary image. The threshold T is calculated dynamically in each sub-block of face image by considering the gray-level intensities of the 15% highest pixel intensities. Assume that the histogram of each sub-block of the edge image of size $his(i)$, where i = 0,1,…,255. Then T is determined as the largest value such that the following equation is satisfied:

$$\sum_{i=T}^{255} his(i) \geq 0.15 h \times w \tag{1}$$

where h and w is the high and the width of each sub-block of an edge image respectively. Fig. 2 shows some of adaptive edge analysis result.

3.2 Local Feature Analysis

Local Feature Analysis (LFA) defines a set of topographic, local kernels that are optimally matched to the second-order statistics of the input ensemble [15]. The kernels are derived from the principal component axes, and consist of "sphering" the PCA coefficients to equalize their variance, followed by a rotation to pixel space. We begin with the zero-mean matrix of original images, X. The X is the texture information of the training image and then follow is the edge information. Then, calculate the principal component eigenvectors P according to $S = PDP^T$. Penev and Atick [15] defined a set of kernels, K as

$$K = PVP^T \tag{2}$$

$$\text{where } V = D^{-\frac{1}{2}} = diag\left(\frac{1}{\sqrt{\lambda_i}}\right) \quad i = 1, …, p$$

where λ_i are the eigenvalues of S. The rows of K contain the kernels. The kernels were found to have spatially local properties and are "topographic" in the sense that they are indexed by spatial location. The kernel matrix K transforms X to the LFA output $O = KX^T$. Note that the matrix V is the inverse square root of the covariance matrix of the principal component coefficients. This transform spheres the principal component coefficients (normalizes their output variance to unity) and minimizes

correlations in the LFA output. Another way to interpret the LFA output O is that it is the image reconstruction using sphered PCA coefficients, $O = P(VP^TX^T)$.

3.3 Sparsification of LFA

LFA produces an n-dimensional representation, where n is the number of pixels in the images. Since we have n outputs described by $p \ll n$ linearly independent variables, there are residual correlations in the output. Penev and Atick presented an algorithm for reducing the dimensionality of the representation by choosing a subset M of outputs that were as decorrelated as possible. The sparsification algorithm was an iterative algorithm based on multiple linear regression. At each time step, the output point that was predicted most poorly by multiple linear regression on the points in M was added to M. Due to the topographic property of the kernels, selection of output points was equivalent to selection of kernels for the representation. The methods addressed image representation but did not address recognition. The sparsification algorithm in selected a different set of kernels, M, for each image, which is problematic for recognition. In order to make the representation amenable to recognition, we selected a single set M of kernels for all images. At each time step, the kernel corresponding to the pixel with the largest mean reconstruction error across all images was added to M. At each step, the kernel added to M is chosen as the kernel corresponding to location

$$\arg\max \left\langle \left\| O - O^{rec} \right\|^2 \right\rangle \tag{3}$$

where O^{rec} is a reconstruction of the complete output, O, using a linear predictor on the subset of the outputs O generated from the kernels in M. The linear predictor is of the form:

$$Y = \beta X \tag{4}$$

where $Y = O^{rec}$, β is the vector of the regression parameters, and $X = O(M,N)$. Here, $O(M,N)$ denotes the subset of O corresponding to the points in M for all N images. β is calculated from:

$$\beta = \frac{YX}{(X^T X)} = \frac{(O^{rec})^T O(M,N)}{O(M,N)^T O(M,N)} \tag{5}$$

Equation (5) can also be expressed in terms of the correlation matrix of the outputs, $C = O^TO$,

$$\beta = C(M,N)C(M,M)^{-1} \tag{6}$$

the termination condition was $|M| = N$.

4   Experimental Results

In order to prove the efficiency and accuracy of our face recognition method, we used a number of face databases, which included face images taken under roughly controlled imaging conditions as test images. The ORL face database (from the Oliver Research Laboratory in Cambridge, UK), Yale face database (from the Yale University), MIT face database, and Ti-Lab database were used in the experiments.

The MIT database has 144 face images with 16 distinct subjects, while the Yale database has 150 face images with 15 distinct subjects. The Ti-Lab database has 58,960 face images with 5,896 subjects. For the ORL database, there are 400 different face images corresponding to 40 distinct persons, but only six images for each of the 40 subjects were included in the testing set. For the FERET database, there are over 6,000 different face images corresponding to 699 distinct persons, but only 289 subjects were included in the testing set.

The experimental setup consisted of an upright frontal view of each of the subjects with a suitable scale and a normal facial expression was chosen to form a database. In our system, the position of the two eyes in a face image is manually located firstly. Then based on the eye positions, all the facial images in the database and the query input are normalized to a size of 80×80 with 256 gray scale levels. Since the images are acquired from different face databases, the pose variations, the lighting conditions and the facial expressions may vary (see Fig. 3). The number of subjects and number of testing images for the respective databases are tabulated in Table 1. The performances of the Dynamic Local Feature Analysis technique was evaluated based on the respective database.

In the experiment, we implemented and evaluated the relative performances of the Dynamic Local Feature Analysis (DLFA) and Local Feature Analysis (LFA) technique. A query image is compared to all of the face images in the database and the face images are then ranked and arranged in ascending order according to their corresponding measured euclidean distances. Table 2 tabulates the recognition rates of DLFA and LFA for each of the five individual databases. Experimental results show that the DLFA outperforms LFA method.

The recognition rates of DLFA based on the ORL, Yale, MIT, FERET, Ti-Lab are 91%, 89%, 92%, 90%, and 87%, respectively. Fig. 4. illustrates the cumulative recognition rates of the both techniques based on the five face databases. The experiments were conducted on a Pentium IV 2.4 GHz computer system. The average runtimes for the DLFA are about 345ms.

Table 1. The number of faces in the ORL, Yale, MIT, FERET, Ti-Lab database

	ORL	Yale	MIT	FERET	Ti-Lab	Total
Subject	40	15	16	289	5,896	6,256
Testing set	240	150	144	1,445	58,960	60,939

Fig. 2. Sample face images of Ti-Lab face database used in our experiments

Table 2. Recognition rates of the LFA and DLFA face recognition techniques with different database

Method	LFA (%)	DLFA (%)
ORL database	63	91
Yale database	62	89
MIT database	58	92
FERET database	61	90
Ti-Lab database	62	87
Average Recognition Rate	61.2	89.8
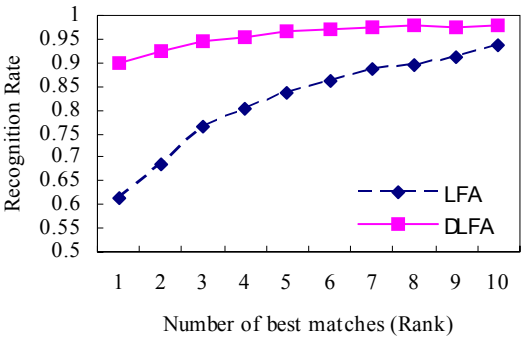
Fig. 3. Comparison of the overall recognition rates

5   Conclusions

In this paper, a robust face recognition method, Dynamic Local Feature Analysis (DLFA), which employs the LFA on integrated with the face shape, is proposed. The main idea of DLFA focuses on the individual features such as the eyes, the nose, the mouth and areas of definite bone curvature differences, such as the cheeks. The DLFA approach offers several advantages over similar facial recognition technologies. DLFA systems are not as sensitive to face deformities and lighting variations as the eigenface method. Also, DLFA systems can recognize faces that are not presented to the camera as a complete frontal view. The system can recognize an individual facing up to 25 degrees away from the camera. Experimental results based on a combination of the MIT, Yale, FERET, Ti-Lab, and ORL databases show that DLFA can achieve recognition rates of 89.8%, 94.7% and 96.6% for the first one, the first three and the first five likely matched faces, respectively. The technique in this paper is computationally simple and can provide a reasonable performance level. In practice, our approach can be used as a robust human face recognition system, which selects those similar faces to an input face from a large face database.

References

[1]   R. Chellappa, C.L. Wilson, and S. Sirohey, Human and Machine Recognition of Faces, A Survey, *Proc, IEEE*, Vol. 83, pp. 705-740, 1995.

[2]   W. Zhao, R. Chellappa, A. Rosenfeld, and P.J. Phillips, Face Recognition: A Literature Survey, UMD CFAR Technical Report CAR-TR-948, 2000.

[3]   Robert J. Baron. Mechanisms of human facial recognition. *International Journal of Man-Machine Studies*, 15(2):137–178, 1981.

[4]   P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern*

[5]   *Analysis and Machine Intelligence*, 19(7):711–720, July 1997.

[6]   Matthew Turk and Alex Paul Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[7]   B. Schoelkopf, A. Smola, and K.-R. Muller. Kernal principal component analysis. In *Artificial Neural Networks ICANN97*, 1997.

[8]   Laurenz Wiskott, Jean-Marc Fellous, Norbert Kr¨uger, and Christoph von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):775–779, July 1997.

[9]   P. Penev and J. Atick. Local feature analysis: A general statistical theory for object representation, 1996.

[10]   Jonathan Howell and Hilary Buxton. Invariance in radial basis function neural networks in human face classification. *Neural Processing Letters*, 2(3):26–30, 1995.

A Fingerprint Matching
Using Minutiae Triangulation

Giuseppe Parziale and Albert Niel

Institute of Digital Image Processing, Joanneum Research
Wastiangasse 6, A-8010, Graz, Austria
{giuseppe.parziale,albert.niel}@joanneum.at

Abstract. We present a new technique for fingerprint minutiae matching. The proposed method connects minutiae using a Delaunay triangulation and analyzes the relative position and orientation of each minutia with respect to its neighbors obtained by the triangle structure. Due to non-linear deformations, we admit a certain degree of triangle deformation. If rotations and translations are present, the triangle structure does not change consistently. Two fingerprints are considered matching, if their triangle structures are similar according the neighbor relationship. The algorithm performance are evaluated on a public domain database.

1   Introduction

Fingerprints are the most reliable human characteristics that can be used for people identification [1, 2]. A fingerprint pattern is composed by ridges and valleys. Ridges present various kinds of discontinuities (minutiae), able to capture the invariant and discriminatory information, used to recognize fingerprints. In automatic fingerprint recognition systems, only ridge bifurcations and ridge endings are commonly used.

Fingerprint matching techniques can be coarsely classified in three groups: minutiae-based, correlation-based and ridge feature-based matching techniques. For an overview of the three methods, refer to [4]. In this work, we will focus on the first class of methods that is also the most popular and widely used [1, 6, 7, 8, 13, 14, 15, 16]. In general, this kind of techniques performs the matching comparing two sets of points (minutiae) and looking for the transformation that establishes the correspondences, translation and rotation between the query and the template point set. Once the minutiae features are extracted from the image, the fingerprint matching becomes a non-rigid point-matching problem [4] with unknown point correspondences and difference in the number of points belonging to the two sets. Moreover, the skin elasticity changes the relative position of the minutiae at each acquisition. In this paper, we present a novel technique for fingerprint matching, taking advantage of the relative position and orientation of each minutia related to their neighbors.

The paper is organized as follows. In the next Section 2, we state the general case of the non-rigid point-matching problem and how it is addressed for

the fingerprint matching. In Section 3, we describe our solution. In Section 4, we give a brief description of the method used for the minutiae extraction and present the algorithm performance computed on a public domain database. Finally, concluding remark are presented in Section 5.

2   Statement of the Problem

Let $Q = \{m_1^Q, m_2^Q, \ldots, m_M^Q\} \subset R^2$ and $T = \{m_1^T, m_2^T, \ldots, m_N^T\} \subset R^2$ represent the sets of 2-D points extracted from the query and the template image, respectively. In general, a similarity transformation [5] $Tr_{s,\theta,t_x,t_y} : Q' \to T'$ that maps each point of a subset $Q' \subseteq Q$ with each point of a subset $T' \subseteq T$ has to be found. Then, $\forall m_j^{Q'}, \exists$ a unique $m_i^{T'}$ such that,

$$Tr\begin{pmatrix} x_j^{Q'} \\ y_j^{Q'} \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \end{pmatrix} + \begin{pmatrix} \cos\vartheta & -\sin\vartheta \\ \sin\vartheta & \cos\vartheta \end{pmatrix} \begin{pmatrix} x_i^{T'} \\ y_i^{T'} \end{pmatrix} \tag{1}$$

where $\vartheta \in [0, 2\pi[$ is a rotation angle and t_x and t_y are the translations along the x and y axes, respectively. The correspondence between points of the two sets is unknown, i.e. the number of the common points of Q' and T' is unknown. To reduce the problem complexity, the local ridge orientation $\theta \in [0, \pi[$, which the minutia belongs to, is introduced as an extra information. Thus, denoting with $m_i = \{x_i, y_i, \theta_i\}$ a minutia represented by its location and its orientation, we say that $m_i^{Q'} \in Q' \subset Q$ and $m_j^{T'} \in T' \subset T$ match, if the Euclidean distance ED is smaller than a given threshold d_0 and the angular difference AD is smaller than an angular tolerance θ_0 [4]:

$$ED(m_j^{Q'}, m_i^{T'}) = \left| Tr\begin{pmatrix} x_j^{Q'} \\ y_j^{Q'} \end{pmatrix} - \begin{pmatrix} x_i^{T'} \\ y_i^{T'} \end{pmatrix} \right| \le d_0 \tag{2}$$

$$AD(m_j^{Q'}, m_i^{T'}) = min(|\theta_j^{Q'}\theta_i^{T'}|, \pi - |\theta_j^{Q'} - \theta_i^{T'}|) \tag{3}$$

The minimum value in (3) is considered because of the angle circularity and we represent the minutia orientation with $0 \le \theta < \pi$. The hyper-sphere defined by d_0 and θ_0 is necessary, because of the presence of local non-linear deformations and errors induced by the minutiae extraction algorithms.

3   Our Approach

The basic idea of our approach is based on the observation that, even if deformations, rotations and translations are applied to a fingerprint image, every minutia keeps always the same neighbor structure. We use this idea, introducing the information related to the position and orientation of each minutia with respect to its neighbors. In [7] a similar approach is presented. There, the local structure represented by a minutia and its k-nearest neighbors is examined, but
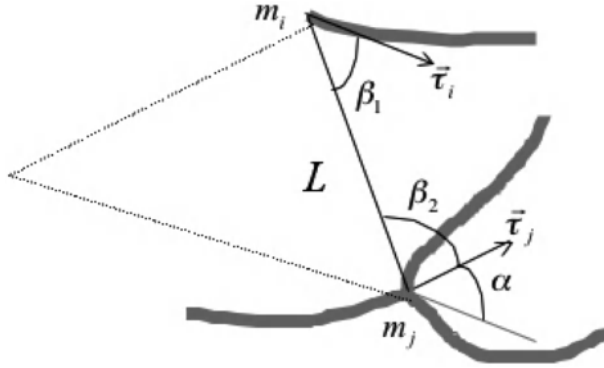
Fig. 1. Features extracted from each minutia pair, connected by triangles

the matching has to be refined using the method suggested in [1] by Jain *et al.*, in which the ridges associated with the minutiae are used to align the input points with the template points. An equivalent method was presented in [8], but the matching is performed using the core and delta points[1]. We want to avoid the use of these two feature points, since they are not always present in the image. We perform the matching, using only the local structure information and no core and delta points are needed.

In our approach, the dependency among minutiae is obtained applying on the point set a Delaunay triangulation: each minutia is used as a triangle vertex. We use this approach, since the Delaunay triangulation was found to have the best structural stability under random positional perturbations [13]. The resulting triangles, obviously, must admit a certain degree of deformation, due to the elasticity of the skin. As consequence, the length of the sides and the internal angles of each triangle are not fixed. Moreover, each minutia will have always the same neighbors, even if rotation, translation and little scale changes are present. To get advantage from this assumption and admit the above triangle deformation, we proceed as follow. As first step, we extract the following quantities from each minutia pair, connected by triangles, for both the query and the template set (Fig. 1):

– the distance L between the two minutiae;
– the angle α between the orientations of the two minutiae (angular difference between $\boldsymbol{\tau_i}$ and $\boldsymbol{\tau_j}$);
– the angles β_1 and β_2 between the orientation of each minutia and the segment connecting them.

---

[1] Core and delta are points of the fingerprint flow path at which position the ridge orientation changes abruptly.

Then, each minutia pair of the query set Q is compared with all the minutia pairs of the template set T, according the following relations:

$$D_L = \frac{|L^Q - L^T|}{min(L^Q, L^T)} < Th_L \tag{4}$$

$$D_\alpha = |\alpha^Q - \alpha^T| < Th_\alpha \tag{5}$$

$$D_\beta = \left| \begin{pmatrix} \beta_1^Q \\ \beta_2^Q \end{pmatrix} - \begin{pmatrix} \beta_1^T \\ \beta_2^T \end{pmatrix} \right| < Th_\beta \tag{6}$$

where Th_L, Th_α and Th_β are fixed thresholds. In the previous relations L^Q and L^T represent two generic distances of the query and the template set, respectively, and α^Q, β_1^Q, β_2^Q, α^T, β_1^T, β_2^T are the relative angles. In (4), the denominator is a normalization factor, so that D_L is a measure of the difference of the two lengths L^Q and L^T.

Once the previous values are computed for the query and template set, an L^Q is chosen and checked against all the L^T. Only if some of the L^Q verifies the equation (4), then the equation (5) is considered and the relative angle α^Q of the considered L^Q is checked against the angle α^T of the remaining L^T. If, again, some L^T are still left, the equation (6) is considered and the relative β_1^Q and β_2^Q of L^Q are checked with the β_1^T and β_1^T of the remaining L^T. As soon as one of the previous equations fails for all the L^T, the considered L^Q is discarded and the next one is considered. As result, a list of possible candidates L^T for each L^Q is obtained. This means that for each minutiae pair (connected by an L^Q) in the query set, there can be more minutia pairs in the template set that present the same local structure. At this point, we use the triangles the segment belongs to. Each segment L belongs to one or two triangles. If, at least, one of the two triangles is similar to its counterpart, we assume that the minutia pairs are matching. Here, the similarity between triangles is measured using equation (4) for all the triangle sides: if the sides of the two triangles respect that relation, the triangles are said to be similar and stored for the next processing step. A list of triangles of Q and its relative candidates of T are obtained by this procedure. Let p be the number of these triangle pairs. Using the coordinates of the matching triangle vertexes, the transformation (1) is computed p times, obtaining a set of p triplets (t_x, t_x and ϑ). We apply these parameters to the query minutiae set through (1), obtaining p sets of points. Now, the minutia matching count is performed. For each transformed set, we count how many minutiae are within a neighborhood of a template minutia according (2). When a query minutia is found in the surrounding, it is deleted from the relative set. This is done, so that a minutia is counted only once, since, especially in regions of the fingerprint with many minutiae, it can be that more than a minutia is present in the surrounding. If this happens, we consider only the nearest minutia and delete it from the list. Let us denote with $n_1, n_2, \ldots, n_p$ the number of matching minutiae for each transformed set. We define the matching score between query and template fingerprints by $Md = 100 \times \sqrt{\frac{n^2}{M \cdot N}}$, where M and N are the number of minutiae in the query and template set, respectively, and $n = max(n_1, n_2, \ldots, n_p)$.

4   Experimental Results

The procedure, used to extract minutiae is here summarized. To reduce sensor imperfections and poor quality due to non-uniform contact with the sensor, we adaptively normalized each image as proposed in [11]. Since we used directional filters to enhance the image, we computed the local ridge orientation and frequency. We followed the idea proposed by Jain *et al.* in [3]. To enhance the ridge flow path, we filtered each image, using a modified version of the Gabor filters (MGF) [12]. Then, each image was binarized using a local mean (in a 16x16 window) as threshold value. After image thinning, we extracted location and orientation of each minutia.

The experiments have been conducted on two fingerprint collections present on the DVD of [4]. They are labeled DB1 and DB2 for the FVC2002 and each of them comprise 800 fingerprint images captured at a resolution of 500 dpi, from 100 fingers (eight impressions per finger). In the implementation, we used $Th_L = 0.1$, $Th_\alpha = \pi/12$ and $Th_\beta = \pi/9$: the larger these values, the larger the number of triangles to be checked, thus increasing the computational time. Moreover, too small values are not convenient, since fingerprints with a certain non-local deformation degree would be wrongly rejected. Thus, these parameters play the role of a *security level*: an increase of their value corresponds to an increase of the false acceptance rate, while decreasing them corresponds to an increase of the false rejection rate. For comparison, we implemented the approach proposed in [6].

Before the evaluation on the databases, we want to quantify the performance of our method independently from the automatic feature extraction algorithm. If some minutiae are missed or false minutiae are generated, the extraction step can influence the matching performance. We do that by a semi-automatic procedure. We locate by hand the minutiae presented in 200 fingerprints (five impressions per finger) of DB2. Since the minutiae were localized by an inexpert user, unable to quantify accurately the minutiae orientation, the method suggested in [3] was used to define the orientation. In our tests, we do not distinguish between ridge endings and bifurcations.

Each image n ($n = 1, \ldots, 200$) of the DB2 subset is matched against the other 199 images of the same subset. We define a correct matching, when an image matches with a template coming from the same finger, otherwise we say it is rejected. Moreover, if an image matches with a template coming from a different finger, we say that the matching is false. The verification and the rejection rate are computed as follows:

$$Verification\_rate = \frac{N_C}{N_C + N_F}, \qquad Rejection\_rate = \frac{N_R}{N_T}$$

where N_T is the number of total matches, N_C is the number of correct matches, N_F is the number of false matches and N_R is the number of rejected images. In Table 1, we report the verification and the rejection rate some experiments with the DB2 subset. Moreover, we define matching rate the following:

$$Matching\_rate = \frac{match\_num[k]}{200}$$

We recorded the best k ($k = 1, \ldots, 5$) matching scores of each image n matched with the other 199 images. If at least one image matches with three of the templates coming from the same finger (more than 50% of the templates of the same finger), we increase the $match\_num[k]$, ($k = 1, \ldots, 5$) by 1. In Table 2, the average matching rate of this experiment are reported.

The last experiment regards a performance comparison. We used the experimental protocol proposed in [10] on DB1 and DB2 sets and compare the proposed algorithm with the matcher presented in [6]. We use for both algorithms the pre-processing step described in Section 4. The results are shown in the Receiver Operating Characteristic (ROC) curves of Fig. 2 obtained by our algorithm. We can observe that our algorithm outperforms the algorithm of Tico and Kousmanen, even if they give a more complex description of each minutia (orientation-based minutia descriptor). We believe that this is due to the fact that our approach binds strongly together the minutiae, while in their approach each feature point is treated as an isolated entity. Their method describes accurately each minutia, but our method describes the relationship among minutia neighbors and this implies a more consistent description of the fingerprint even if rotations, non-linear distortions and translations are applied on the image.

Table 1. The verification and rejection rate for the manually minutia detection experiment

Verification rate	Rejection rate
99.10%	3.32%
98.12%	3.44%
100%	4.66%
99.81%	2.12%
99.00%	1.21%
99.54%	2.40%
99.76%	3.45%
99.85%	4.88%
98.32%	4.39%
99.98%	1.31%

Table 2. Top 5 matching rate of the experiment with manually minutiae detection

Number of best matches	1	2	3	4	5
Matching rate	98.88%	98.83%	98.20%	98.45%	97.98%
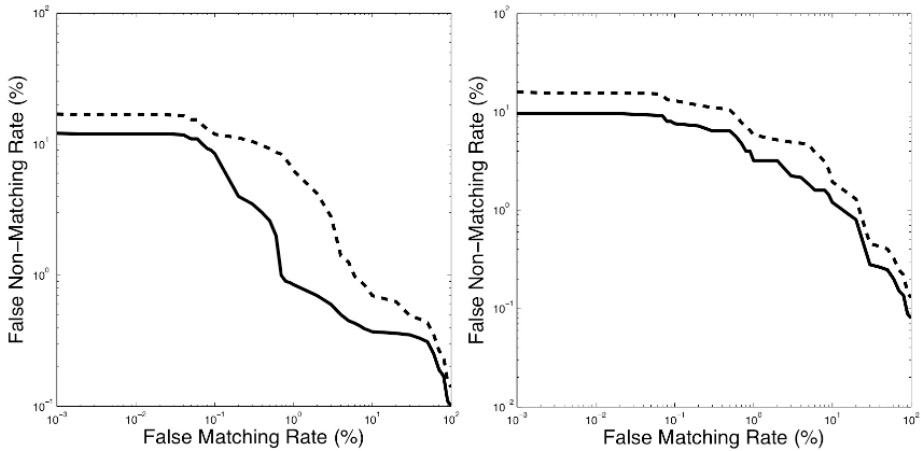
Fig. 2. ROC-curves on (a) DB1 and (b) DB2 of FVC2002 obtained with the proposed algorithm (solid line) and the algorithm proposed in [6] (dashed line)

5   Conclusions

We have introduced a new fingerprint minutiae-matching algorithm. The basic idea consists in connecting the neighbor minutiae with triangles using a Delaunay triangulation and analyzing the relative position and orientation of the grouped minutiae. Even if rotations, translations and non-linear deformations are present, the obtained triangle structure does not change significantly, except where the feature extraction algorithm fails. The proposed method was tested on a public domain database of fingerprint images and compared with another algorithm proposed in the literature. The two methods differ essentially in the minutia description. While the method proposed in [6] describes accurately each minutia, our method describes the minutia relationship that results more consistent. We gave also the performance of our algorithm using images with minutiae extracted by hand.

References

[1] A. K. Jain, L. Hong, R. Bolle: On-Line Fingerprint Verification, PAMI, Vol. 19, No. 4, pp. 302-313, 1997.   241, 243
[2] S. Pankanti, S. Prabhakar, A. K. Jain: On the Individuality of Fingerprints, PAMI, Vol. 24, No. 8, pp. 1010-1025, 2002.   241
[3] L. Hong, Y. Wan, A. K. Jain: Fingerprint Image Enhancement: Algorithms and Performance Evaluation, PAMI, Vol. 20, No. 8, pp.777-789, August 1998.   245
[4] D. Maltoni, D. Maio, A. K. Jain, S. Prabhakar: Handbook of Fingerprint Recognition, Springer Verlag, June 2003.   241, 242, 245
[5] E. Sernesi: Geometria, Bollati Boringhieri, 1989, Torino.   242

[6] M. Tico, P. Kousmanen: Fingerprint Matching Using an Orientation-Based Minutia Descriptor, PAMI, vol. 25, No. 8, pp. 1009-1014, Aug. 2003.   241, 245, 246, 247

[7] X. Jiang, W. Y. Yau: Fingerprint Minutiae Matching Based on the Local and Global Structures, ICPR00, vol. 2, Sept. 2000.   241, 242

[8] W. Zhang, Y: Wang, Core-Based Structure Algorithm of Fingerprint Verification, *ICPR02*, Vol. 1, Aug. 2002.   241, 243

[9] A. Senior, R. Bolle: Improved Fingerprint Matching by Distortion Removal, IEICE Trans. Inf. and Syst., Vol. E84-D, No. 7, July 2001.

[10] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, A. K. Jain: FVC2000: Fingerprint Verification Competition, PAMI, vol. 24, no. 3, pp. 402-412, Mar. 2002.   246

[11] B. G. Kim, H. J. Kim, D. J. Park: New Enhancement for Fingerprint Images, ICPR02:, vol. 3, pp. 879-882.   245

[12] J. Yang, L. Liu, T. Jiang, Y. Fan: A Modified Gabor Filter Design Method for Fingerprint Image Enhancement, Pattern Recognition Letters, vol. 24 (12), pp. 1805-1817, 2003.   245

[13] G. Bebis, T. Deaconu, M. Georgiopoulos: Fingerprint Identification Using Delaunay Triangulation, ICIIS99, Maryland, Nov. 1999.   241, 243

[14] N. K. Ratha, R. Bolle, V. D. Pandit, V. Vaish: Robust Fingerprint Authentication Using Local Structural Similarity, 5th. IEEE WACV, Dec. 04-06, 200, Palm Springs, California, USA.   241

[15] Y. Hao, T. Tan, Y. Wang: Fingerprint matching based on error propagation. ICIP02, Sept. 22-25, 2002, Rochester, New York, USA.   241

[16] D. Lee, K. Choi, J. Kim: A Robust Fingerprint Matching Algorithm Using Local Alignment. ICPR02, Vol. 1, Aug. 2002.   241

Estimating Fingerprint Deformation

Arun Ross[1], Sarat C. Dass[2], and Anil K. Jain[2]

[1] West Virginia University, Morgantown, WV 26506 USA
[2] Michigan State University, East Lansing, MI 48824 USA
ross@csee.wvu.edu
{sdass,jain}@msu.edu

Abstract. Fingerprint matching is affected by the nonlinear distortion introduced in fingerprint impressions during the image acquisition process. This nonlinear deformation causes fingerprint features such as minutiae points and ridge curves to be distorted in a complex manner. In this paper we develop an *average* deformation model for a fingerprint impression (baseline impression) by observing its relative distortion with respect to several other impressions of the same finger. The deformation is computed using a Thin Plate Spline (TPS) model that relies on ridge curve correspondences between image pairs. The estimated average deformation is used to distort the minutiae template of the baseline impression *prior* to matching. An index of deformation has been proposed to select the average deformation model with the least variability corresponding to a finger. Preliminary results indicate that the average deformation model can improve the matching performance of a fingerprint matcher.

1 Introduction

Automatic fingerprint matching involves determining the degree of similarity between two fingerprint impressions by comparing their ridge structure and/or the spatial distribution of the minutiae points. When direct-contact fingerprint sensors are used, the image acquisition process introduces non-linear distortions in the ridge structure due to the non-uniform finger pressure applied by the subject on the sensor and the elastic nature of the human skin. For reliable matching, these non-linear deformations must be accounted for prior to comparing two fingerprint images. Deformation models based on affine transformations invariably lead to unsatisfactory matching results since the distortions are basically elastic in nature. Thus, alternate techniques to handle such distortions have been suggested in the literature (see, for example, [1, 2, 3, 4, 5, 6]). However, almost all techniques proposed thus far deal with the problem of non-linear distortion on a case by case basis, i.e., for *every* pair of fingerprint impressions (or for *every* fingerprint impression), a distortion removal technique is applied. No attempt has been made to develop a *finger-specific deformation model* that can be computed offline and then used during matching. The main advantage of an offline technique is that once a finger-specific model has been computed, recomputation of the model is not necessary during the matching stage. In this paper we describe

(a)

(b)
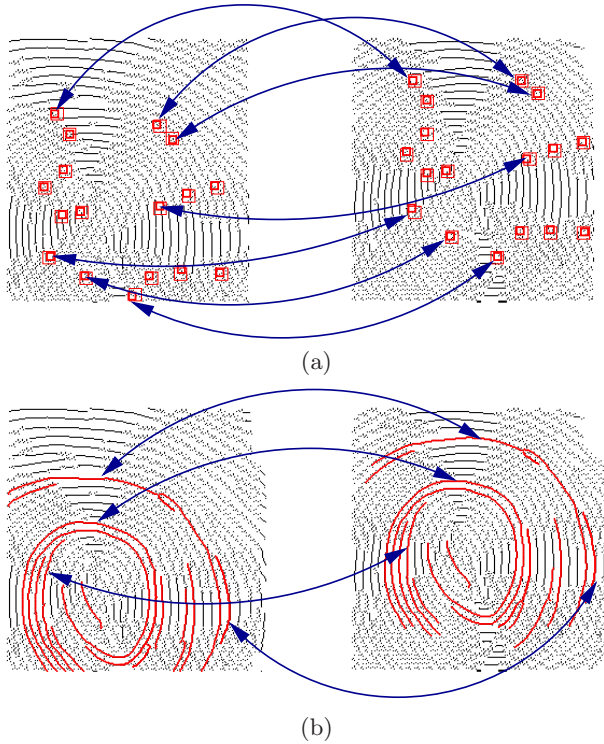
Fig. 1.   (a) Minutiae point correspondences and (b) ridge curve correspondences between two impressions of a finger

an *average* deformation model for a fingerprint impression (baseline impression) by observing its relative distortion with respect to several other impressions of the same finger. The distortion is estimated using ridge curve correspondence between pairs of fingerprint impressions. The estimated average deformation is then used to distort the template minutiae set prior to matching it with that of a previously unseen query fingerprint. We also propose an index of deformation for ranking the average deformation models corresponding to every impression of a finger.

2   The Fingerprint Warping Model

Given a pair of grayscale fingerprint images, I_0 and I_1, we first obtain their thinned versions, R_0 and R_1. A thinned image is a binary image with grayscale values of 0 (indicating valleys) and 255 (indicating ridges). Each thinned image can be thought of as a collection of ridge curves. Minutiae points are then extracted from R_0 and R_1 resulting in two minutiae sets $M_0 = (m_{0,1}, m_{0,2}, \ldots, m_{0,k_0})$ and $M_1 = (m_{1,1}, m_{1,2}, \ldots, m_{1,k_1})$ of cardinalities k_0

and k_1, respectively. Here, each minutiae point $m_{i,j}$ is characterized by its location in the image, the orientation of the associated ridge, and the grayscale intensity of pixels in its vicinity. Minutiae correspondences between M_0 and M_1 are obtained using the elastic string matcher described in [7]. The output of this matcher is a similarity score in the range [0,1000] and a set of correspondences of the form $C = \{(m_{0,a_j}, m_{1,b_j}) : j = 1, 2, \ldots, k\}$ where $k \leq \min\{k_0, k_1\}$, and the a_js (b_js) are all distinct. Figure 1(a) shows an example of the minutiae point correspondences between two impressions of a finger. Once the correspondence between M_0 and M_1 has been established, the ridge curves associated with these minutiae points are extracted from R_0 and R_1 using a simple ridge tracing technique. A minutiae point that is a ridge ending has one ridge curve associated with it while a ridge bifurcation has three associated ridge curves (Figure 1(b)).[1]

Having determined the corresponding ridge curves, we next establish a correspondence between *points* on these curves by sampling every q-th point ($q = 20$) on each of the ridge curves. We denote this set of corresponding ridge points by $\mathcal{U} = (u_1^*, u_2^*, \ldots, u_N^*)^T$ and $\mathcal{V} = (v_1^*, v_2^*, \ldots, v_N^*)^T$. We use the thin plate spline (TPS) model to estimate the non-linear deformation F based on these points. TPS represents a natural parametric generalization from rigid to mild non-rigid deformations. The deformation model for TPS is given in terms of the warping function $F(u)$, with $F(u) = c + A \cdot u + W^T s(u)$, where $u \in R^2$, c is a 2×1 translation vector, A is a 2×2 affine matrix, W^T is a $N \times 2$ coefficient matrix, and $s(u) = [\sigma(u - u_1^*), \sigma(u - u_2^*), \ldots, \sigma(u - u_N^*)]^T$. Here, $\sigma(u) = ||u||^2 \log(||u||)$ if $||u|| > 0$ and $\sigma(u) = 0$, otherwise. There are 6 and $2N$ parameters corresponding to the affine[2] and non-linear parts of the deformation model, respectively, resulting in a total of $2N + 6$ parameters to be estimated. The restriction $F(u_j^*) = v_j^*$, $j = 1, 2, \ldots, N$ provides $2N$ constraints. For the parameters to be uniquely estimated, we further assume that W satisfies the two conditions (i) $1_N^T W = 0$ and (ii) $U_s^T W = 0$, where 1_N is the vector of ones of length N. Thus, the parameters of the TPS model can be obtained from the matrix equation

$$\begin{bmatrix} H & 1_N & \mathcal{U} \\ 1_N^T & 0 & 0 \\ \mathcal{U}^T & 0 & 0 \end{bmatrix} \begin{bmatrix} W \\ c^T \\ A^T \end{bmatrix} = \begin{bmatrix} \mathcal{V} \\ 0 \\ 0 \end{bmatrix}, \tag{1}$$

where H is the $N \times N$ matrix with entries $h_{ij} = \sigma(u_i^* - u_j^*)$. This gives rise to a TPS model that minimizes the bending energy subject to the perfect alignment constraint (i.e., $F(u_j^*) = v_j^*$). A more robust TPS model can be obtained by relaxing this constraint, and instead determining the function F which minimizes the expression

$$\sum_{j=1}^{N} (v_j^* - F(u_j^*))^T (v_j^* - F(u_j^*)) + \lambda J(F), \tag{2}$$

---

[1] Ridge endings and ridge bifurcations may be interchanged in the thinned image. We do account for such anomalies when determining ridge curve correspondences.

[2] The affine parameters are determined using minutiae point correspondences only and not the ridge point correspondences.

(a) $\Phi = 15.54$ (b) $\Phi = 17.97$

(c) $\Phi = 48.79$ (d) $\Phi = 94.34$
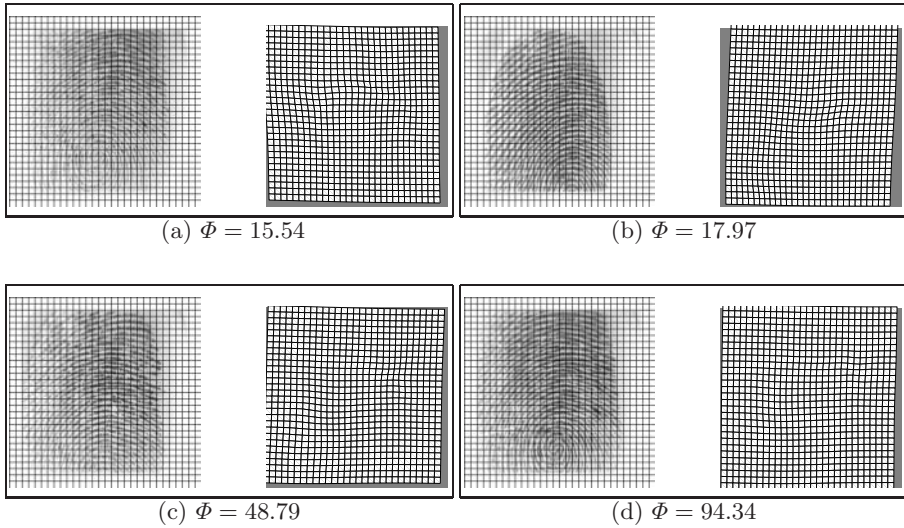
Fig. 2. The average deformation model (shown as deformations on a reference grid) corresponding to 4 templates of a finger sorted in increasing Φ-values. (a) is chosen to be the optimal template since it has the least Φ-value

where $J(F) = \sum_{j=1}^{2} \int_{(x,y)} \left\{ \left(\frac{\partial^2 F_j(x,y)}{\partial x^2} \right)^2 + 2 \left(\frac{\partial^2 F_j(x,y)}{\partial x \partial y} \right)^2 + \left(\frac{\partial^2 F_j(x,y)}{\partial y^2} \right)^2 \right\} dx \, dy$
represents the bending energy associated with $F = (F_1, F_2)^T$, F_j is the jth component of F, and $\lambda > 0$. The case $\lambda = 0$ gives rise to the TPS model described by equation (1). For general $\lambda > 0$, the parameters of the resulting TPS model can be obtained using equation (1) with H replaced by $H + \lambda I_N$, where I_N is the $N \times N$ identity matrix.

3   Average Deformation Model

Suppose we have L impressions of a finger, $T_1, T_2, \ldots, T_L$. Each impression, T_i, can be paired with the remaining impressions to create $L - 1$ pairs of the form (T_i, T_j), $j \neq i$. For the pair (T_i, T_j), we obtain a non-linear transformation F_{ij} by employing the technique described in section 2. Note that F_{ij} transforms *every* pixel in the template fingerprint, T_i, to a new location. Thus, we can compute the *average* deformation of each pixel u in T_i as, $\bar{F}_i(u) = \frac{1}{L-1} \sum_{j \neq i} F_{ij}(u)$. There will be L average deformation models corresponding to the L impressions of the finger. The average deformation is the typical deformation that arises when we compare one fingerprint impression of a finger (the baseline impression) with other impressions of the same finger. Figure 2 shows that changing the baseline impression for the finger will result in a different average deformation model for that finger (the Φ values are discussed in section 3.1). Figure 3 shows the average deformation for 3 different fingers; it can be clearly seen that the average

warping functions are different for the 3 fingers indicating that the fingerprint deformation is finger-specific.

3.1   The Φ Index of Deformation

We suggest a method to rank the average deformation models pertaining to multiple impressions of a finger. In order to do this, we first define the pixel-wise covariance matrix associated with the i-th average deformation, $\bar{F}_i$, as follows: $D_{\bar{F}_i}(u) = \frac{1}{L-1} \sum_{j \neq i} (F_{ij}(u) - \bar{F}_i(u)) \cdot (F_{ij}(u) - \bar{F}_i(u))^T$. Here, F_{ij} is the deformation function that warps T_i to T_j. The covariance matrix defined at each pixel u, is a measure of the variability associated with the estimated deformation functions. Two choices of pixel-wise measures of variability are given by (i) the determinant, $\phi(D_{\bar{F}_i}(u)) = |D_{\bar{F}_i}(u)|$, and (ii) the trace, $\phi(D_{\bar{F}_i}(u)) = \text{tr}(D_{\bar{F}_i}(u))$. Pixels with large (small) values of ϕ indicate high (low) variability in the deformations F_{ij}. We propose to use the values of ϕ to determine the optimal model for a given finger. We define the ith index of deformation, Φ_i, as $\Phi_i = \frac{1}{|S|} \sum_{u=1}^{|S|} \phi(D_{\bar{F}_i(u)})$, where, $\phi(D) = tr(D)$, and $|S|$ is the number of pixels in the image. Subsequently, we choose T_{i*} as the template with the smallest variability in deformation if $i^* = \arg \min_i \Phi_i$. In effect, we choose that template T_i that minimizes the average variation across pixels measured in terms of Φ_i. Low (high) values of the index of deformation indicate that the warping functions are similar (dissimilar) to each other.

4   Experimental Results

In order to reliably estimate fingerprint deformation, we need several impressions of the same finger (~ 16). Large number of impressions per finger are not available in standard fingerprint databases (e.g., FVC 2002 and NIST). Therefore, fingerprint images of 50 fingers were acquired using the Identix sensor (256×255, 380 dpi) over a period of two weeks in our lab. There were 32 impressions corresponding to every finger, resulting in a total of $1,600$ impressions. One half of the impressions ($L = 16$ for each finger, resulting in 800 impressions) were used as templates to compute the average deformation model for each finger, while the remaining 800 impressions were used as query images for testing. For each template image, T, the minutiae set, M_T, and the thinned image, R_T, were extracted. The average deformation model of T, $\bar{F}_T$, was obtained based on pairings with the remaining 15 impressions of the same finger (equation (2) with $\lambda = 0.1$). The minutiae set M_T was transformed to the deformed set, $MD_T \equiv \bar{F}_T(M_T)$ using $\bar{F}_T$. A total of 800 sets (50×16) of deformed minutiae points were thus obtained. In order to test the matching performance of the deformed minutiae sets, and the utility of the index of deformation, Φ, the following two experiments were conducted.

   In the first experiment, the matching performance using the average deformation model was evaluated. Every template image, T, was compared with every query image, Q, and two types of matching scores [7] were generated for each

(a) $\Phi = 15.54$ (b) $\Phi = 37.59$

(c) $\Phi = 85.18$ (d) $\Phi = 35.30$
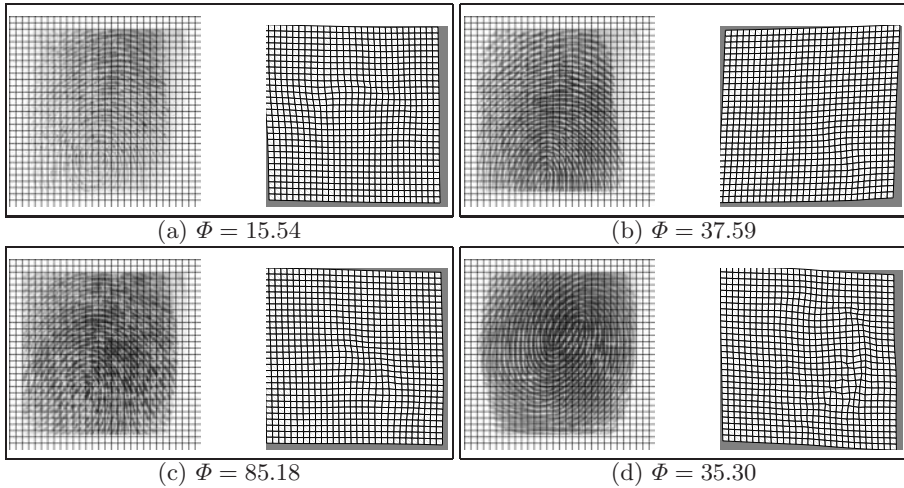
Fig. 3. The average deformation model (shown as deformations on a reference grid) of 4 different fingers

comparison: the matching score obtained by matching (i) M_T with M_Q, and (ii) MD_T with M_Q. The Receiver Operating Characteristic (ROC) curve plotting the genuine accept rate (GAR) against the false accept rate (FAR) at various matching thresholds is presented in Figure 4(a). An overall improvement of 2% is observed when the average deformation model is used to distort M_T prior to matching. In the second experiment, the advantage of using the index of deformation is demonstrated. The Φ-index of deformation (with $\phi(D) = \mathrm{tr}(D)$) is used to rank the templates according to variability in the distortion. The template images can now be split into two sets: (i) impressions with the least Φ values for every finger (the Φ-optimal templates) and (ii) the remaining impressions for every finger (the Φ-suboptimal templates). We repeated the matching procedure outlined above using these two template sets. The resulting ROC curve is shown in Figure 4(b). From the figure, it is clear that using Φ-optimal templates results in better performance compared to using Φ-suboptimal templates. Further, the Φ-suboptimal templates still yield better performance compared to the non-distorted templates thus demonstrating the importance of the average deformable model.

5 Summary and Future Work

In this paper, an average deformation model for fingerprint impressions has been proposed. The proposed technique has been shown to improve the performance of a fingerprint matching system. An index of deformation has been suggested to select the "optimal" average deformation model corresponding to multiple impressions of a finger. Future work includes adopting an incremental approach
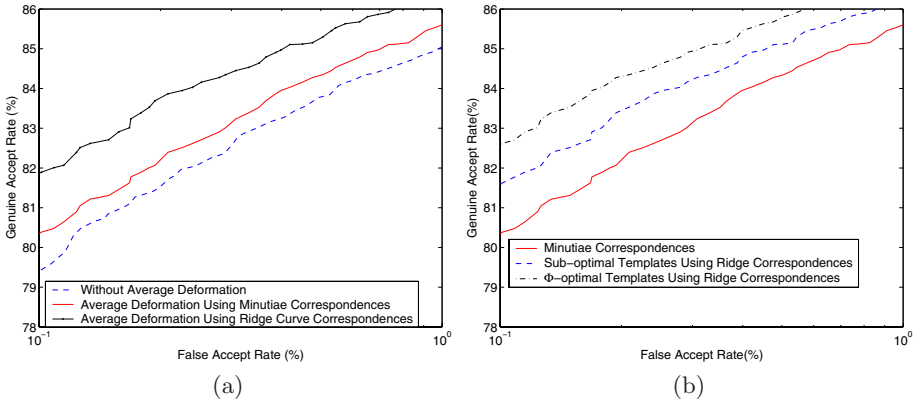
Fig. 4. (a) Improvement in matching performance when ridge curve correspondences is used to develop the average deformation model. (b) Matching performance when the Φ index of deformation is used to select optimal templates

to updating the average deformation model, determining a more robust measure (than simple pixel-wise averaging) to compute the average deformation model, and employing curve matching techniques to establish ridge curve correspondences between image pairs.

References

[1] Ratha, N. K., Bolle, R. M.: Effect of controlled acquisition on fingerprint matching. In: Proceedings of the International Conference on Pattern Recognition. Volume 2., Brisbane, Australia (1998) 1659–1661   249

[2] Dorai, C., Ratha, N., Bolle, R.: Detecting dynamic behavior in compressed fingerprint videos: Distortion. In: Proceedings of Computer Vision and Pattern Recognition. (2000) 320–326   249

[3] Kovács-Vajna, Z. M.: A fingerprint verification system based on triangular matching and dynamic time warping. IEEE Transactions on PAMI **22** (2000) 1266–1276   249

[4] Bazen, A. M., Gerez, S.: Fingerprint matching by thin-plate spline modelling of elastic deformations. Pattern Recognition **36** (2003)   249

[5] Senior, A., Bolle, R.: Improved fingerprint matching by distortion removal. IEICE Transactions on Information and Systems **E84-D** (2001) 825–831   249

[6] Watson, C., Grother, P., Cassasent, D.: Distortion-tolerant filter for elastic-distorted fingerprint matching. In: Proceedings of SPIE Optical Pattern Recognition. (2000) 166–174   249

[7] Jain, A. K., Hong, L., Bolle, R.: On-line fingerprint verification. IEEE Transactions on PAMI **19** (1997) 302–314   251, 253

Comparison of Classification Methods for Time-Series Detection of Perspiration as a Liveness Test in Fingerprint Devices

Stephanie A. C. Schuckers[1], Sujan T. V. Parthasaradhi[2], Reza Derakshani[2], and Lawrence A. Hornak[2]

[1] Department of Electrical and Computer Engineering
Clarkson University, PO Box 5720, Potsdam, NY, 13699 USA
sschucke@clarkson.edu
[2] Center for Identification Technology Research (CITeR)
Lane Department of Computer Science and Electrical Engineering
West Virginia University, PO Box 6109, Morgantown, WV 26506-6109, USA
{reza,sujan,lah}@csee.wvu.edu

Abstract. Fingerprint scanners may be susceptible to spoofing using artificial materials, or in the worst case, dismembered fingers. An anti-spoofing method based on liveness detection has been developed for use in fingerprint scanners. This method quantifies a specific temporal perspiration pattern present in fingerprints acquired from live claimants. The enhanced perspiration detection algorithm presented here improves our previous work by including other fingerprint scanner technologies, using a larger, more diverse data set, and a shorter time window. Several classification methods were tested on fingerprint images from 33 live subjects, 33 spoofs created with dental material and Play-Doh, and 14 cadaver fingers. Each method had a different performance with respect to each scanner and time window. However, all the classifiers achieved approximately 90% classification rate for all scanners, using the reduced time window and the more comprehensive training and test sets.

1   Introduction and Background

Biometric systems are subject to various threats like attacks at the sensor level, replay attacks on the data communication stream and attacks on the database [1]. This paper will focus on countermeasures to attacks at the sensor level of fingerprint biometric systems or spoofing, the process of defeating a biometric system through an introduction of a fake biometric sample or, worst case, a dismembered finger. Liveness detection, i.e. to determine whether the introduced biometric is coming from a live source, has been suggested as a means to circumvent attacks that use spoof fingers.

Previous work has shown that it is possible to spoof a variety of fingerprint technologies through relatively simple techniques. These include utilization of latent fingerprints, and molds created from casts of live fingers or from latent fingerprints lifted from a surface and reproduced with photographing etching techniques [2] - [7].

Casts have been made from wax, silicon and plastic, and molds from silicon or gelatin (gummy finger) [4], [5].

Our laboratory has demonstrated vulnerability to spoofing using dental materials for casts and Play-Doh for molds [6], [7]. Furthermore, we have tested fingerprint scanners with cadaver fingers. Ten attempts were performed for all available security levels for optical, capacitive AC, capacitive DC, and electro-optical technologies [6]. Results showed that the spoofing rate for cadaver fingers was typically 90% when verified against an enrolled cadaver finger, whereas for Play-Doh and water-based clay, results varied from 45-90% and 10-90%, respectively, when verified against the original enrolled live finger.

In order to avoid spoof attacks of fingerprint biometric systems, various liveness countermeasures have been considered including sensing of finger temperature [8], laser detection of the 3-D finger surface and pulse [9], pulse oximetry [8], [10], ECG [8], and impedance and electrical conductivity of the skin (dielectric response) [11]. Summaries of liveness and anti-spoofing methods are given in [6], [12], [13]. Most methods require additional hardware which is costly and, unless integrated properly, may be spoofed with an unauthorized live person. In addition, most previously developed methods are not available commercially and/or have not been tested rigorously in order to determine their effectiveness.

Previously, we have developed an anti-spoofing method which is based on a time-series of fingerprint images captured from a DC capacitance-based CMOS fingerprint scanner [7]. The method uses the physiological process of perspiration to determine liveness of a fingerprint. The initial algorithm extracted the gray levels along the ridges to form signals, calculated a set of features, and used a neural network to perform classification. While these initial results were encouraging, they raised a number of issues, including the performance of the techniques across a more diverse population, the contraction of the time series data to achieve user transparency of the technique, and the applicability of the approach to other fingerprint sensor technologies.

2   Methods

2.1 Data Collection

Three types of fingerprint scanner technologies were used in this study: capacitive DC (Precise Biometrics, 100SC), electro-optical (Ethentica, Ethenticator USB 2500), and optical (Secugen, EyeD Hamster HFDUO1A). For each device, fingerprint images were collected from live, spoof, and cadaver fingers. Protocols for data collection from the subjects were followed that were approved by the West Virginia University Institutional Review Board (IRB) (HS#14517 and HS#15322). Thirty-three volunteers were solicited and represented a wide range of ages (20-60 years), ethnicities, and both sexes (17 men and 16 women). The criteria for data collection protocol were enrollment in 1 of 5 attempts and the verification in at least 1 of 6 attempts with the enrolled finger. The images utilized in this paper are the first image and images from approximately two seconds and five seconds. To generate spoof fingerprint images, finger casts were created using dental impression materials from

thirty subjects who participated in generation of the time series of live fingerprint images. Fourteen cadaver fingers (from 4 subjects, of male age 41, female ages 55, 65, and 66) were collected in collaboration with the Musculoskeletal Research Center (MSRC) at the West Virginia University Health Science Center. Only fingerprint images from cadaver fingers which were able to enroll were considered for study. Examples of the time-series of live, spoof, and cadaver fingerprint images are shown in Fig. 1.

2.2 Perspiration Detection Algorithm

The basis for our original method and details of the algorithm are discussed in detail in [7]. In brief, when in contact with the fingerprint sensor surface, live fingers, as opposed to cadaver or spoof, demonstrate a distinctive spatial moisture pattern which evolves in time due to the physiological perspiration process. Optical, electro-optical, and solid-state fingerprint sensors are sensitive to the skin's moisture changes on the contacting ridges of the fingertip skin. To quantify the perspiration phenomenon, our algorithm maps a 2-dimensional fingerprint image to a "signal" which represents the gray level values along the ridges. Variations in gray levels in the signal correspond to variations in moisture both statically (on one image) and dynamically (difference between consecutive images). The static feature measures periodic variability in gray level along the ridges due to the presence of perspiration around the pores. The dynamic features quantify the temporal change of the ridge signal due to propagation of this moisture between pores in the initial image relative to image captures two (or five) seconds later. The cadaver and spoof fingers fail to provide the mentioned static and dynamic patterns due to the lack of active pore-emanated perspiration

The basic steps performed in the algorithm are described as follows. First, two fingerprint images are captured within a 2 (or 5) second interval (referred to as first and last capture). The results are enhanced by having the subjects wipe their fingers immediately before capture. Captured fingerprints are preprocessed to remove noise and the hollow pores seen in high-resolution. The last capture is used to locate the ridges. Ridges that are not long enough to cover at least 2 pores are discarded. Using the thinned ridge locations as a mask, the gray levels of the original image underneath these ridge paths are recorded. The resulting signals for the first and the last capture are representative of the moisture level along the ridges for a given pair of images in the time series.

Prior work [7] derived measures from these ridge signals and used a feed forward multi-layer perceptron for classification. In the new study, in order to increase the robustness of the classification, two additional measures were introduced and are presented here. In the case that the fingerprint signal swings beyond a device's dynamic range (i.e. the device enters cut-off or saturation due to extreme dryness/moisture), the information about the minimums and maximums and their rate of change, utilized in the second dynamic measure, will be lost. These two measures address this by taking advantage of the upper and lower cut off region lengths of the fingerprint signals and converting them into perspiration rates. The equations for the new dynamic measures are given below.

Fig. 1. Example fingerprint images from live (top), and spoof and cadaver fingers (bottom). Top row shows captures at 0, 2 and 5 seconds (left to right) with magnified insets demonstrate progression of perspiration pattern in time. Bottom-left pair shows the 0 and 2 second captures of a spoof finger, and the bottom right pair shows the same for a cadaver finger

Dry Saturation Percentage Change. This fifth dynamic measure (DM5) indicates how fast the low cut-off region of the ridge signal is disappearing, thus extracting further perspiration rate information from the low-cutoff region:

$$DM5 = \frac{\sum_{i=1}^{m} \delta(C_{1i} - LT) - \delta(C_{2i} - LT)}{0.1 + \sum_{i=1}^{m} \delta(C_{2i} - LT)} \tag{1}$$

C_{1i} is referring to the ith point (pixel gray level) in the first capture ridge signal. C_{2i} is the same except for that it is for the second capture. i is equal to 1 to the length of the ridge signal (m). m is the same for C_1 and C_2 since the same mask was used for C_1 and C_2. LT is the low-cutoff threshold of the ridge signal. δ is the discrete delta function. 0.1 is added to the denominator to avoid division by zero.

Wet Saturation Percentage Change. The sixth dynamic measure (DM6) is the dual of DM5 and indicates how fast the high cut-off region of the ridge signal is appearing:

$$DM6 = \frac{\sum_{i=1}^{m} \delta(C_{2i} - HT) - \delta(C_{1i} - HT)}{0.1 + \sum_{i=1}^{m} \delta(C_{1i} - HT)} \qquad (2)$$

HT is the high-cutoff (saturation) threshold of the ridge signal.

One static and six dynamic measures are used as the augmented feature set for classification of fingerprints. Classification was performed separately for each time window, dividing images into live and spoof where spoof fingerprint images include images from Play-Doh spoofs and cadavers. With approximately 75 images for each scanner, 50% of the data was used as a training set and the remaining 50% as the test set for classification. Three classification methods were used: neural networks, discriminant analysis, and One R. For One R classifier, bucket size of 6 was used, which chose the static measure to form a rule for all scanners.

3   Results

Fig. 2 presents the classification rate for live and spoof fingerprints for each device and time window. The capacitive DC device demonstrates between 80-93.3% classification for live fingers and 80-95% for spoof fingers, depending on the method and time window. There is little difference in the results for two seconds as compared to five seconds. For the electro-optical device, 62.5-93.3% classification is achieved for live and 81-100% for spoof. There is a modest improvement in live classification from two to five seconds (62.5-81.3% to 81.3-93.3%), with a smaller increase in spoof classification (81-94.7% to 95.2-100%). For optical, classification ranged from 73.3-100% for live and 85.7-95.4% for spoof with a small change for live classification from two to five seconds (73.3-93.8% to 80-100%). In addition, exploratory statistical analysis performed on the average values shows that the means for DM2 and DM5 for capacitive DC, SM, DM2, and DM6 for electro-optical, and SM, DM2 and DM5 for optical, are statistically different.

4   Discussion and Future Work

Detection of liveness has been suggested as a means to make spoofing more difficult. The difficulty with proposed measurements so far [6], [8]-[12] is that they usually require extra hardware capture liveness features which makes them expensive and bulky. Furthermore, proposed liveness methods have not been rigorously tested and evaluated with relation to impact on statistical measurements like false reject and false accept ratios, user acceptance, universality, and collectability.
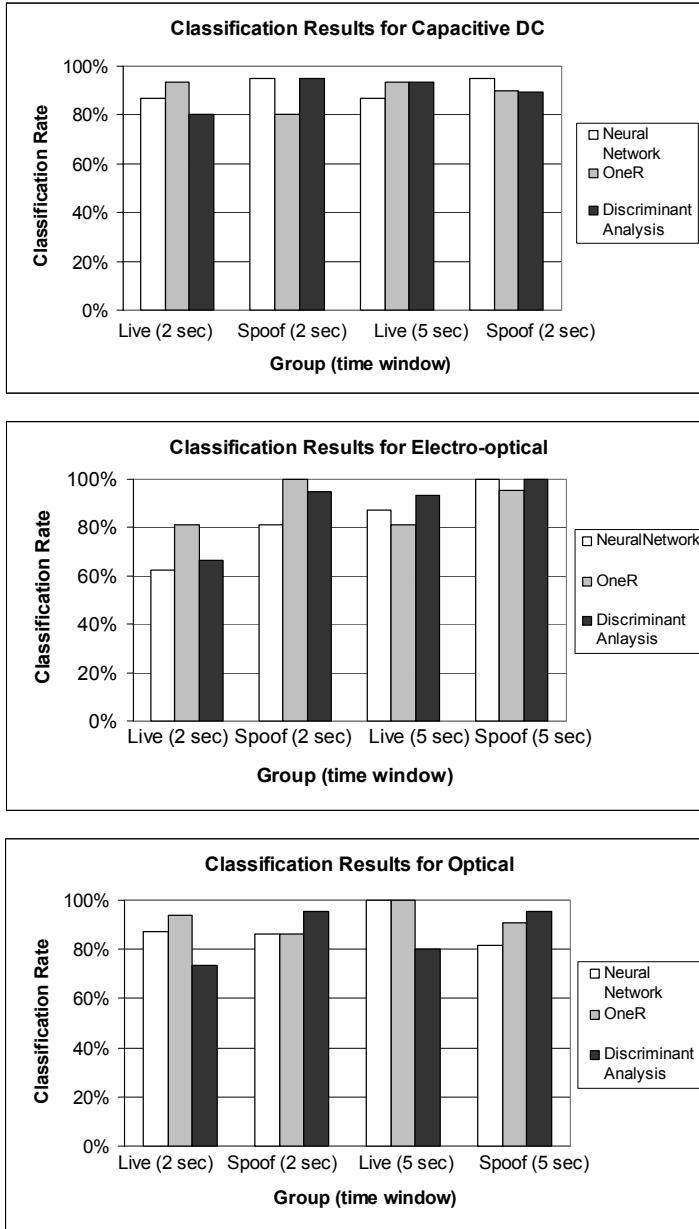
Fig. 2. Classification rates of different scanner technologies for live and spoof (2 and 5 second windows) using neural network, One R, and discriminant analysis classification techniques

The research presented here suggests a liveness-detection method which is software-based and more difficult to spoof, since the spoof would have to replicate perspiration emanating from the pores and spreading across the ridges. This study

expands our former research [7] to consider (1) a variety of technologies, (2) a larger, more diverse dataset, and (3) a shorter time window. The dataset presents a diverse set of fingers and therefore begins to consider potential problems (dry finger, moisture-saturated finger, ridge variations, etc.). Even with this diversity, we were able to achieve approximately 90% classification using common classifiers, with a shorter time window of two seconds attaining similar classification results.

Training was separate for each device and time window. A device-independent algorithm was not developed due to the large differences in the measurements across devices and since the statistical analysis showed different features having relevance for difference devices. Between the statistical analysis and classification results, a device-specific approach would most likely be the most successful for classification.

Potentially, subjects having dry and overly moist fingers may receive a false rejection. But this is a problem in general i.e. independent of liveness issue. Therefore for future work, environmental testing will be necessary to demonstrate applicability to a wide variety of settings within the range of the typical function of the device. Furthermore, features are averaged across the entire fingerprint image, while targeting areas of the image that are changing due to perspiration may improve the separation of live and spoof measurement.

References

[1]   Ratha N. K.: Enhancing security and privacy in biometrics-based authentication systems, IBM Systems Journal, vol.40, (2001) 614-634.

[2]   Thalheim L., and Krissler J.: Body check: biometric access protection devices and their programs put to the test, c't magazine (2002).

[3]   Willis D., and Lee M.: Biometrics under our thumb, Network Computing (1998).

[4]   van der Putte T., and Keuning J.: Biometrical fingerprint recognition: don't get your fingers burned, In: Proceedings of the Fourth Working Conference on Smart Card Research and Advanced Applications. Kluwer Academic Publishers (2000) 289-303.

[5]   Matsumoto T., Matsumoto H., Yamada K., and Hoshino S.: Impact of artificial 'gummy' fingers on fingerprint systems, Proceedings of SPIE, vol. 4677, (2002).

[6]   Schuckers S. A. C.: Spoofing and anti-spoofing measures, Information Security Technical Report, Vol. 7, (2002) 56–62.

[7]   Derakhshani R., Schuckers S. A. C., Hornak L., and O'Gorman L.: Determination of vitality from a non-invasive biomedical measurement for use in fingerprint scanners. Pattern Recognition Journal, Vol. 36, (2003) 383-396.

[8]   Osten D., Carim H. M., Arneson M. R., Blan B. L.: Biometric, personal authentication system", Minnesota Mining and Manufacturing Company, U.S. Patent #5,719,950, (1998).

[9]   Seifried K.: Biometrics - What You Need to Know," Security Portal (2001).

[10]   Lapsley P. D. , Less J. A., Pare D. F., Jr., Hoffman N.: Anti-fraud biometric sensor that accurately detects blood flow, SmartTouch, LLC, U.S. Patent #5,737,439, (1998).

[11]   Kallo P., Kiss I., Podmaniczky A., and Talosi J.: Detector for recognizing the living character of a finger in a fingerprint recognizing apparatus, Dermo Corporation, Ltd. U.S. Patent #6,175,64, (2001).

[12]   Valencia V. and Horn C.: Biometric Liveness Testing. In: Woodward, J.D., Orlans N.M., Higgins R. T., (eds): Biometrics. Osborne McGraw Hill, New York (2003).

[13]   International Biometric Group: Liveness Detection in Biometric Systems, white paper, Available at http://www.ibgweb.com/reports/public/reports/liveness.html.
[14]   DENTSPLY International, York, PA http://ww.dentsply.com.
[15]   Kerr Dentistry, Orange, CA, http://www.kerrdental.com.

On Optimal Estimate of Directional Maps
for Fingerprint Images

Wei Gong

Cogent Systems, Inc.
209 Fair Oaks Avenue, South Pasadena, California 91030, USA
wgong@cogentsystems.com

Abstract. A criterion function for optimally estimating directional map of a fingerprint image is proposed and analyzed. It is shown that the algorithm proposed by Kass and Witkin [1] is an optimal solution of a special case in the framework. By introducing a concept of dynamic weighting, several other estimators based on the proposed framework are investigated. Experiment results show that these new estimators provide better accuracy.

1 Introduction

An important characteristic of fingerprint is its ridges run parallel to one another in local areas so that a direction is naturally associated with each point and can be used to effectively describe the fingerprint [2,4]. In processing fingerprint images, therefore, the very first stage is usually to estimate at each pixel the direction of the ridge structure. This results in a directional map of the fingerprint, which has been playing a critical role in fingerprint image analysis [2,3].

A popular algorithm to estimate direction of a fingerprint image is based on gradient map of the fingerprint. In order to get a reliable direction estimation an idea is to "average" gradients over a window surrounding a pixel [1,5]. For gradient, however, a simple summation cannot be used because of the periodicity of direction [1,2,5]. Kass and Witkin [1] resolved this problem by doubling the angles of gradient vectors and proposed a highly efficient algorithm. Their algorithm has been used and studied by many researchers [2,5].

This paper will propose a more general optimization framework for direction estimation .It will be shown that Kass and Witkin's algorithm is a special case of the framework. By introducing a concept of dynamic weighting, several new estimators based on the framework are presented. Experiments are given to compare these algorithms to Kass and Witkin's one, and to demonstrate their better accuracy.

2 Criterion for Optimal Estimation

Consider a fingerprint image $I(x,y)$, the gradient $(\partial I(x,y)/\partial x, \partial I(x,y)/\partial y)$ at (x,y) can be expressed as $r(x,y)e^{i\alpha(x,y)}$ where r,α indicate the magnitude and angle of the gradient vector, respectively. Since the ridge direction is always perpendicular to the

gradient, The problem of direction estimation can be stated as: *given a set of gradient vectors* $\Gamma = \left\{ r_k e^{\alpha_k} \mid 1 \le k \le M \right\}$, *find out an angle* φ *(or equivalently, a direction) such that* φ *is the closest to set of angles* $\Theta = \{\theta_k = \alpha_k + \pi/2 \mid 1 \le k \le M\}$. We call Θ *the primitive direction set* of the estimation, which is derived from the gradient vectors by turning their angles by $\pi/2$. To mathematically express the problem, a criterion has to be established to quantify how close a single direction is to Θ. Noticing a direction φ (or θ_k) and its opposite $\varphi + \pi$ (or $\theta_k + \pi$) represent the same ridge structure, the criterion has to treat the same a direction and its opposite. Let

$$C(\theta_k, \varphi) = |\cos(\theta_k - \varphi)| \tag{1}$$

$C(\theta, \varphi)$ is actually projection of θ onto φ. It is a cyclic function with period π with respect to both θ and φ, so above requirements are met. Furthermore, it reaches its maximum (1) when $\theta_k = \varphi$ and decreases monotonously as θ_k turns away from φ and finally get to its minimum (0) when θ_k is perpendicular to φ. Thus $C(\theta_k, \varphi)$ can be used effectively to quantify how close φ is to θ_k. The larger $C(\theta_k, \varphi)$ is, the closer φ is to θ_k. By summing this single direction projection over the direction set Θ, total projections of the given set of gradient vectors onto directions φ is defined as

$$F(\varphi) = \sum_{k=1}^{M} R(r_k) f(C(\theta_k, \varphi)) = \sum_{k=1}^{M} R(r_k) f(|\cos(\theta_k - \varphi)|) \tag{2}$$

Here $R(r) \ge 0$ and $0 \le f(x) \le 1$ are called modulation functions and f is an increasing function. From above discussion, therefore, the optimal direction estimations can be defined as the directions maximizing the total projection defined by Eq. (2). That is

$$F(\varphi_{optimal}) = \max_{0 \le \varphi < \pi} F(\varphi) \tag{3}$$

Based on an optimal estimate, consider following quantity

$$G(\varphi_{optimal}) = F(\varphi_{optimal}^{\perp}) \big/ F(\varphi_{optimal}) \tag{4}$$

where $\alpha^{\perp} = \alpha + \pi/2$ denotes a direction orthogonal to α. $G(\varphi_{optimal})$ is the ratio of the projection of Θ onto a direction orthogonal to $\varphi_{optimal}$ over the projection of Θ onto $\varphi_{optimal}$ itself. So it quantifies how well the energy of Θ concentrates onto $\varphi_{optimal}$. In other words, $G(\varphi_{optimal})$ tells how consistent the primitive direction set Θ is with $\varphi_{optimal}$. The larger $G(\varphi_{optimal})$ means the less consistency in Θ. We call $G(\varphi_{optimal})$ inconsistency of the estimate. From $F(\varphi_{optimal}) = \max\limits_{0 \le \varphi < \pi} F(\varphi) \ge F(\varphi_{optimal}^{\perp})$, we have

$$0 \le G(\varphi_{optimal}) \le 1 \tag{5}$$

3   Optimal Estimate with Quadratic Modulation

In this section, we look into a case in which f is in following quadratic form.

$$f(x) = x^2 \qquad (6)$$

Then, let T denotes transpose of a matrix , it can be easily shown that

$$F(\varphi) = xAx^T \qquad \text{with } x = (\cos\varphi, \sin\varphi), \; A = \begin{pmatrix} H_{cc} & H_{cs} \\ H_{cs} & H_{ss} \end{pmatrix} \qquad (7)$$

$$H_{cc} = \sum_{k=1}^{M} R(r_k)\cos^2\theta_k \;,\; H_{ss} = \sum_{k=1}^{M} R(r_k)\sin^2\theta_k \;,\; H_{cs} = \sum_{k=1}^{M} R(r_k)\sin\theta_k \cos\theta_k \qquad (8)$$

Accordingly, the optimization can be formulated as

$$\max_{\varphi} F(\varphi) \Leftrightarrow \max_{\|x\|=1} xAx^T \Leftrightarrow \max_{\|x\|>0} \frac{xAx^T}{xx^T} \qquad (9)$$

It is actually a problem of maximizing Rayleigh Quotient [6]. It is known that its maximum is the largest eigenvalue of A that is reached at the unit eigenvector corresponding to the largest eigenvalue. By zeroing eigenpolynomial of A as follows

$$\lambda^2 - (H_{cc} + H_{ss})\lambda + H_{cc}H_{ss} - H_{cs}^2 = 0 \qquad (10)$$

the eigenvalues λ_1, λ_2 of A is given by

$$\lambda_1 = \frac{1}{2}(H_{cc} + H_{ss} + \sqrt{(H_{cc} - H_{ss})^2 + 4H_{cs}^2}) , \; \lambda_2 = \frac{1}{2}(H_{cc} + H_{ss} - \sqrt{(H_{cc} - H_{ss})^2 + 4H_{cs}^2}) \qquad (11)$$

Obviously, we have $\lambda_1 \geq \lambda_2$, this gives the largest eigenvalue $\lambda_{max} = \lambda_1$. It can be seen that $\lambda_1 = \lambda_2$ holds if and only if $(H_{cc} - H_{ss})^2 + 4H_{cs}^2 = 0$, or equivalently,

$$H_{cc} = H_{ss} \text{ and } H_{cs} = 0 \qquad (12)$$

In this case, from Eq. (7), $F(\varphi) = xAx^T\big|_{x=(\cos\varphi, \sin\varphi)} = H_{cc}$ holds for any φ so that no dominant direction exists. Provided Eq. (12) is not satisfied, we see $H_{cc} - \lambda_{max} = 0$ and $H_{ss} - \lambda_{max} = 0$ cannot be both true. By solving

$$\begin{pmatrix} H_{cc} & H_{cs} \\ H_{cs} & H_{ss} \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} = \lambda_{max}\begin{pmatrix} x \\ y \end{pmatrix} \qquad (13)$$

the eignevector of A that corresponds to λ_{max} can be given as follows

$$\begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \begin{cases} (-H_{cs}, H_{cc} - \lambda_{max})^T & (H_{cc} - \lambda_{max} \neq 0) \\ (H_{ss} - \lambda_{max}, -H_{cs})^T & (H_{ss} - \lambda_{max} \neq 0) \end{cases} \qquad (14)$$

So the optimal direction estimation $\varphi_{optimal}$ $(0 \leq \varphi_{optimal} < \pi)$ is given by

$$\varphi_{optimal} = \begin{cases} arctg(y_0/x_0) & x_0 y_0 \geq 0 \\ arctg(y_0/x_0) + \pi & x_0 y_0 < 0 \end{cases} \tag{15}$$

From Eq. (2), it is easy to show $F(\varphi) + F(\varphi^{\perp}) = H_{cc} + H_{ss}$ for any φ. Also noticing $F(\varphi_{optimal}) = \lambda_1$, it can be shown the inconsistency defined by Eq. (4) is given by

$$G(\varphi_{optimal}) = \frac{F(\varphi_{optimal}^{\perp})}{F(\varphi_{optimal})} = \frac{H_{cc} + H_{ss} - \sqrt{(H_{cc} - H_{ss})^2 + 4H_{cs}^2}}{H_{cc} + H_{ss} + \sqrt{(H_{cc} - H_{ss})^2 + 4H_{cs}^2}} \tag{16}$$

An example of $G(\varphi_{optimal}) = 1$ is when directions in Θ uniformly distribute over $[0,\pi]$ so that no direction can dominate over others. Also it can be easily shown that $G(\varphi_{optimal}) = 0$ if and only if all θ_k's, whose weights $R(r_k) > 0$, are in the same direction as $\varphi_{optimal}$ so that a perfect match between $\varphi_{optimal}$ and Θ is achieved.

Now we are going to compare our solution to Kass and Witkin' formulas [1,5]. Let

$$S = \sum_{k=1}^{M} R(r_k)\sin(2\theta_k),\ C = \sum_{k=1}^{M} R(r_k)\cos(2\theta_k),\ Q = \sum_{k=1}^{M} \sqrt{(R(r_k)\sin(2\theta_k))^2 + (R(r_k)\cos(2\theta_k))^2}$$

Their formulas can be written as

$$\varphi_{Kass} = \frac{1}{2}arctg(S/C) = \frac{1}{2}arctg(2H_{cs}/(H_{cc} - H_{ss})) \tag{17}$$

$$coherence = (\sqrt{S^2 + C^2})/Q = \sqrt{(H_{cc} - H_{ss})^2 + 4H_{cs}^2}/(H_{cc} + H_{ss}) \tag{18}$$

From Eq. (14), (15), and (16), it can be shown that

$$tg(2\varphi_{optimal}) = \frac{2H_{cs}}{H_{cc} - H_{ss}},\ coherence = \frac{1 - G(\varphi_{optimal})}{1 + G(\varphi_{optimal})} \tag{19}$$

So the algorithm proposed by Kass and Witkin is actually an optimal estimation under quadratically modulating the above-proposed general criterion function. The inconsistency associated with the optimal direction, although not exactly identical to their coherence value, is closely related and should be equivalent in real use.

4   Design Modulation Function f by Dynamic Weighting

Our optimizing method is based on projection of the primitive direction set $\{\theta\}$ onto the estimation φ. Since $\{\theta\}$ is usually corrupted by noise, to improve accuracy of the estimation, a strategy is to filter out the noisy signals from $\{\theta\}$. One of common practice to do this is to weight the directions in $\{\theta\}$ so that a noisier direction has less influence on estimation by being given a smaller weight. Since for a noise-free fingerprint image its local gradients should have a very small variation, a reasonable hypothesis is: if φ is the true direction, the larger the angle between θ and φ, the

noisier θ is. In other words, the larger the angle between θ and φ, the smaller weight should be assigned to θ. Here we face again to the fact that the angle between θ and φ should be measured so that it keeps unchanged when either θ or φ turns into its opposite. So similar to Section 2, we define a weighting function based on $|\cos(\theta - \varphi)|$. Accordingly, a primitive weighting function is defined as

$$W_h(\theta) = \frac{1}{S_h} h(|\cos\theta|) \qquad (S_h = \int_0^\pi h(|\cos x|)dx) \qquad (20)$$

here $h(x)$ is a non-negative increasing function. Obviously $W(\theta)$ is a periodic function with period being π. Now we can define a weighting function as

$$DW_h(\theta,\varphi) = W_h(\theta - \varphi) \qquad (21)$$

It can be easily shown that $DW(\theta,\varphi)$ is maximized at $\theta = \varphi$ and decreases as $|\cos(\theta - \varphi)|$ is decreasing, which indicates the noisy signal carried by θ is increasing. Thus $DW(\theta,\varphi)$ really weights the signal based on its noise level. By weighting the projection (see Eq. (1)) using $DW(\theta,\varphi)$, the modulation f is defined as

$$f(|\cos(\theta - \varphi)|) = DW(\theta,\varphi)|\cos(\theta - \varphi)| = \frac{1}{S_h} h(|\cos(\theta - \varphi)|)|\cos(\theta - \varphi)| \qquad (22)$$

Since the weighting $DW(\theta,\varphi)$ on a directional data θ is included in f, it is built into optimizing process. Differing from traditional weighting, this scheme has the weight assigned to each θ automatically and dynamically adjusted by optimizing process itself, and the weighting itself will be optimized as well in the process. So we call it dynamic weighting. As a result, an optimal weighting, which matches the optimal direction the best, is always in effect in the final estimate.

Now let us consider a function defined by

$$C_h(t) = \int_{-t}^{t} W_h(t)dt \qquad (0 \le t \le \frac{\pi}{2}) \qquad (23)$$

$C_h(t)$ measures how concentrating the primitive weighting is around the direction 0. If

$$C_{h_1}(t) \ge C_{h_2}(t) \qquad \text{for any } t \in [0,\pi/2] \qquad (24)$$

we call W_{h_1} is more concentrating than W_{h_2}, which means W_{h_1} has its energy more localized than W_{h_2} does. In Figure 1(c), for instance, it can be seen that W_3 is more concentrating than W_2, and W_2 is more concentrating than W_1. By (21), a more concentrating weighting has signal closer to the true direction φ treated more critical (i.e. weighted more heavily) so that the resultant estimate is less affected by the noisy data that are supposed to be farther from φ. Therefore a more concentrating weighting may be expected to give a more accurate estimate.

5   Accuracy Analysis on Optimizing with Dynamic Weighting

We will analyze several dynamic weighting systems by simulation. To simplify our discussion, we set $R(r) \equiv 1$. The following lists the criterion functions $\{F_i(\varphi)\}$ as well as corresponding dynamic weighting functions we are going to study.

$$F_1(\varphi) = \frac{1}{S_1} \sum_{k=1}^{M} \cos^2(\theta_k - \varphi) \qquad\qquad (W_1(\theta) = \frac{1}{S_1} |\cos\theta|)$$

$$F_2(\varphi) = \frac{1}{S_2} \sum_{k=1}^{M} \cos^4(\theta_k - \varphi) \qquad\qquad (W_2(\theta) = \frac{1}{S_2} |\cos\theta|^3) \qquad (25)$$

$$F_3(\varphi) = \frac{1}{S_3} \sum_{k=1}^{M} e^{-\frac{(1-|\cos(\theta_k-\varphi)|)^2}{0.2}} |\cos(\theta_k - \varphi)| \qquad (W_3(\theta) = \frac{1}{S_3} e^{-\frac{(1-|\cos(\theta_k-\varphi)|)^2}{0.2}})$$

$F_1(\varphi)$ has been discussed in Section 3 and its optimal estimate is identical to that by Kass and Witkin. In Figure 1(c), their corresponding primitive dynamic weighting functions $W_i(\theta)(1 \le i \le 3)$ are shown for $-90° \le \theta \le 90°$. It is seen that, in order of W_1, W_2, W_3, the dynamic weightings get more and more concentrating so that it is expected that the resultant direction estimates should become more and more accurate. Experiments on synthesized data will be presented to verify this prospect.

Consider a noisy fingerprint image patch shown in Figure 1(a). We use Gaussian functions to simulate the distribution $D(\theta)$ of the directions in the primitive direction set Θ of this image.

$$D(\theta) = \max(k_{ridge} e^{-\frac{\theta^2}{2\sigma^2}}, k_{crease} e^{-\frac{(\theta-\theta_0)^2}{2\sigma_0^2}}) \qquad (0 \le \theta < \pi) \qquad (26)$$

By this formula, the ridge direction is 0 (horizontal), and the crease runs in direction θ_0. σ and σ_0 tell how wide the directions, respectively from ridge and crease, vary. k_{ridge} and k_{crease} specify strength of respective ridge and crease. It is assumed that $k_{ridge} \ge k_{crease}$. Figure 1(b) shows an example of $D(\theta)(0° \le \theta \le 180°)$ with $k_{ridge} = 1.2$, $k_{crease} = 1$, $\sigma = \sigma_0 = \pi/9$ (i.e. $20°$ in degree), and $\theta_0 = 70°$.

Given $D(\theta)$, an effective direction estimation should be close to 0. In our experiments, $D(\theta)$ is variously parameterized to simulate different relative noise level, i.e. normal ($k_{ridge} = 2$), high ($k_{ridge} = 1.2$), and very high ($k_{ridge} = 1$) with $k_{crease} = 1$ for all the cases. Also θ_0 varies from $30°$ through $90°$ to simulate different angles crease runs. $\sigma = \sigma_0 = \pi/9$, however, is fixed. For each variation of $D(\theta)$, $F_i(i = 1,2,3)$ are applied on it and maximized to get the optimal estimations. An estimation closer to 0, therefore, means a better performance of the underlying dynamic weighting. Table 1 shows the optimal directions (in degree) found with F_i under differently parameterized $D(\theta)$. Since in some cases, there may be 2 estimates that maximize F_i exist, under each F_i, 2 columns are included to present the values. When the corresponding estimate does not exist, the cell is left blank.
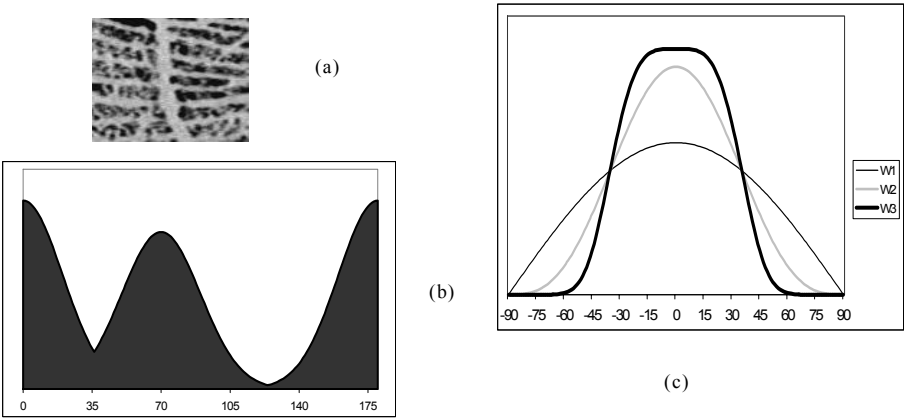
Fig. 1. (a) A patch from a fingerprint image that shows horizontal ridge with a crease in vertical. (b) Distribution of directions in the primitive direction set, which is derived by turning gradient directions by $\pi/2$, of the image (a). The axis of direction uses degree, instead of radian, as unit. (c) 3 primitive weighting functions W_1, W_2, W_3 defined in Eq. (25)

From the results presented in Table 1 , we observe followings

(1) If the angle between ridge and crease directions is under a certain value so that the directions from noise and ridge can not be effectively seperated from each other, the estimating usually behaves like an average weighted by the signal strength. So the higher the ratio of signal to noise is, the better the estimate is.

(2) If the angle between ridge and crease directions becomes larger so that a separation of noisy signal from ridge signal becomes possible, a highly concentrating weighting, like W_2, W_3, will try to seek a solution based on ridge signal only due to the noise suppress by dynamic weighting.

(3) If the ridge has a strength not distinguished from that of noise, a highly concentrating weighting system, like W_2, W_3, tries to provide multiple solutions correspond to the signals with the similar strength. This is a good nature since a proper selection procedure can be adopted to deal with very noisy images.

Table 1. Optimal estimates of dynamic weighting system W_1, W_2, W_3

θ_0	$k_{ridge}=1, k_{crease}=1$						$k_{ridge}=1.2, k_{crease}=1$			$k_{ridge}=2, k_{crease}=1$		
	F1		F2		F3		F1	F2	F3	F1	F2	F3
30	15		15		15		13	13	13	6	5	6
40	20		20		20		17	15	16	9	6	7
50	25		25		25		21	17	19	12	7	6
60	30		30		30		25	15	15	13	5	4
70	35		13	57	6	64	27	8	4	12	3	1
80	40		4	76	1	79	25	3	1	8	1	0
90			0	90	0	90	0	0	0	0	0	0

(4) A more concentrating dynamic weighting provides more precise estimation. W_2, W_3 perform better than the Kass and Witkin's algorithm (W_1).

As a conclusion, by use of highly concentrating dynamic weighting systems like W_2, W_3, direction estimate can be improved significantly in accuracy. It, however, has to be aware that a highly concentrating weighting system may not provide an analytic solution, which calls for numerical methods being used and needs a further study.

References

[1]   Kass M. and Witkin A., "Analyzing Oriented Patterns", *Computer Vision, Graphics, and Image Processing*, Vol.37, No.3, pp. 362-385, 1987.

[2]   Davide Maltoni, Dario Maio, Anil K. Jain, and Salil Prabhakar, "Handbook of Fingerprint Recognition", Springer-Verlag, New York, Inc., 2003.

[3]   Lin Hong, Yifei Wan, and Anil Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluaton", IEEE Trans. On PAMI, Vol.20, No.8, pp.777-789.

[4]   Cappelli R., Lumini A., Maio D., and Maltoni D., " Fingerprint Classification by Directional Image Partitioning", IEEE Trans. On PAMI, Vol.21, No.5, pp.402-421, 1999.

[5]   Asker M. Bazen and Sabih H. Gerez, "Directional Field Computation for Fingerprints Based on the Principal Component Analysis of Local Gradients", Proceedings of the ProcRISC/IEEE workshop, pp.215-222, 2000.

[6]   Lloyd N. Trefethen and David Bau, "Numerical Linear Algebra", Society for Industrial and Applied Mathematics, October 2000.

Fingerprint Reference Point Detection

Manhua Liu[1], Xudong Jiang[2], and Alex Chichung Kot[1]

[1] EEE, Nanyang Technological University, 639798, Singapore
{pg05080538,eackot}@ntu.edu.sg
[2] Institute for Infocomm Research, 21 Heng Mui Keng Terrace, 119613, Singapore
xdjiang@i2r.a-star.edu.sg

Abstract. A robust fingerprint recognition algorithm should tolerate
the translation and rotation of fingerprint image. This paper proposes
an algorithm to locate a unique reference point and compute a unique
reference orientation for all types of fingerprints, which can be used for
translational and rotational alignments. The reference point localization
is based on scale space analysis of Poincare index (PI) and orientation
variance (OV), and the reference orientation is computed based on anal-
ysis of orientation differences between the radial directions from the
reference point and the local ridge orientations along these radials. In
addition, a nonlinear smoothing method with better performance in fil-
tering impulse noise and the noise caused by scars and ridge breaks etc.
is proposed to compute the orientation fields of varying scales except
for the finest scale. Experimental results demonstrate that the proposed
algorithm can locate the reference point and compute the reference ori-
entation with high accuracy for all types of fingerprints.

1 Introduction

Fingerprint is widely used for personal identification due to its easier accessibility,
uniqueness and reliability. Consistent feature extraction is crucial for fingerprint
recognition. However, translation and rotation usually exists among different fin-
gerprint samples from the same finger. One popular solution is to consistently
locate a unique reference point and compute a unique reference orientation for
alignment. Since singular points, i.e. core and delta points, are unique land-
marks of fingerprint, they were commonly used for fingerprint classification as
reference points [1], [2]. However, we often get partial fingerprint image with the
delta points left outside and plain arch fingerprints have no singular point. In
order to consistently locate a unique reference point for all types of fingerprints
and the partial fingerprints, we define the reference point as the point with the
maximum curvature on the convex ridge, which is usually located in the central
area of fingerprint (see Fig. 1). Therefore, if core points exist in a fingerprint, the
core point on the convex ridge is the reference point. As for the corresponding
reference orientation, it should be unique and can be consistently determined to
reflect the rotation of fingerprint image.

There are many approaches proposed for singular point detection in litera-
tures. Karu and Jain [1] and Zhang et al. [2] referred to a PI method. But this
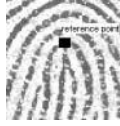
Fig. 1. Convex ridges of fingerprint and the reference point

method is sensitive to noise and unable to detect the reference point in plain arch fingerprint. Koo and Kot [3] proposed a method of singular point detection based on finding the maximum curvature. This method is also sensitive to noise and ineffective for poor quality fingerprints. Jain et al. [4] proposed a method to locate a reference point by multiple resolution analysis of the differences of the sine component integration between two defined regions of the orientation field. Although this method can handle gracefully the local noise, the localization of reference point is sensitive to the fingerprint rotation and not very consistent for plain arch fingerprints.

This paper proposes a method to consistently locate a unique reference point for all types of fingerprints based on multi-scale analysis of PI and OV. To compute reliable orientation fields of varying scales, we estimate the finest scale orientation using the gradient averaging method and propose a nonlinear method to estimate the orientations of larger scales. The PI method is employed from the largest scale to reduce the search region of the reference point. However, this method usually locates more than one candidate points near the true core point in the finest scale, and it fails to work when no core point exists in plain arch fingerprint. The OV, which is sensitive to the local orientation change, is employed to replace the PI in above two cases. Furthermore, this paper proposes a method to compute a reference orientation associated with the reference point. The following sections present our proposed algorithm concerned with the reference point localization and the reference orientation computation in detail.

2   Localization of the Reference Point

A good method of reference point localization should be able to consistently locate a unique reference point for all types of fingerprints with noise robustness. To suppress the local noise, a nonlinear smoothing method is proposed to compute the reliable orientation fields of larger scales. The reference point is located based on the scale space analysis of PI and OV.

2.1   Orientation Smoothing

Many methods are proposed for orientation estimation in literatures. The gradient squared averaging method is a conventional method and most widely used in orientation estimation [4],[5] because of its high resolution and efficiency. It is employed in this work to compute the finest scale orientation. This averaging

(a)   (b)

Fig. 2. The orientation fields of a poor quality fingerprint with a scar smoothed using: (a) the conventional averaging method and (b) the proposed nonlinear method

smoothing method is a kind of linear method and effective to attenuate Gaussian noise, but it cannot work well to suppress impulse noise of the orientation field. Unfortunately, impulse noise such as the noise caused by scars and ridge breaks etc. usually exist in poor quality fingerprints. The orientation in this noisy area has large deviation from the accurate local ridge orientation while using the averaging smoothing method. A nonlinear smoothing method is proposed to compute the orientation fields of larger scales. Let $\theta(i,j)$ ($\in [-\pi/2, \pi/2)$) be the orientation of the finest scale block (i,j), the larger scale orientation on a smoothing window W_i of size $w_i \times w_i$ finest scale blocks is estimated as:

$$Var|_{k_{min}} = \min_{k=1}^{M} \frac{1}{w_i^2} \sum_{(i,j) \in W_i} |\sin(\theta(i,j) - (-\pi + (k-1) \times \pi)/M)| \qquad (1)$$

$$\Psi = (-\pi + (k_{min} - 1) \times \pi)/M \qquad (2)$$

where M (8 in our experiment) is the number of quantified orientations in $[-\pi/2, \pi/2)$. The basic idea of this nonlinear method is to find a dominant orientation that has minimum orientation difference with all the orientations in its neighborhood. If most orientations in its neighborhood are homogenous, a small number of orientations with strong deviation caused by local noise do not influence on the final orientation estimation. Therefore, this nonlinear method works better in filtering the impulse noise and the noise caused by the scars and ridge breaks etc. than the conventional averaging method (see Fig. 2). With different W_{i-1}, we can get the multi-scale orientations. The proposed method with $M \times w_i \times w_i$ evaluations of trigonometric function is more time-consuming than the averaging method and the estimated orientation is quantified. We compute the orientations of larger scales in the proposed nonlinear method.

2.2   Reference Point Localization

This work proposes a multi-scale space method to consistently locate a unique reference point for all types of fingerprints. The PI method [2] is employed to reduce the search region of reference point from the largest scale to a much smaller singular region. The PI of block (i,j) is computed by summing up the direction changes through full counter-clockwise turn around a digital curve of a 3×3 rectangle, i.e. around the 8 outside surrounding blocks of the rectangle. It

(a)   (b)   (c)
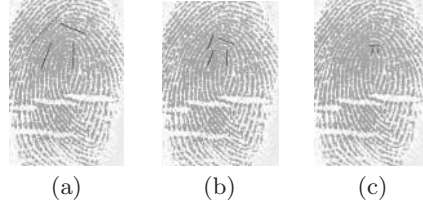
Fig. 3. Orientation fields of varying scales in the neighborhood of reference point: (a) the largest scale, (b) one of the middle scales and (c) the finest scale

takes the value 1, 1/2, -1/2 and 0 for two core points, a core point, a delta point and an ordinary point respectively. Therefore, the blocks with PI larger than 3/8 are located as the effective search region of core point for a little robustness. Furthermore, if the orientation difference between the upper left and the upper right block, i.e. $\theta(i-1, j-1) - \theta(i-1, j+1)$, and the orientation of upper left block $\theta(i-1, j-1)$ are larger than 0, block (i, j) is on the convex ridge and considered as candidate search region of reference point in the next finer scale, otherwise it is not a concern. These steps iterate through varying scales until the finest scale or no blocks with PI larger than 3/8 (see Fig. 3).

However, there are two problems for the PI method. One is that no blocks with PI larger than 3/8 exist in fingerprint such as in plain arch fingerprint. Another problem is more than one block in the finest scale located as the reference point. Therefore, the OV, which is sensitive to the local orientation change, is employed to replace the PI to locate a unique reference point in above two cases.

The normal way of variance computation is not applicable to compute OV because of the π periodicity of local ridge orientation. We compute the OV in equation (1) based on a 3×3 neighborhood. $Var|_{k_{min}}$ approximates to 0 if the orientations over the neighborhood are homogeneous, and increases when they are discordant. Thus, $Var|_{k_{min}}$ describes variance of the orientations over the neighborhood. The whole process of our proposed multi-scale space method of reference point localization is summarized as below:

1. Estimate the orientation of each finest scale block (of size 9×9 pixels in our experiments) using the gradient averaging method.
2. Estimate the local ridge orientation of each largest scale block (of size $w_i \times w_i$ finest blocks) using the proposed nonlinear smoothing method in (2).
3. Compute the PI of each block. The blocks on the convex ridge with PI larger than 3/8 are selected as the search region of reference point in the next finer scale. If no such blocks exist, the OV of each block is computed using equation (1). The block with maximum OV is selected as the search region of reference point in the next finer scale.
4. Estimate the orientation of each block with finer scale $w_{i-1}(w_{i-1} < w_i)$ in the obtained smaller search region from the above step using (2). Repeat steps 3-4 until w_{i-1} equals to 1.
5. Compute the OV of each candidate block in the finest scale in (1) and locate the block with maximum OV as the unique reference point.

3   Computation of the Reference Orientation

There are few papers concerned with computing a reference orientation. Asker Bazen [6] proposed a method to estimate an orientation associated with singular point by computing the orientation difference between the detected singular region and an ideal singular region of the same type. This method works well if the singular region is very similar to the ideal model. But most singular regions have ridge flow patterns different from the ideal model, and this method is inapplicable for plain arch fingerprint. In this work, the reference orientation is defined based on analysis of the orientation differences between radial directions from the reference point and the local ridge orientations along these radials (see Fig. 4a). For the fingerprint whose reference point is core point, the local ridge orientation most parallel to the corresponding radial direction is unique and defined as the reference orientation (see Fig. 4b). For plain arch fingerprint, however, there exist two different such orientations, the average of which is defined as the unique reference orientation (see Fig. 4c). If a radial direction has minimum difference with a set of local ridge orientations along the radial, the dominant orientation of the set of local ridge orientations is considered most parallel to the radial direction. The orientation difference is computed as:

$$V(k) = \frac{1}{M} \sum_{(i,j) \in \Omega_k} |sin((\theta(i,j) - \theta_k)|, \theta_k = k\pi/8, k = 0, 1, ..., 15. \qquad (3)$$

where Ω_k is a set of M local ridge orientations along the radial with direction θ_k. Obviously, $V(k)$ with range [0 1] is 1 if θ_k is orthogonal to the orientations in Ω_k and 0 if it is parallel to them. If only one local minimum exists in $V(k)$, the dominant orientation of Ω_k corresponding to minimum $V(k)$ is computed as the unique reference orientation. If $V(k)$ is bipolar and contains two local minimums such as in plain arch fingerprint, two dominant orientations along the radials with local minimum $V(k)$ are computed and the average of them is computed as the unique reference orientation. To consistently compute the reference orientation for all types of fingerprints, the length of Ω_k is initialized as 3 (The minimum distance between two core points is assumed as 3 finest scale blocks) and adapted by analysis of $V(k)$.

| (a) | (b) | (c) |
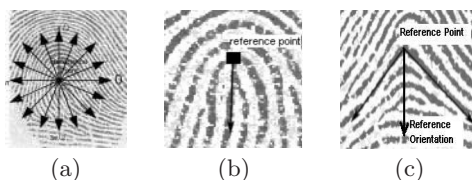
Fig. 4. (a) 16 radial directions from the reference point, (b) the reference orientation for core point, and (c) the reference orientation for plain arch fingerprint

4   Experimental Results

The proposed algorithm of reference point localization and reference orientation computation has been tested on the FVC2000 DB2 set A (800 fingerprints). The poor quality fingerprints in the database are all included for testing. To quantitatively evaluate the proposed algorithm, the desired location and orientation of the reference point are manually determined by visual inspection for each fingerprint. The Euclidean distance between the desired and the estimated position (the central pixel of the detected finest scale block) is computed as the distance error (DE) of reference point localization. Similarly, the orientation difference between the desired and computed reference orientation is considered as the orientation error (OE) of the reference orientation computation.

The test database contains 13 partial images with the reference point left outside. In our proposed algorithm, the detected reference point in this case is usually located on the boundary of fingerprint (see Fig. 5a), and this kind of fingerprint is differentiated as partial image since the reference point is not consistent with that in other sample images. However, if the reference point of the partial image is located in the inner region of fingerprint (see Fig. 5b), this localization is considered as wrong detection. The proposed algorithm can correctly detect 11 of them with only 2 failures in this way.

Since the DE computation is influenced by the resolution of the finest scale block and the accuracy of the manually detected position, we define three coarse level measures of the DE to evaluate the accuracy of reference point localization (see Table 1). If the DE is not larger than 10 pixels, the localization is regarded as accurate. For the DE larger than 10 pixels but not larger than 20 pixels, the detected reference point has small location error and still can be used in fingerprint recognition. However, if the DE is larger than 20 pixels, the localization is regarded as wrong, which will result in significant error of fingerprint recognition.

Similarly, three coarse level measures of OE is also defined to evaluate the accuracy of reference orientation computation. The reference orientation computation in our proposed algorithm depends on the reference point localization. Therefore, we analyze the computation error of the reference orientation in terms of the accuracy of reference point localization (see Table 2).

(a)   (b)
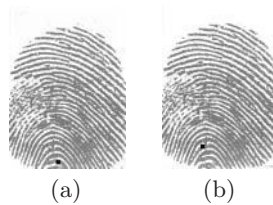
Fig. 5. reference point localization in Partial fingerprints: (a) located on the boundary of fingerprint. (b) located in the inner region of fingerprint

Table 1. Accuracy analysis of the reference point localization

Reference point	$DE \leq 10pixels$ (Accurate)	$10pixels < DE \leq 20pixels$ (Small error)	$DE > 20pixels$ (Significant error)
Core point (672)	571 (84.97%)	80 (11.90%)	21 (3.13%)
Non-core point (115)	62 (53.91%)	20 (17.39%)	33 (28.70%)
Total (787=800-13)	633 (80.43%)	100 (12.71%)	54 (6.86%)

Table 2. Accuracy analysis of the reference orientation computation

Distance error (DE) ($pixels$)	$OE \leq \pi/16rads$ (Accurate)	$\pi/16rads < OE \leq \pi/8rads$ (Small error)	$OE > \pi/8rads$ (Significant error)
≤ 10 (633)	587 (92.73%)	19 (3.00%)	27 (4.27%)
> 10 and ≤ 20 (100)	70 (70.00%)	15 (15.00%)	15 (15.00%)
> 20 (54)	24 (44.44%)	11 (20.37%)	19 (35.19%)
Total (787)	681 (86.53%)	45 (5.72%)	61 (7.75%)

Almost all of the significant errors of our proposed algorithm are resulted from the heavy noise, which cannot be well attenuated in the very poor quality fingerprints (see Fig. 6).

From above experimental results, we can see our proposed algorithm works well to consistently locate a unique reference point and compute the corresponding reference orientation for all types of fingerprints. And the experimental results can be further improved by reducing the size of the finest scale block.

5   Conclusions

This paper proposes an effective algorithm to consistently locate a unique reference point and compute a unique reference orientation for all types of fingerprints, which can be used for translational and rotational alignment. In addition, a nonlinear smoothing method is proposed to compute the reliable orientation fields of larger scales. Experimental results show that the proposed algorithm performs well in terms of accuracy for all types of fingerprints.
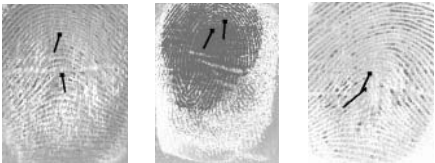
Fig. 6. Some examples of reference point localization and reference orientation computation with significant error

References

[1] K. Karu, A. K. Jain: Fingerprint Classification. Pattern Recognition, **29** (1996) 389-404. 272

[2] Q. Zhang, K. Huang and Hong Yan: Fingerprint Classification Bsaed on Extraction and Analysis of Singularities and Pseudoridges. The Pan-Sydney Area Workshop on Visual In-formation Processing (VIP2001), Sydeney, Australia **11** (2002) 83-87. 272, 274

[3] Wai Mun Koo and Alex Kot: Curvature-Based Singular Points Detection. 3rd International Conference on Audio- and Video-Based Biometric Person Authentication, Lecture Notes in Computer Science (LNCS) **2091** (2001) 229-234. 273

[4] Anil K. Jain, Salil Prabhakar, Lin Hong, and Sharath Pankanti: Filterbank-Based Fingerprint Matching. IEEE Transactions on Image Processing, **9** (2000) 846-859. 273

[5] D. Maio and D. Maltoni: Direct Gray-Scale Minutiae Detection in Fingerprints. IEEE Transactions on Pattern Analysis and Machine Intelligence, **19** (1997) 27-40. 273

[6] Asker M. Bazen and Sabih H. Gerez: Systematic Method for the Computation of the Directional Fields and Singular Points of Fingerprints. IEEE Transactions on Pattern Analysis and Machine Intelligence, **24** (2002) 905-919. 276

Direct Gray-Scale Minutiae Extraction

A. Wahab, E. C. Tan, and A. Jonatan

School of Computer Engineering
Nanyang Technological University, Nanyang Avenue, Singapore 639798, Singapore
`{asabdul,asectan,G896208}@ntu.edu.sg`

Abstract. Most existing minutiae extraction methods require image pre-processing which has certain drawbacks. Direct gray-scale minutiae extraction approaches that work on the original image is thus preferred. The use of fuzzy neural network (FNN) as a recognition system to detect the minutiae pattern has been shown to be promising and several types of FNNs have been proposed. Here, we propose a new approach to perform direct gray-scale minutiae extraction based on a more powerful generic self-organizing FNN (GenSoFNN).

1   Introduction

Most fingerprint verifications are based on the matching of minutiae patterns. Minutiae have discontinuities that interrupt the otherwise smooth flow of ridge. By locating and comparing the minutiae of two fingerprints, one can tell whether the fingerprints are from the same source. Minutiae extraction process falls into two major categories, the classical and direct gray-scale approaches [1]-[2]. The classical approach which is widely used today, incorporates image-processing steps like equalization, binarisation, thinning, and so on. However, there are several drawbacks in this approach such as the loss of valuable data and the inability to differentiate between noise and the expected image. The direct gray-scale approach extracts the minutiae from the fingerprint image in its gray-scaled format directly using image-based features [2], the following of ridgelines [3], or the neural network and fuzzy logic. It does not require any pre-processing and post-processing. Therefore, it has a higher computational efficiency and a faster response. Moreover, there are several other reasons that this method is preferred to the classical approach: (i) A lot of information may be lost during the binarisation process. (ii) Binarisation and thinning are tedious and time consuming. (iii) Some of the pre-processing procedures, especially the binarisation technique, are proven unsatisfactory when applied to low quality images. (iv) The errors generated during pre-processing tend to propagate from one stage to another.

The use of FNN in fingerprint image processing has been reported [4]. A specific type of FNN called the Pseudo Outer Product Fuzzy Neural Network (POPFNN) [5] is employed and trained to recognize the shape of ridge ending and bifurcation. However, its computational complexity is high. Since then, different types of FNNs have been developed with increasing level of accuracy and generalization ability. We propose a new way to perform direct gray-scale minutiae extraction based on a recent

FNN architecture called GenSoFNN [6]. Its performance is compared with that of another FNN known as Falcon-MART [7] using fault acceptance rate (FAR), fault rejection rate (FRR) and response time.

2   Fuzzy Neural Network (FNN)

FNN is the integration of fuzzy logic [8] and artificial neural network. Its main advantage is the ability to model a problem domain using linguistic instead of a complex mathematical model [9]. The linguistic model is a fuzzy-rule base consisting of a set of IF-THEN rules that are highly intuitive and easily understood by human users.

2.1 Falcon-MART

It is a modification of the original Falcon-ART [10] architecture. Similar to POPFNN, Falcon-ART also has five layers. Falcon-ART performs poorly when two or more groups of data are very close or similar to each others, giving rise to rules that fire with similar strengths. To overcome this, any difference between the respective elements is magnified in Falcon-MART. Moreover, to minimize the effects of noisy data, a more stable and progressive learning is adopted in Falcon-MART. A pseudo termination criterion is used to stop the learning process when the change in error is sufficiently small. Nevertheless, the FNN suffers from inconsistency of the rule base because of the inherent clustering characteristics of adaptive resonance theory (ART).

2.2 GenSoFNN

The GenSoFNN adopts the Mamdani's fuzzy model [9] and two motivations drove its development. The first is to define a systematic way of crafting the linguistic model required in fuzzy neural systems and the second is to create a generalized network architecture whereby different fuzzy inference schemes can be mapped onto such a network with ease. GenSoFNN models a problem by first performing a cluster analysis of the numerical training data and subsequently deriving the fuzzy-rule base from the computed clusters. The training cycle of the GenSoFNN consists of three phases: self-organizing, rule formulation and parameter learning. These are performed sequentially with a single pass of the training data. The discrete incremental clustering (DIC) technique is developed and incorporated into the GenSoFNN to automatically compute the input-output clusters from the numerical training data. The fuzzy rules are subsequently formulated by connecting the appropriate input and output clusters during the rule-mapping phase of the training cycle. Consequently, the popular back-propagation learning algorithm [11] based on negative gradient descent is employed to tune the parameters.
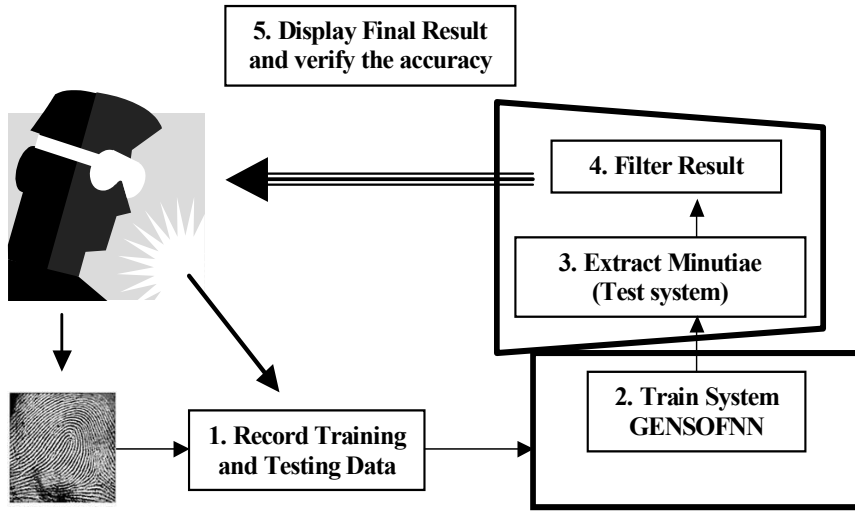
Fig. 1. Minutia extraction process

3   System Architecture

The minutiae extraction system works in the sequences as shown in Fig. 1. Owing to similarities of pattern in pixels next to each other, filtering is performed to remove minutiae points that are too close to each other, and maintain the strongest point.

There are two types of minutiae observed: ridge ending and bifurcation. In overall instances of fingerprint images, both ridge ending and bifurcation are rather similar. For ridge ending, it is 4-by-4 dark pixels surrounded by bright pixels in three directions, and vice versa for the bifurcation as shown in Fig. 2. From that understanding, only several areas are of concern to detect the existence of a minutia. The first is the average value of 4x4 pixels in the centre of the minutia. Therefore, inputs to the FNN are these average values of the area of interest. The first step is defining various windows to capture the information (level of brightness or darkness) surrounding a minutia point. The central block represent the average value of the minutiae, and the other windows represent the surrounding, where

$$Sum(x, y) = \sum_{i=0}^{I} \sum_{j=0}^{J} Val(x+i, y+j); \quad Ave(x, y) = Sum(x, y)/(IJ)$$

Val (x, y) = gray level value of the pixel, *I* = horizontal window size, J = vertical window size, and *Ave* (x, y) = average gray level value of the window. The average value is captured by using the windows in Fig. 3.

The minutia might be facing different directions and thus the orientation is of interest as well. Because of the 4x4 window, only 153x153 pixels are recorded for testing data. Each detectable pixel is captured eight times in eight different directions. Therefore, the total number of testing samples in this case is (153*153*8) = 187272 instances. A global mean, 12x12 (Fig. 4), is added to increase the accuracy. However,

this system could not differentiate between a real termination and a sharp corner in a ridge. Therefore, it is necessary to find out whether the detected minutia has a sufficient length by capturing the tail as shown in Fig. 5.
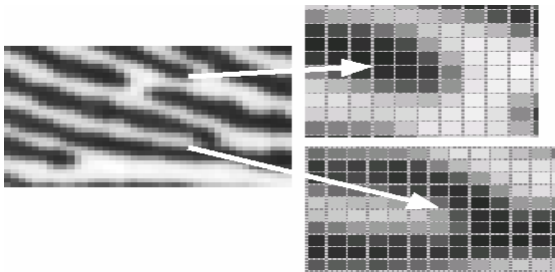
Fig. 2. Zoomed ridge ending (top) and bifurcation (below)
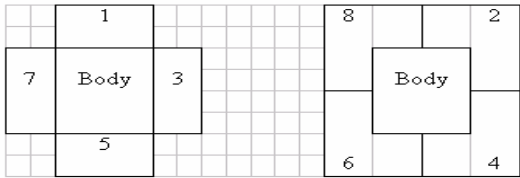
Fig. 3. Modeling windows
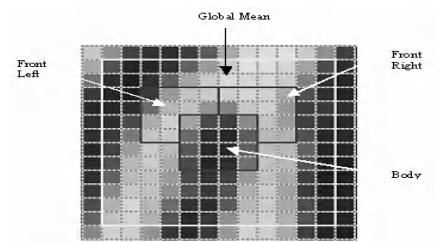
Fig. 4. Additional window for 8-input system   **Fig. 5.** Tail window
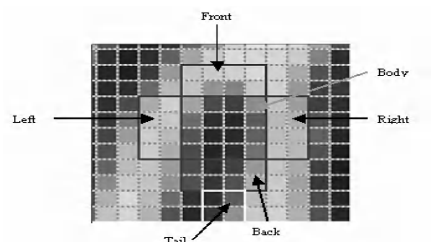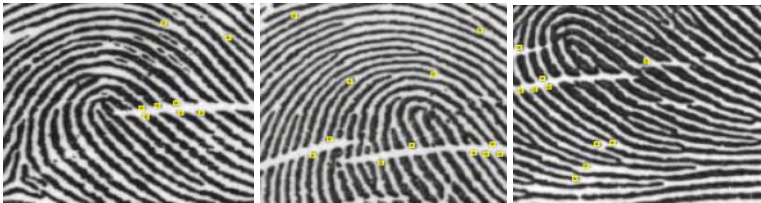
Fig. 6. FP1, FP2 and FP3 fingerprints for termination
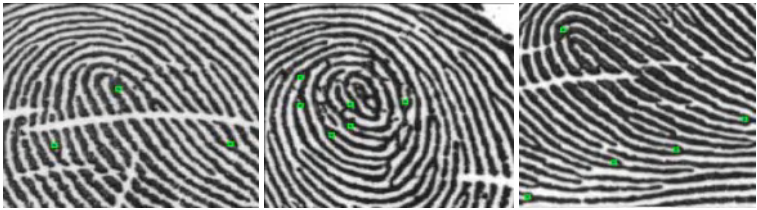
Fig. 7. FP4, FP5 and FP3 fingerprints for bifurcation

4   Experiments and Results Comparison

Figs. 6 and 7 show five images (2 FP3) that were used as samples in this minutiae extraction system; all are 150x150 in dimension.

Several metrics were used to compare experimental results, namely, False Acceptance Ratio (FAR), Miss-Ratio, and False Rejection Rate (FRR) which are defined as follows, where the total number of testing samples is 23409x8 = 187272.

FAR = 100% x (Num. of false detected points)/(Total num. of testing samples)
Miss-Ratio = (Num. of false minutiae detected)/(Num. of expected minutiae)
FRR = 100% x (Num. of undetected points)/(Total num. of testing samples)

The results in Tables 1 and 2 show a good performance in both detection capability and rejection of false points. It could be seen from the significant decrease in both FRR and FAR or Miss-Ratio. Examples of overall improvement are also shown in Fig. 8. The results of classical vs. FNN approach are shown in Tables 3 and 4.

5   Conclusion

The direct gray-scale minutiae extraction has been a sought-after method for automatic fingerprint verification. A novel approach to do that has been implemented using GenSoFNN. It is a FNN that has been proven to excel in relation with other FNNs. From our experiment, GenSoFNN has been proven a better substitute for Falcon-MART in terms of generalizing the problem domain. It was observed that GenSoFNN performs better as more training set can be included compared with the Falcon-MART.
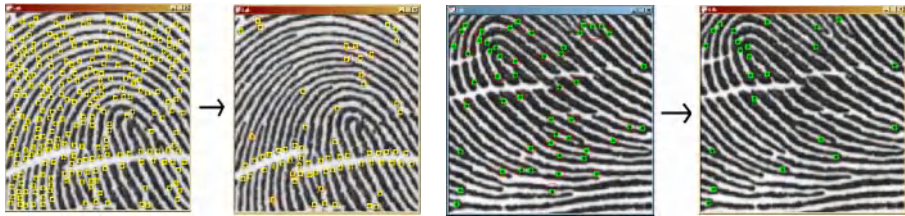
Fig. 8. Improvement for ridge-ending detection on FP2 (left), and for bifurcation detection on FP3 (right)

Table 1. Comparison of 5-ip using Falcon-MART and 9-ip using GenSoFNN

Image	Information	Undetected	FRR	FALSE	FAR	MissRatio
FP1 (30) Ending	previous	2	6.67	30	0.16	1.00
	modified	1	3.33	15	0.08	0.50
FP2 (53) Ending	previous	3	5.66	223	1.22	4.21
	modified	4	7.55	11	0.06	0.21
FP3 (32) Ending	previous	3	9.38	130	0.71	4.06
	modified	2	6.25	6	0.03	0.19

Table 2. Comparison of 5-ip using Falcon-MART and 9-ip using GenSoFNN

Image	Information	Undetected	FRR	FALSE	FAR	MissRatio
FP4 (6) Bifurcation	previous	3	50.00	41	0.22	8.20
	modified	3	50.00	2	0.01	0.33
FP5 (32) Bifurcation	previous	25	21.88	127	0.69	4.54
	modified	28	12.50	11	0.06	0.34
FP3 (16) Bifurcation	previous	15	6.25	117	0.64	7.31
	modified	15	6.25	5	0.03	0.31

Table 3. Comparison between Classical and FNN approach

	Classical Approach	FNN Approach
FRR	11.25%	14.31%
Miss-Ratio	22.29%	31.41%

Table 4. Result from Simon-Zoria *et al.* using classical approach

No:	Expected	Detected	Undetected	False
1	28	25	0	6
2	50	45	5	5
3	46	45	1	15
4	56	48	8	7
5	49	44	5	11
6	53	45	8	8
7	60	50	10	8
8	51	46	5	18
9	48	38	10	9
10	30	29	1	18
Total	471	415	53	105

(FRR = 11.25%, Miss-Ratio = 0.22)

References

[1]   Simon-Zorita, D.: Minutiae Extraction Scheme for fingerprint recognition systems. IEEE Int. Conf. on Image Processing. 3 (2001) 254–257.
[2]   Tico, M., Immonen, E., Ramo, P., Kuosmanen, P., Saarinen, J.: Fingerprint Recognition Using Wavelet Features. ISCAS 2001 IEEE International Symposium 2 (2001) 21 -24.
[3]   Maio, D., Maltoni, D., Cappelli, R., Wayman, J.L., Jain, A.K.: FVC2000: Fingerprint Verification Competition. IEEE Trans. Pattern Analy. & Machine Intell. 3 (2002) 402-412.
[4]   Quek, C., Wahab, A.: Direct Gray-scale Minutiae Extraction in Fingerprint Identification. Proc. ISCA 13th Int. Conf. on Parallel and Distributed Computing system. Las Vegas, Nevada USA. (2000) 646-651.
[5]   Zhou, R.W., Quek, C. (1996): A Pseudo Outer-Product Based Fuzzy Neural Network. Proc. IEEE Int. Conf. on Neural Networks. 2 (1996) 957-962.
[6]   Tung, W.L., Quek. H.C.: GenSoFNN: A Generic Self-organising Fuzzy Neural Network. IEEE Trans. on Neural Networks. 5 (2002) 1075-1086.
[7]   Tung W.L., Quek H.C. (2000): A Novel Approach to the Derivation of Fuzzy Membership Functions using the Falcon-MART Architecture. Pattern Recognition Letters. 22 (2000) 941-958.
[8]   Zadeh, L.A.: Calculus of Fuzzy Restrictions. Fuzzy Sets and Their Applications to Cognitive & Decision Processes. Ed. Academic, New York (1975) 1-39.
[9]   Mamdani, E.H.: Application of Fuzzy Logic to Approximate Reasoning Using Linguistic Systems. IEEE Trans. on Computer, C-26 (1977) 1182-1191.
[10] Lin, C.J, Lin, C.T.: An ART-based Fuzzy Adaptive Learning Control Network. IEEE Trans. on Fuzzy Systems. 4 (1997) 477-496.
[11] Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning Internal Representations by Error Propagation. Ed. Rumelhart, D.E, McClelland, J.L. Parallel Distributed Processing, 1, Chap 8, Cambridge, MA:MIT Press (1986).

Improving Fingerprint Recognition
Based on Crease Detection

Jie Zhou[1], Chenyu Wu[1], Zhaoqi Bian[1], and David Zhang[2]

[1] Department of Automation
Tsinghua University, Beijing 100084, China
`jzhou@tsinghua.edu.cn`
[2] Department of Computing
The Hong Kong Polytechnic University, Kowloon, Hong Kong
`csdzhang@comp.polyu.edu.hk`

Abstract. Conventional algorithms for fingerprint recognition are mainly based on minutiae information. But it is difficult to extract minutiae accurately and robustly for elder people. One of main reasons is that there are too many creases on the fingertips of elder people. In this paper, we will propose a novel algorithm to improve fingerprint recognition based on crease detection. First, creases are extracted by using some special filters. Then the minutiae detected by using conventional algorithms can be further processed and those on or near the creases are discarded as false minutiae. The experimental results show that the performance can be improved by applying crease detection to discard the false minutiae. The false rejection rate can be reduced 6% on average for the fingerprints with creases.

1 Introduction

With an increasing emphasis on the emerging automatic personal identification applications, fingerprint-based identification is receiving a lot of attention [1,2]. Most classical fingerprint recognition algorithms [3,4] take the minutiae and the singular points, including their coordinates and direction, as the distinctive features to represent the fingerprint in the matching process. Minutiae extraction mainly includes the below steps: orientation field estimation, ridge extraction or enhancement, ridge thinning and minutiae extraction. Then the minutiae feature is compared with the minutiae template; if the matching score exceed a predefined threshold, these two fingerprints can be regarded as belonging to a same finger. But many research reported that the performance of the minutiae-based algorithms degraded heavily in elder people's applications.

There are always many creases existing in the fingerprints of elder people (which occupies nearly 20 percent of the population), so it is important to study this kind of pattern for fingerprints recognition. For the conventional fingerprint identification algorithms, crease is one of main reasons for false minutiae extraction. If we can detect the crease, it is possible for us to eliminate some false minutiae. Consequently, the recognition result will be improved. Unfortunately, there are no other works

aiming on this topic except for our earlier work [5], in which we proposed an algorithm to detect creases in fingerprint images.

In this paper, we will study the topic of fingerprint recognition based on crease detection. First, creases are extracted by using some special filters. Then the minutiae detected by using conventional algorithms can be further processed and those on or near the creases are discarded as false minutiae. The experimental results show that the performance of conventional fingerprint recognition algorithm can be improved by applying crease detection to discard the false minutiae.

2   Crease Detection

As we know, the fingerprint texture is composed of ridges and valleys, which can be described by a physical model [6]. Along each ridge (with high gray level in the fingerprint images) or valley, gray value varies little and smoothly. On its orthogonal direction, there is a prominent periodical variation in gray level through the ridges and valleys. The direction of the ridge or valley on each point (also called the texture's direction) constitutes the orientation field of the whole fingerprint, which can be computed by using the algorithm proposed in [7].

Different with the fingerprint texture, a crease can be seemed as an ensemble of stripe patterns in the fingerprint. Each stripe pattern appears with a similar gray level to the neighboring valleys, but its direction has a large difference compared with the texture's direction of its neighborhood. In [5], we developed a mathematical model for such stripes. In order to extract the crease from the fingerprint texture, we designed a set of 2-D oriented filters [5]. Using the defined filters, we can detect the crease from the fingerprint. First, we get the *mask* image for the input fingerprint [4] and only the valid regions need to be further processed. 12 filters are utilized. Then, Principal Component Analysis is used to get the principal axes for each connected candidate region: The axis with the larger eigenvalue represents the orientation of the crease, the eigenvalue is used to compute the length of crease, and we can also get the width of the crease using the smaller eigenvalue. After removing short creases, we combine all detected results into one image (see [5] for more details of crease detection). Although the proposed algorithm of crease detection still lacks robustness in some sense, it is enough to be used for the improvement to conventional fingerprint recognition systems. In this paper, we will mainly address on the applications of crease detection.

3   Improving Fingerprint Recognition Based on Crease Detection

In conventional minutiae-based fingerprint recognition algorithm, the influence of crease will lie in two aspects. (1) Its existence will disturb the computation of orientation field because on the points near the crease, the orientation of texture and crease may be easily misclassified. (2) In the step of ridge extraction or enhancement, false ridge endings will easily be produced due to the creases' existence, especially for the wide creases. Since ridge enhancement will be done according to the computed orientation filed, false ridge bifurcations will also be produced when wrong orientation of the fingerprint is computed. So, if we apply the crease detection into

the conventional algorithm to overcome these kinds of false ridge endings and bifurcations, the performance will be improved.

The flowchart of our algorithm is depicted in Fig. 1.

To judge if the minutiae are false or true, we establish a simple criterion: For each minutia, we compute the distance between it and each detected crease. If the distance is smaller than a predefined threshold, we can declare that this minutia is most possible to be produced by the influence of creases. Then, it can be discarded as false minutia. Denote $p_i(x_i, y_i)$, $i = 1,2,\cdots,m$, as all minutiae detected by conventional algorithm, where m is the number of all minutiae. Also denote M as the set of false minutiae produced by crease, $N(p_i)$ as a neighborhood of $p_i(x_i, y_i)$, and C as the combined set of all creases. Then

$$p_i(x_i, y_i) \in M, \quad if\ \exists (x_t, y_t) \in \{N(p_i) \cap C\}.\tag{1}$$

4   Experimental Results

The experiments are carried on THU fingerprint database, which is quite similar with the databases used in [4]. It contains a total 1760 fingerprint images, i.e. 8 fingerprints (sized 512×320) per finger from 220 individuals, which were captured with live-scanners manufactured by Digital Persona Inc.. There were no restrictions of the impression's position and direction when these fingerprints were captured. These fingerprint images varied with different qualities. About 15% of them have more than one crease.

A conventional fingerprint recognition system and our improved system are compared on these fingerprint images, both of which consist of two parts, i.e., processing part and matching part. The processing part mainly includes the steps of effective region segmentation, orientation field estimation, ridge enhancement, ridge thinning, and minutiae extraction. In our improved algorithm, a step of false minutiae removal based on crease detection is added after minutiae extraction. The important steps are briefly described as below:
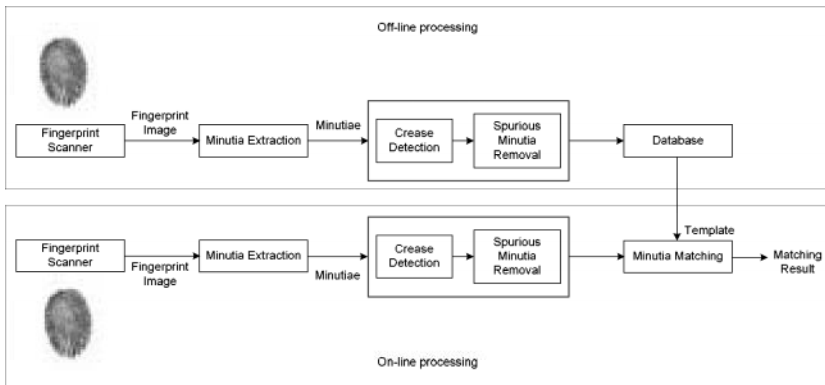
Fig. 1. The flow chart of the improved fingerprint recognition algorithm

1. Effective Region Segmentation: The fingerprint image is divided into many blocks (whose size is 16×16 pixels). For each block, the variance of the gray levels is computed. If the value exceeds the predefined threshold, this block is regarded as an effective block. Combining all effective blocks together, some post-process steps are taken, such as dilation and erosion in mathematical morphologic.

2. Ridge Enhancement: The method is quite similar with that used in [4]. Since the gray-level values on ridges attain their local maxima along a direction normal to the local ridge orientation. Pixels can be identified to be ridge pixels based on this property. The fingerprint image is convolved with a matched Gabor-like filter, which is capable of adaptively accentuating the local maximum gray-level values along a direction normal to the local ridge orientation. Then a threshold can be easily chosen to segment the ridges adaptively.

3. Minutiae Extraction: Operating on the thinned image, the minutiae are straightforward to detect: endings of the ridges are found at the termination of the lines and bifurcations are found at the junctions of three lines. But there will always be extraneous minutiae found due to a noisy image or artifacts introduced during the filtering and thinning. So some post-processing steps are needed to reduce the extraneous features: a bifurcation having a branch shorter than a defined threshold is eliminated; two endings of a short line are eliminated; two endings that are closely opposing each other are eliminated; endings at the boundary of the effective region are also eliminated. Then the following parameters are recorded for each surviving minutiae: (1) x-coordinate, (2) y-coordinate, and (3) local ridge orientation.

4. Minutiae Matching: First a Hough transform is used to convert point pattern matching to a problem of detecting the highest peaks in the Hough space of transformation parameters. It accumulates evidence in the discretized space of the transformation parameter by deriving transformation parameters that relate each two points from the input minutiae and the template minutiae. The transformation parameters corresponding to the highest peak will be used for the registration. Then translate and rotate the input minutiae according to the registration parameters. By comparing these two sets of minutiae, a matching can be achieved by placing a bounding box around each point in the template minutiae and finding a point from the input minutiae in the box. The error between the matched pair is defined as the weighted sum of their distance and orientation's difference. Then, the matching score between the input fingerprint and the template fingerprint is computed by using the number of the matched pairs minus the averaged error of all matched pairs.

As to the two systems, their difference is if false minutiae removal based on crease detection is added or not. So, the performances of false minutiae removal based on crease detection can be fairly compared by the final recognition results of these two systems.

First, each fingerprint in the database is matched with the other fingerprints. A matching is labeled correct if these two fingerprints are of the same individual. The false rejection rate (FRR) and false acceptance rate (FAR) of two algorithms (i.e. a

conventional algorithm and our improved algorithm) with different threshold values are recorded and their receiver operating curves (ROC) plotting FAR versus FRR are given in Fig. 2. The false rejection rate is defined as the percentage of true matches (i.e. two fingerprints belonging to a same finger) with the matching score less than the threshold value. The false acceptance rate is defined as the percentage of wrong matches (i.e. two fingerprints belonging to different fingers) with the matching score more than the threshold. From Fig. 2, we can see that FRR can be reduced 1% on average by using our improved algorithm. Considering that the improvement mainly occurs on the fingerprints with creases, we also compare these two algorithms on those "creased" fingerprints, which occupy about 15% of the fingerprint database. The results show that FRR can be reduced about 6% on average by using our improved algorithm on these fingerprints with creases while preserving a similar FAR.

In the experiments, we find that most false minutiae caused by creases in the tested fingerprint images can be successfully removed with our algorithm. An example of two fingerprints belonging to a same subject is provided in Fig. 3, in which the minutiae detected by the conventional algorithm are marked in red and green and the green ones denote the false minutiae removed by using our improved algorithm. From them, we can see that the false minutiae of these two fingerprints are quite different although the creases are rather robust. That means these minutiae caused by creases may produce an additional matching error between these two fingerprints and discarding them may bring us a better recognition performance.

5   Conclusions

It is difficult for conventional minutiae-based fingerprint recognition algorithms to extract minutiae accurately and robustly from fingerprints with many creases. In this paper, we proposed a novel algorithm to improve fingerprint recognition based on crease detection. First, creases are extracted by using some specially defined filters. Then the minutiae detected by using conventional algorithms can be further processed and those on or near the creases are discarded as false minutiae. The experimental results show that the performance can be improved by applying crease detection to discard the false minutiae, especially for the fingerprints with creases.
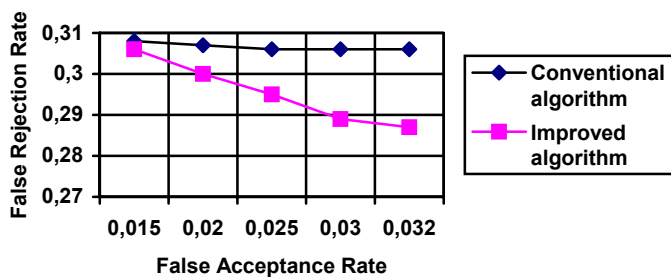
Fig. 2. A comparison of ROC curves between a conventional fingerprint recognition algorithm and our improved algorithm on THU database. Only considering the fingerprints with creases, the FRR can be reduced 6% on average by using the improved algorithm
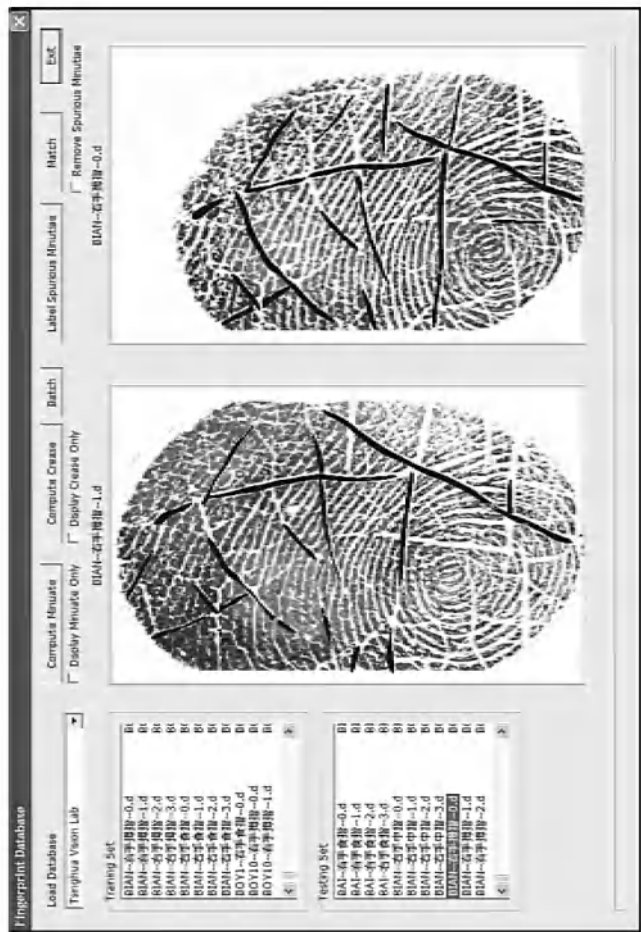
Fig. 3. An example of false minutiae removal using our algorithm. In these two fingerprint images, the false minutiae marked with green can be discarded

In our future works, we will further improve the performance of crease detection algorithm. We also want to study the possibility to apply the crease information to the fingerprint matching stage.

Acknowledgement

References

[1]   Jain A., Bolle R., Pankanti S. (Eds.), BIOMETRICS: Personal Identification in Networked Society, Kluwer, New York, 1999.

[2]   D. Zhang, Automated biometrics: Technologies and systems, Kluwer Academic Publisher, USA, 2000.

[3]   Jain A. and Hong L., "On-line fingerprint verification", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.19 (4), 302-314, 1997.

[4]   Jain A., Hong L., Pankanti S. and Bolle R., "An Identity Authentication System Using Fingerprints", Proceedings of the IEEE, vol.85(9), 1365-1388, 1997.

[5]   Wu C., Zhou J., Bian Z. and Rong G., "Robust Crease Detection in Fingerprint Images", Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol.2, 505-510, 2003.

[6]   Acton S. T., Mukherjee D. P., Havlicek J. P. and Bovik A. C., "Fingerprint Classification Using an AM-FM Model", IEEE Transactions on Image Processing, vol.10(6), 951-954, 2001.

[7]   Zhou J. and Gu J., "A Model-based Method for the Computation of Fingerprints' Orientation Field", IEEE Trans. On Image Processing, Vol.12(3), 2004.

Image-Based Approach
to Fingerprint Acceptability Assessment

Kaoru Uchida

Media and Information Research Laboratories
NEC Corporation, Kawasaki, 211-8666, Japan
`k-uchida@bc.jp.nec.com`

Abstract. Reliable acceptability assessment of an image acquired by a fingerprint scanner is one of the major requirements in securing fingerprint-based authentication systems. As a realistic solution to this, we propose and discuss a quality assessment algorithm, which, without any additional sensing hardware, extracts and analyzes features observed in a single input image. The image acceptability judgment is made as a result of discriminant function analysis of the features obtained from (1) a spatial changing pattern of gray level, which reflects the difference in the characteristics of the substance that constitutes the object, and (2) the frequency pattern of the image, which shows the existence and density of "micro-structures" on the surface of the object, such as sweat pores. Experiments show that this software-based approach is quite effective in correctly assessing the attributes of the object on the scanner, while it is platform independent and sufficiently fast.

1 Introduction

A fingerprint image acquisition subsystem, which senses ridge patterns on the surface of a finger and makes a digital image of it, plays an important role in fingerprint-based personal authentication systems [1][2][3]. Development of various new types of reliable yet relatively inexpensive scanners with a good imaging capability has been reported, among which is an optical type sensor based on in-finger light dispersion sensing [4]. This new approach, incorporating the advantages of both prism-type optical sensors using FTIR (Frustrated Total Internal Reflection) and capacitive ones, makes good images even of dry or sweaty fingers, which would have been difficult with conventional scanners.

On the other hand, we must keep in mind that improved imaging capability inevitably leads to a possibility that these *more sensitive* scanners would make good images of, and thus accept, objects that have different attributes from those natural, live fingers. These *unacceptable* images include anything that are not live, real fingerprints, which ought to be regarded as "false fingerprints," for example, imprints by artificial, fake fingers and *latent* images (i.e. ones that remain on the scanner surface because of the skin oil residue from the previously applied finger). As for artificial fingers, some studies have found that *gummy*

replicas made of inexpensive and readily available gelatin are accepted by various commercially available fingerprint authentication devices [5].

Therefore, while an image acquisition subsystem should be able to accept a wide range of real fingerprint images, it should at the same time be equipped with an object attribute assessment mechanism to reliably detect unacceptable images, since this is the key technology to prohibit impostors' fraudulent attempts of presenting non-genuine finger data to obtain false authentication.

The need for mechanism to avoid erroneous acceptance has long been recognized among the vendors and researchers in the field, and various approaches have been proposed, mainly in patent literature. Some of these approaches are based on *sensor fusion* techniques, in which the scanner is equipped with additional sensors to observe features other than the image of the applied object. These include, for example, measuring object's temperature or resistance, or installing a camera to observe the impression process. Other approaches include *active sensing* techniques, such as those that analyze the reaction of the object in response to some stimulus, or observe the dynamic change of colors or pressures during the impression.

Most of these approaches incur additional cost for special sensors. Besides, live finger detection mechanisms slow the identification process, making the use of the system less convenient (for example, while decision using sweating patterns [6] does not need additional hardware, its dynamic observation of perspiration demands the user to keep still for 5 seconds). Thus, these approaches have not been widely adopted in real systems in the market, especially in low-end products.

In this paper, we propose a new method, "*FACA*" (Fingerprint image ACceptability Assessment), for determining acceptability of an input image, which does not need any special hardware, but uses only a single acquired image.

2   Characteristic Differences Observed in Images

Inspections of captured and processed images lead us to have observations about characteristic differences that can be used in the assessment. This research uses an optical scanner based on in-finger light dispersion sensing [4] that captures images of 720×480 pixels with 800 dpi resolution.

2.1   Characteristics in Gray-Level Profile Patterns

To observe the characteristics of the presented object, the spatial transition of pixel values along a virtual line (an "*observation line*") is calculated, which results from the repetitive and alternating occurrence of ridges and valleys. From this transition pattern, which we would call a "*profile*," multiple features are extracted that represent the profile pattern.

We can assume that the characteristics and the difference observed in the profile patterns, which we use in discrimination of acceptable or unacceptable images, are caused by the way incident light is transmitted through an object on
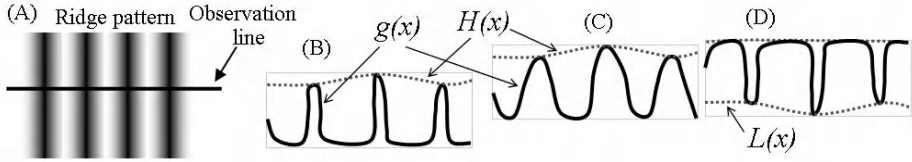
Fig. 1. Illustration of a typical fingerprint pattern, (A); and profile patterns with envelops, (B), (C), and (D). Profile lines $g(x)$ are shown as *solid*, and the upper and lower envelopes ($H(x)$ and $L(x)$) are shown as dotted lines

the scanner and is dispersed from its surface. The degree of the light's penetration and attenuation in the object reflects the difference in the characteristics of the substance that constitutes the object, namely, whether it is a real, live finger or something else.

To quantify these characteristics, we define a set of "*profile envelopes*" in the profile. The *upper* or *lower* profile envelope is the curve that connects the peaks or bottoms of the profile line. In Fig. 1, (A) illustrates a modeled ridge pattern in a fingerprint image, with an observation line on it; figures (B), (C) and (D) illustrate examples of profile lines (as solid lines, indicated as $g(x)$) and envelope lines (as *dotted lines*, $H(x)$ and $L(x)$) over and below the profile lines.

We observe, in actual samples, that the differences in the profile patterns between *acceptable* and *unacceptable* images are notable in the average level of the profile and envelope lines, and their positional relationships. For example, each illustration (B), (C) and (D) in Fig. 1 shows a typical profile pattern that exhibits the above-mentioned differences. As you can see, the locational relations between the profile line and envelop lines are significantly different.

To measure these differences, effective in this case are such features as relative size of the area between $H(x)$ and $g(x)$ and the area between $g(x)$ and $L(x)$.

2.2   Characteristics in Frequency Patterns

In typical fingerprint images, a regular ridge pattern corresponds to a strong peak, $F(u_p, v_p)$ in the frequency domain, which we call the "*principal peak.*" Figure 2 shows two pairs of an original, actual image and its frequency pattern; (A) in the left is an example of an *acceptable* image and (B) in the right is an *unacceptable* image. As we can see, there are apparent differences between the two frequency patterns, and to quantitatively evaluate these differences, we introduce some auxiliary measures.

In frequency domain $F(r, \theta)$, the average frequency strength and its variance are computed for each radius r:

$$\overline{F(r)} = \frac{1}{2\pi r} \int_0^{2\pi} F(r, \theta) d\theta \ , \qquad \sigma_F^2(r) = \frac{1}{2\pi r} \int_0^{2\pi} (F(r, \theta) - \overline{F(r)})^2 d\theta \ . \quad (1)$$
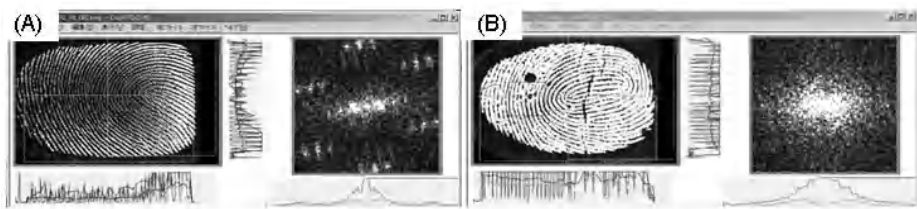
Fig. 2. Two sets of an original image and its frequency pattern; (A) in the left is an example of an *acceptable* image and (B) in the right is an *unacceptable* image
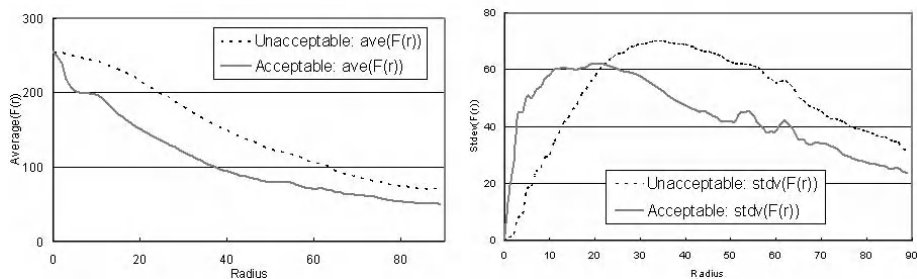
Fig. 3. Average of $\overline{F(r)}$ (left) and $\sigma_F(r)$ (right) observed in images from *acceptable* and *unacceptable* sample groups, as a function of radius r

Based on actual observations and analysis of sample patterns, we note that the differences in the frequency patterns between *acceptable* and *unacceptable* images are notable in the presence and strength of weaker, non-principal peaks in the area having larger r (distance from the center), and the shape of the $\overline{F(r)}$ and $\sigma_F(r)$ curves.

In Fig. 3, the average values of $\overline{F(r)}$ and $\sigma_F(r)$ are drawn as a function of radius r [1]. As you can see there is a considerable difference in their patterns. We can notice, for the *acceptable* group curves, for example, that there are distinctive peaks near radius 52 and 62 in the $\sigma_F(r)$ curve and there are also small rises at around 56 in the $\overline{F(r)}$ curve. We can assume that these correspond to the weak, six peak groups in the marginal area in Figure 2 (A).

The characteristics and the differences observed in the frequency patterns, which we use in discrimination, reflect the existence and density of "micro-structures" on the surface of the object, such as sweat pores, whose pitch is higher than those of ridge patterns, that are intrinsic to live and/or good fingers.

---

[1] These graphs have been obtained by computing $\overline{F(r)}$ and $\sigma_F(r)$ for 500 actual image training samples each from *acceptable* and *unacceptable* image groups.

3   FACA Algorithm and its Implementation

FACA (Fingerprint image ACceptability Assessment) proceeds in the following three stages:

1. feature extraction from gray level profiles on observation lines,
2. feature extraction from the frequency patterns in observation areas, and
3. the acceptability decision based on the extracted features.

3.1   Feature Computation for Profile-Based FACA

Along observation lines placed on the fingerprint area, computed are the upper envelope line $H(z)$ and the lower envelope line $L(z)$.

Based on the above observations, multiple *profile features* are computed including, for example:

$$X_{p_1} = \overline{g(z) - L(z)} = \frac{1}{Z} \int (g(z) - L(z))dz \ , \quad X_{p_2} = \overline{H(z) - g(z)} \ , \quad (2)$$

which are expected to be effective in differentiating the patterns (B), (C) and (D) in Fig. 1.

3.2   Feature Computation for Frequency-Based FACA

To differentiate (A) and (B) in Fig. 2 and two curves in Fig. 3, let us define a "concentric band," $B(p, q)$ as a doughnut-shape band between two circles with radiuses p and q, where $p < q$. For a concentric band $B(p, q)$, the G-value and H-value, which are the average values of $\overline{F(r)}$ and $\sigma_F(r)$ in $B(p, q)$ are computed as follows:

$$G(p, q) = \frac{1}{q - p} \int_p^q \overline{F(r)}dr \ , \qquad H(p, q) = \frac{1}{q - p} \int_p^q \sigma_F(r)dr \ . \quad (3)$$

Based on the earlier discussion, multiple *frequency features* are computed using the G-values and H-values computed on some of the concentric bands including, for example,

$$X_{f_1} = \frac{G(r_1, r_2)}{G(r_3, r_4)} \ , \quad \text{and} \quad X_{f_2} = \frac{H(r_5, r_6)}{H(r_6, r_7)} \quad (r_1, r_2, \dots : \text{constant}), \quad (4)$$

which are expected to be effective in differentiating the acceptable and unacceptable graph patterns in Fig. 3.

3.3   Acceptance Decision

Training set of 5,000 actual images, consisting of both *acceptable* and *unacceptable* images, are used to construct a discrimination model, i.e., to compute a set of coefficients in a discriminant function. By using multi-variable regression

Table 1. Summary of error rates and speed

	FFT128	FFT64	noFFT	
False reject (FR)	3.18%	3.94%	17.4%	error for 5,000 "acceptable" images
False accept (FA)	4.04%	4.38%	9.09%	error for 2,000 "unacceptable" images
Total errors	7.22%	8.32%	26.5%	sum of FR and FA
Processing time	33.4 ms	22.7 ms	12.7 ms	(for one image)
Relative time	1.00	0.68	0.38	

analysis, the discriminant coefficients are determined as results of the training phase.

In the actual assessment phase, when an image is presented, the same set of features, both from profile patterns and frequency distribution, are extracted from the image, and discriminant analysis is performed using the obtained regression coefficients, and the input image is assessed either *acceptable* or not according to the computed discriminant score.

4   Experiments and Evaluation

4.1   Conditions

We implemented the above scheme on software and conducted experiments using images collected from real fingerprints (5,000 images) and *non-real* ones (2,000: mixture of replica[2] and latent.).

In statistical analysis, we adopted 11 profile features (such as $\overline{L(z)}$, $g(z) - L(z)$, $g(z)$, $\overline{H(z)^2}$, $\overline{H(z) + L(z)}$, ... for horizontal and vertical directions) and 5 frequency features (such as $G(55, 65) - G(45, 55)$, $H(5, 15)/H(45, 55)$, $H(55, 65)/H(45, 55)$, ...) to carry out 16-dimensional discriminant analysis.

4.2   Results

Table 1 shows the measured performance in error rates, and processing time[3] of several implementations.

"FFT128" and "FFT64" are the implementations using observation area size of 128 and 64 for FFT. The latter turned to be 32% faster with accuracy degradation in error rate of 1.1%. For comparison, implementation with only the profile FACA and no frequency FACA ("noFFT") was tentatively tested also.

The observed trade-off between false accept (FA) and false reject (FR), or between security and usability, is shown as an ROC curve in Fig. 4.

---

[2] Some were produced as is described in [5]

[3] Processing time is measured on a personal computer with 800 MHz Pentium III.
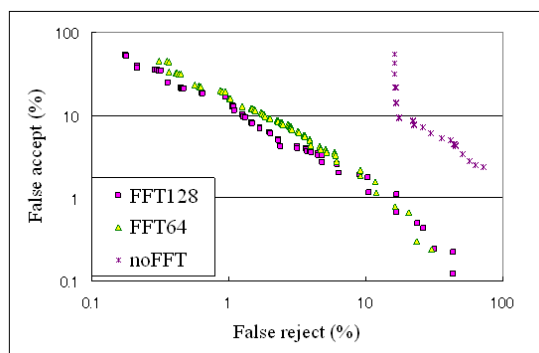
Fig. 4. False reject versus false accept for various FACA implementations

5   Conclusions

The defense technology against impostors' fraudulent attempts to present non-genuine data, for example artificial fingerprints, is all the more important as the scanner's imaging capability improves, since the *more sensitive* scanner is expected to make good images of objects other than live fingers. As a realistic solution for determining if the image is "*acceptable*," we proposed and discussed a reliable quality check algorithm to deny the acceptance of a doubtful image, based on image assessment by statistical analysis of features obtained from a single image.

The proposed method uses a part of area (or lines) for observation of features that is expected to well represent the characteristics of the whole image. This make the approach well-suited for applications on low-end platforms with limited computational resources.

References

[1] Special Issue on Automated Biometrics, Proceedings of the IEEE, vol. 85, September 1997. 294
[2] A. Jain, ed., Biometrics: Personal Identification in Networked Society, Kluwer Academic Publishers, 1999. 294
[3] L. O'Gorman, "Fingerprint Verification," in *Biometrics: Personal Identification in Networked Society*, pp. 43–64, Kluwer Academic Publishers, 1999. 294
[4] T. Higuchi, "Fingerprint authentication system / New fingerprint sensor," NEC Technical JournalüCVol.55, No.3, pp.19–22, March 2002. 294, 295
[5] T. Matsumoto, H. Matsumoto, K. Yamada, and S. Hoshino, "Impact of artificial *Gummy* fingers on fingerprint systems," Proceedings of SPIE, Vol. 4677, pp.275–289, 2002. 295, 299
[6] R. Derakhshani, S. Schuckers, L. Hornak, and L. O'Gorman, "Determination of vitality from a non-invasive biomedical measurement for use in fingerprint scanners," Pattern Recognition, Vol.36, No.2, pp.383–396, 2003. 295

Reducing Silicon Fingerprint Sensor Area

Jean-François Mainguet[1], Wei Gong[2], and Anne Wang[2]

[1] Atmel
Avenue de Rochepleine, BP 123, 38521 Saint-Egrève Cedex, France
jean-françois.mainguet@gfo.atmel.com
[2] Cogent Systems, Inc.
209 Fair Oaks Avenue, South Pasadena, California 91030, USA
{weigong,annewang}@cogentsystems.com

Abstract. Silicon fingerprint sensor cost is proportional to the area of silicon. Sweep sensors have been proposed to replace square sensors in order to reduce the cost, but are still quite large. This paper describes a simulation to determine what could be a reasonable minimum width of a sweep sensor. A database acquired with the FingerChip® is used as a reference, and FAR/FRR curves are computed using cropped images against full and reduced images.

1 Introduction

Silicon fingerprint sensors are now commonly offered on the market for authentication/identification purposes. They are replacing optical fingerprint sensors, mainly to reduce the cost and the size as they are flat. Most of the silicon fingerprint sensors are "touch" sensors: you just need to press your finger on the sensor surface, exactly like optical sensors. As a result, the sensor area has the same range as a finger, a few cm^2. Unfortunately, the cost of the silicon is directly proportional to the area, and such an area is big for silicon manufacturers.

To reduce the cost, sweep sensors have been proposed [1]. The main advantage is a much smaller silicon area, which reduces the cost, but also the footprint that is also an important feature for portable devices. At the moment, the width of such sensors is roughly the width of a finger, about 1.0 to 1.5cm.

Is it possible to reduce this width, to reduce again the cost? This is the topic of the present paper. After presenting the current sensor sizes offered in the industry today, a method to simulate a smaller sensor will be presented, FAR/FRR curves will be computed and results will be discussed.

2 Area of Silicon Fingerprint Sensors

Table 1 lists most of the silicon fingerprint sensors and some others. Only the sensing area is taken into consideration. Fig. 1 shows the sensors, width for X-axis, height for Y-axis. All the points above the diagonal are rectangular sensor, on the diagonal, you find the exactly square sensors, and below the diagonal, the sweep sensors.

It is obvious that the larger area sensor will produce a larger fingerprint image, and so will contain more information for matching. This will enhance the recognition rate

up to a certain point, depending on the finger size, acquisition accuracy and authentication software performance. All these sensors are generally using different physical effects to capture the fingerprint, and different authentication software, but generally based on minutiae extraction. It is interesting to note most of sensors have a width comprised between 9.8mm and 15mm, with a peak near 10 to 12mm, showing the cost pressure to reduce the area of silicon. The largest one is not a silicon sensor, and so has less incentive to reduce its size.

Table 1. List of fingerprint sensors / sensing area

Company	Part number	Type	Res. dpi	x pixels	y pixels	x mm	y mm
Atmel	AT77C101B	thermal	508	280	8	14.0	0.4
STM	TCS1AD	capacitive	508	256	360	12.8	18.0
	TCS2AF	capacitive	508	208	288	10.4	14.4
	TCS3A	capacitive	508	256	4	12.8	0.2
Authentec	AF-S2	modulation	250	128	128	13.0	13.0
	AES4000	modulation	250	96	96	9.8	9.8
	AES3500	modulation	500	128	128	6.5	6.5
	AES2500	modulation	500	192	16	9.8	0.8
Infineon	FTF 1100	capacitive	513	224	288	11.1	14.3
Fujitsu	MBF110	capacitive	500	300	300	15.0	15.0
	MBF200	capacitive	500	256	300	12.8	15.0
	MBF300	capacitive	500	256	32	12.8	1.7
Sony	CXA3271AGE	capacitive	317	128	192	10.3	15.4
	FIU900	capacitive	317	128	128	10.2	10.2
Fingerprint Cards	FPC1010	capacitive	363	152	200	10.6	14.0
	FPC1030	capacitive	363	152	32	10.6	2.2
Ethentica	T-FPM	polymer	403	306	225	14.2	19.3
NTT [2]		capacitive	311	124	166	10.1	13.5
BMF	BLP-100	pressure	406	256	384	16.0	23.4
Fidelica	FIS-3001	pressure	508	256	256	12.8	12.8
KAIST		capacitive	508	210	100	10.5	5.0
Seoul university[3]		capacitive	600	256	64	10.8	2.7
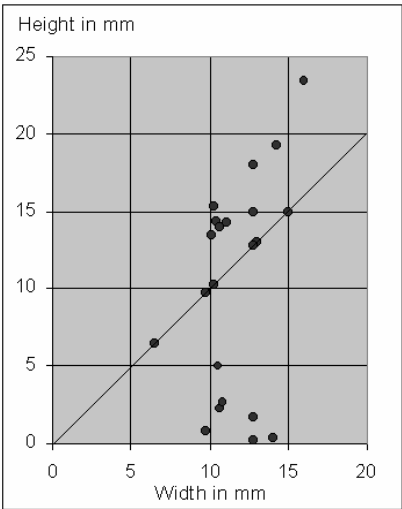
Fig. 1. Size of fingerprint sensors

One sensor is well below the 10mm width, with only 6.5mm. This is obvious for reducing the cost, and the recognition accuracy is certainly an issue. For this sensor, it is recommended to acquire several images, while moving the finger, to obtain a patch of small images.

No sweep sensor is below 9.8mm width. Is it possible to reduce this width without degrading the recognition rates, taking advantage that more data will be collected in the vertical direction? It is less expensive to simulate a lower width by cropping images in a database than manufacturing a new sensor and perform tests: this is the method proposed in the present paper.

3   Proposed Method

A database has been collected, using the FingerChip® from Atmel which is 280 pixels (14mm) wide, 508dpi. With this database, False Acceptance Rate (FAR) vs. False Rejection Rate (FRR) curves are computed, using the method defined for the FVC2002 competition [4] excepted for FRR: only comparisons between the current image and the first of each finger is taken into account, because later, we need to compare a cropped and shifted image with the original image, acting as the "enrolment" image. The FVC2002 competition is using all possible combination of images of the same finger, except for symmetrical pairs.

The database contains 4029 images from 410 fingers of 228 persons leading 3619 FRR pairs for each test. Not all FAR pairs have been used, only a random subset (always the same) because of too large number of tests to perform: 326,223 FAR pairs (instead of 1,564,016), but enough to get an accurate FAR/FRR curve for FAR $\sim 10^{-4}$.

The database is a challenging database: voluntary changing conditions (such as different acquisition device, temperature of the device) have been used in order to produce all possible kind of images. In the real world, this will not happen as the device will be used in the same configuration for the same user (same ergonomics, same host processor…). Moreover, only USB has been used, and some artifacts happen only in that configuration, such as stop and go in images that happens from time to time because of the operating system which stops the acquisition if a more important task occurs (PCs are not true real time systems).

As a result, the FRR is not as good as observed in a real world system, but it is necessary to have such a challenging database in order to increase the sensitivity to parameter variations, and to obtain clear results.

Only minutiae (ending points and bifurcations) are used to describe a fingerprint. Each minutia features a triplet of its x, y coordinates and its orientation. This means only localized information on a fingerprint is required, which is interesting for our purpose of reducing the sensor size. Comparison of fingerprints is based on minutia only. The comparison has no restriction on underlying relative translation and rotation between matched fingerprints.

To simulate smaller device, we need to study two parameters:

1. Simulate a sweep sensor that has less pixels (width).
2. Simulate a shift, as the user may not sweep his/her finger exactly at the same place.

Studied widths are between 200 and 60 pixels (i.e. between 10 and 3mm). Above 200 pixels, there are no significant losses, which confirms the fact that most of the sensors are above 10mm width. Studied shifts are between -100 and 100 pixels (i.e. between -5 and 5mm). Obviously, when the width is smaller than the shift, there is no overlapping of data, and FRR must fall to 100%. Note that fingerprint height is variable and limited by the reconstruction software to 500 pixels (25mm) in maximum. The mean value of fingerprint height is 400 pixels (20mm).

When simulating the enrollment, two main cases may happen:

1. The enrollment and recognition are done using the same device. To simulate for this, comparison is done between the first image, which is cropped but never shifted, and the challenging image, which is cropped and shifted.
2. The enrollment is done using a large sensor, and recognition with a small sensor, which may be the case if an enrollment station exists, or if a special enrollment method requires several shifted sweeps to reconstruct a large image.

These two cases are evaluated in this study.

We used a step of 20 pixels for width and shift, leading to 177 FAR/FRR curves to compute. The total number of FRR pairs is >600.000 while the total number of FAR pairs is >57millions. Fortunately, the matching algorithm we used is one of the fastest and gives reasonable computing time on a 2 GHz PC.

4   Cropping and Shifting Fingerprints

Special care has been taken to generate images as close as possible to a real sweep with a smaller device. The fingerprint images are not simply cut on each side, because users are not forced to sweep vertically. Fingerprints are reduced, using the particular fact that the section that contains information has the width of the sensor, here 280 pixels.

Fig. 2 shows an original image. The sweep is not vertical, and the sensor edges are clearly visible. A software routine exactly removes the right amount of pixels on each edge, depending on the desired width and shift.

Fig. 3 shows a smaller width, yet still centered (no shift). This is the image that will be kept as the "enrollment image" for future comparison. Fig. 4 & 5 shows the same width, but with an additional shift. This simulates a shifted sweep.

Fig. 2., 3., 4. & 5. Original image / no shift 100 width pixels, / +80 pixels , -80 pixels shift

5   Performance Measurement

When the width is reduced, without shift, performances are degraded: the FRR increases a lot, because recognition is becoming more and more difficult. Fig. 6 shows the degradation: the FAR/FRR curves are shifting to the right, toward high FRR values.

As the curves are slightly parallel, we will use the FRR value for a FAR equals to 10^{-4}, normalized to the reference FRR value when the full fingerprint is used. We will always speak now of a percentage of the original performance.

We consider that 75% of the original performance is the limit to get an acceptable result. This value has been empirically defined using several different fingerprint programs over the database with known performances.

6   Results

Fig. 6 shows the degradation of the FAR/FRR, from 100% of the original performance when using full images. Only cropped images are used, without computed shift.

There is almost no significant loss up to 200-pixel width, this is why no computation has been performed between 280 and 200 pixels (10mm). It is interesting to note that practically all sensor manufacturers are making devices larger than this.

Fig. 7 displays the same information, but normalized to the original performance.

Two curves are displayed: the dotted curve corresponds to the matching with the full original fingerprint, as if the enrollment was done with a large sensor: this is the best result, which is normal if considering the fact that the reference image contains the maximum data.
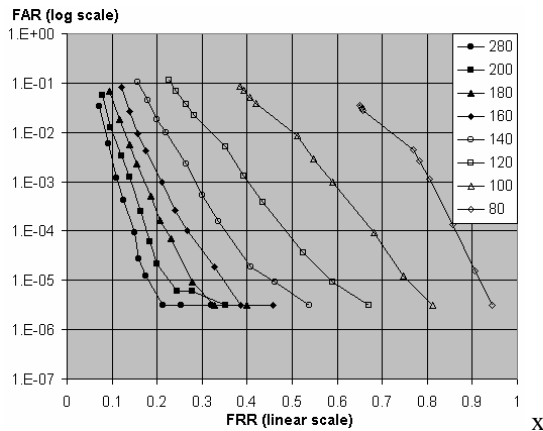
Fig. 6. FAR/FRR curves using semilog diagram. FAR uses log scale whereas FRR uses linear scale, giving almost linear curves. The original FRR value is quite high because of the database which is voluntary challenging. The curve shape is a good clue to indicate if there are enough samples; here, for FAR=10^{-4}, there are about 32 false accepted pairs, and below 10^{-5}, results are less reliable. Below 80 pixels, the recognition rate is too poor and the curves are stuck to the edge (not displayed)
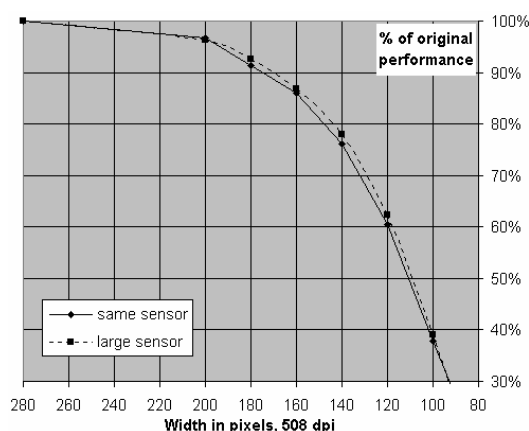
Fig. 7. Cropped but non-shifted images. Normalized FRR to the original FRR @ FAR=10^{-4}. 75% of the original performance is considered as a limit for acceptable results. The two curves are similar, showing that very few shifts exist in the database: people are sweeping always the same, without stringent finger guide

The plain curve shows the matching with a cropped image, as if the enrollment was done with the same small sensor (data identical to Fig. 6). It is interesting to note that there is only a small difference between the two curves, especially with small widths: we may expect that any shift during the acquisition will make a larger difference, because there is less data overlapping, and that below a certain width, the two curves will separate more than that.

When the fingerprints were acquired, no special incentive was done to center the finger on the sensor during the sweep, and a small and not stringent guide was existing. Even if the restriction was set to this low level, from this result, we may still conclude that most fingerprints are centered in the database. This is beyond our expectation.

Similar curves are computed using a fixed shift, compared to the full original image and the cropped image without shift. This is resulting in a grid of values, displayed in fig. 9 (full original image) and fig. 11 (cropped unshifted image).

The no shift curve in the center is the same as in fig. 7. The surface is symmetrical, which shows that there is no significant difference existing between left or right side shift. This was an expected result, and a kind of check that the cropping routine is correct.

It is extremely interesting to note that the performance result is directly linked to the overlap of images. Fig. 8 & 10 display the overlap with the reference image (original image & cropped but unshifted image). It is clearly visible that the performance is directly correlated to the overlap. Fig. 9 shows that in the center, there is always the necessary data overlap as the full image is used as reference.

To get 75% of the original performance, we need to have about 140 pixels (7mm) overlapping between two images, with a mean height of 400 pixels (20mm).
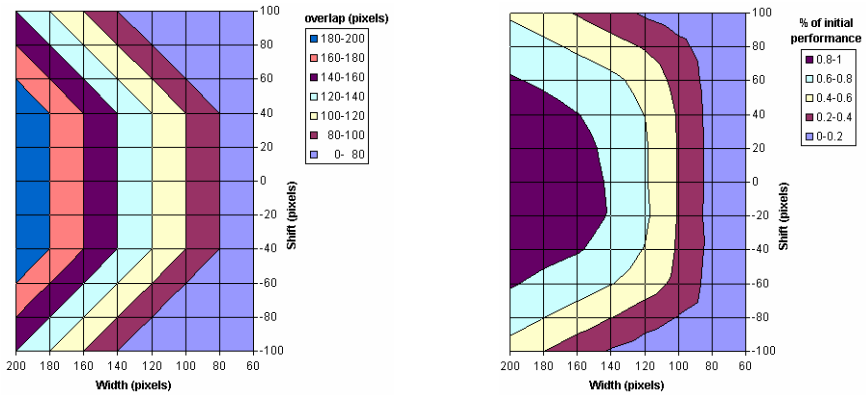
Fig. 8. & 9. Overlap with original image. Normalized performance depending on width and shift, compared to the original image (enrollment with a large sensor). Notice the correlation between performance and overlap
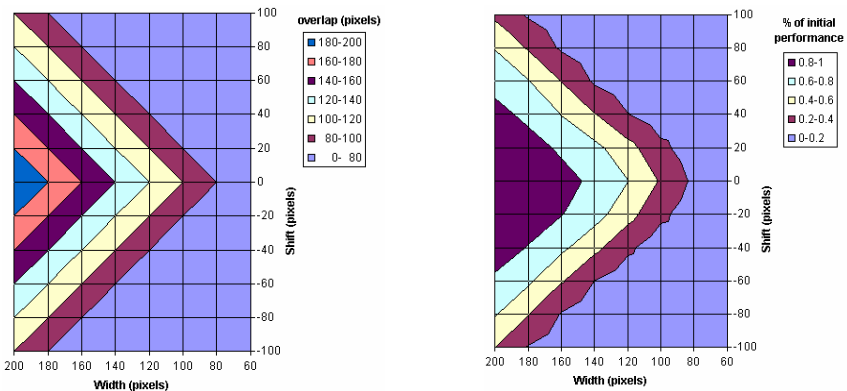
Fig. 10. & 11. Overlap with cropped unshifted image. Same as fig. 8 & 9, but compared to the cropped unshifted image (enrollment with the same sensor)

7   Conclusion and Remarks

7mm is the minimum width for a sweep type fingerprint sensor to perform recognition with acceptable accuracy using a state-of-the-art minutiae-based authentication software. To take advantage of this, it is mandatory to have a special enrollment procedure to take the full fingerprint image to compensate for shift, or use a finger guide to force the user to always show the same finger area.

To get a full fingerprint image in enrollment, using a full size sensor to enroll finger is an obvious solution. But this configuration will not be always available in real application. As a possible alternative, an interesting idea is to use in enrollment a reduced sensor to capture different parts of a fingerprint, in other words, to capture a

fingerprint at different shifts, and then use an elaborated algorithm to reconstruct the full fingerprint image from these partial fingerprint images.

Reconstruction of full image from multiple partial images is not new in industry. For a sweep sensor, a necessary technique is to reconstruct the image from image slices captured when the finger is swept over the sensor [3]. Also, a similar approach has already been used in optical (flat) sensors to capture a rolled fingerprint image that was thought in the past to be obtainable only through scanning fingerprint card. The basic idea behind reconstruction is to find common (i.e. overlapping) areas between partial images. For above-mentioned applications, the partial images are consecutive captures of movement (sweeping or rolling) of a finger on the sensor in a very short time interval. So a very close correlation among the partial images can be expected and the common areas in different images can be reliably assumed to be spatially consistent.

To reconstruct full image from multiple captures of a shifted finger on a reduced sensor, however, brings about new challenges. Because the partial images represent independent sweeping on the sensor, following problems are introduced: (1) Distortions of the images along the sweeping direction (in vertical) differs from image to image. Depending on the factors such as sweeping speed and pressure, the vertical dimensions of the images of the common areas may show significant difference. (2) Since user's finger may undergo different horizontal (orthogonal to sweeping direction) movement in different sweeping, the same common areas in different captures may skew in different way. Consequently, common areas between different partial images will differ in size and shape, which makes it harder to find them reliably. Our future work will be investigating into this issue and find out a usable strategy and solution.

References

[1]   J.F. Mainguet, M. Pegulu and J.B. Harris, "Fingerprint recognition based on silicon chips" Future Generation of Computer Systems 16 (2000), Elsevier, pp. 403-415.
[2]   S. Shigematsu, H. Morimura, Y.Tanabe, T. Adachi, K.Machida1, "A Single-Chip Fingerprint Sensor and Identifier" in IEEE Journal Of Solid-State Circuits, Vol. 34, No. 12, pp. 1852-1859, December 1999.
[3]   J.-W. Lee, D.-J. Min, J. Kim, W. Kim, "A 600-dpi Capacitive Fingerprint Sensor Chip and Image-Synthesis Technique" in IEEE Journal Of Solid-State Circuits, Vol. 34, No. 4, pp. 469-475 April 1999.
[4]   D. Maio, D. Maltoni, R. Cappelli, J. Wayman, and A. Jain "FVC2002: Second Fingerprint Verification Competition" in ICPR 2002
       http://bias.csr.unibo.it/fvc2002/default.asp.

Externalized Fingerprint Matching

Claude Barral[1], Jean-Sébastien Coron[2], and David Naccache[2]

[1] Gemplus Card International
34 rue Guynemer, 92447 Issy-les-Moulineaux, France
{jean-sebastien.coron,david.naccache}@gemplus.com
[2] Gemplus Card International
La Vigie. Avenue des Jujubiers, La Ciotat, F-13705, France
claude.barral@gemplus.com

Abstract. This paper describes a very inexpensive memory card (phonecard-like card) capable of performing fingerprint matching. In the proposed protocol the card stores the user's fingerprint information to which random minutiae were added at enrolment time (we denote this scrambled template by t). The card also stores a binary string w encoding which of the minutiae in t actually belong to the holder. When an identification session starts, the terminal reads t from the card and, based upon the incoming scanner data, determines which of the minutiae in t are genuine. The terminal forms a candidate w' and sends it to the card. All the card needs to do is test that the Hamming weight of $w \oplus w'$ is smaller than a security threshold d. It follows that the card only needs to embark passive data storage capabilities, one exclusive-or gate, a shift register, a counter and a comparator (less than 40 logical gates).

1 Introduction

Generally, a fingerprint biometric system comprises four main modules:

- A capture unit, which acquires the raw biometric fingerprint data D of an individual (typically a bitmap of the finger's ridges).
- A feature extraction module f in which the acquired biometric data is processed to extract a feature-set $f(D)$ that models D. Typically $f(D)$ is the position and orientation of minutiae.
- A matching module μ in which an extracted feature-set $f(A)$ can be compared to a reference pattern $f(B)$. This comparison process (figure 4) outputs a score $0 \leq \mu(f(A), f(B)) \leq 1$.
- A decision-making module in which the user's claimed identity is either accepted or rejected based on the matching score: if $\mu(f(A), f(B)) > \alpha$ return accept else return reject. α is an application-dependent security parameter.

We assume that the reader is familiar with the notions of FAR and FRR and refer the reader to [3] for a formal definition of the matching problem.

2   Fingerprint Match-On-Card: Biometric Smart Cards

Biometric smart-cards has the capacity to store a template $f(D)$ in EEPROM and perform both matching and decision-making when presented with a candidate D' (or $f(D')$ if the algorithm f is public[1]).

Typically, an accept will 'open' the card and permit access to some of its EEPROM files, enable the generation of a digital signature or debit a purse.

It is customary to require that these steps take place in less than a second (convenience). When coded in a card a matching algorithm would use at least 2,000 bytes of RAM. Code would usually occupy 2,000 to 12,000 ROM bytes.

Code complexity (matching involves many floating-point trigonometric operations) and RAM consumption are two decisive cost factors in the design of such solutions.

The following section provides a novel solution to this problem. The solution, called *Externalized Fingerprint Matching*, allows to implement μ in simple (microprocessor-less) memory cards. This is particularly important for addressing cost-effectively very large markets (*e.g.* China, 1.3 billion inhabitants) and for deploying disposable biometric IDs such as visas, hotel room keys or visitor/subcontractor badges.

3   Externalizing the Fingerprint Matching

The new idea consists in adding *false minutiae* to $f(D)$ and *reversing the burden of proof* to have the *card challenge the reader* to find out, based on the acquisition coming from the scanner, which minutiae are genuine:

3.1   Enrolment

The enrolment protocol is the following:

1. The issuer extracts $f(D)$, picks a set of random minutiae r and merges it into $f(D)$. We denote the result of this operation (illustrated in Figure 1) by $t = f(D) \cup r$.
2. The issuer encodes t as a binary string u where bit $u_i = 1$ if the i-th minutia in t belongs to $f(D)$ and $u_i = 0$ if the i-th minutia in t belongs to r.
3. The issuer signs, using a public-key signature scheme the data $\{t, u, d\}$ where d is a security parameter whose choice is discussed below. Let σ be the signature of $\{t, u, d\}$.
4. The issuer delivers an identity card containing $\{t, u, d, \sigma\}$. The card allows the free reading of t and d.

---

[1] Most match-on-card algorithms are proprietary but usually available under NDA. The following companies sell, license or develop match-on-card code: Precise biometrics, Veridicom, Gemplus, Siemens biometrics, Ikendi, Dermalog, Identix, Fingerprint cards AB.
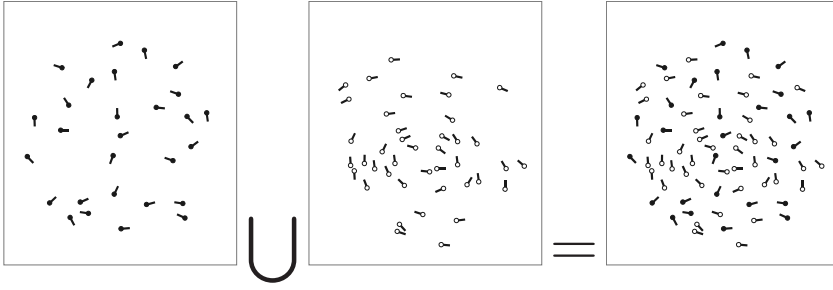
Fig. 1. Fingerprint Scrambling with False Minutiae

3.2   Identification

The identification protocol is the following:

1. The terminal receives from the scanner a fingerprint candidate D'.
2. The terminal reads t from the card and partitions t (based upon D') into two sets $t = t_{\text{true}} \cup t_{\text{false}}$. The terminal encodes this partition as a binary string u' where bit $u'_i = 1$ if the i-th minutia in t belongs to t_{true} and $u'_i = 0$ if the i-th minutia in t belongs to t_{false}. The terminal sends u' to the card.
3. The card computes $w = u \oplus u'$. If the Hamming weight of w (denoted $H(w)$) is smaller than d the card outputs σ and u. At this step the card considers that the presented finger is legitimate.
4. The terminal verifies σ with respect to the issuer's public-key and if σ is correct and $H(u \oplus u') \leq d$ then the terminal considers that the scanned finger and the card match each other and are approved by the issuer.

3.3   Evaluating the Protocol's FAR

The security of the above protocol is determined as follows.

The correct fingerprint is characterized by the reference vector u whose length is $n + m$ and whose Hamming weight is n. Since u is unknown to the attacker we assume that it has a random distribution over the vectors of weight n.

Assume that the Hamming weight of u', the vector submitted by the attacker, is equal to $n + k$, where k is a non-negative integer. Letting $w = u' \oplus u$ we have $w = w_1 \vee w_2$ where $w_1 = u \wedge \neg u'$ and $w_2 = \neg u \wedge u'$. Let $i = H(w_1)$.

We have $H(u') = n+k = H(u)+H(w_2)-H(w_1)$, which gives $H(w_2) = i+k$, whereby $H(w) = H(w_1) + H(w_2) = 2i + k$. Since $H(u') = n + k$, the number of possible choices for w_2 is $\binom{n+k}{i+k}$ and the number of possible choices for w_1 is $\binom{m-k}{i}$.

The number of possible u vectors for a given integer i is therefore:

$$R(n, m, k, i) = \binom{n + k}{i + k} \times \binom{m - k}{i}$$
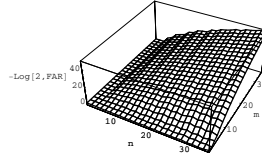
Fig. 2. FAR for $d = 4$

Summing over all possible i, we obtain the probability over u, which we denote $P(n, m, k)$, that the attack succeeds with a candidate u' of weight $n + k$:

$$P(n, m, k) = \frac{\sum_{i=0}^{(d-k)/2} R(n, m, k, i)}{\binom{m+n}{n}}$$

If k is negative, we obtain the probability:

$$P(n, m, k) = \frac{\sum_{i=0}^{(d+k)/2} R'(n, m, k, i)}{\binom{m+n}{n}}$$

where

$$R'(n, m, k) = \binom{n+k}{i} \times \binom{m-k}{i-k}$$

Eventually, the FAR is the maximum probability, over all possible k, that a candidate u' is accepted:

$$\text{FAR} = \max_{k=-n}^{m} P(n, m, k)$$

Letting FAR $= 10^{-e}$, typical $\{n, m\}$ values for $e = 5$ and $d = 0$ would be: $\{6, 17\}, \{7, 14\}, \{8, 12\}, \{9, 11\}, \{10, 10\}, \{11, 9\}$. Variations in d affect the FAR as shown in the graphics below (Figure 2 shows the FAR for $d = 4$ and Figure 3 shows the FAR for $m = n$ and different d values):
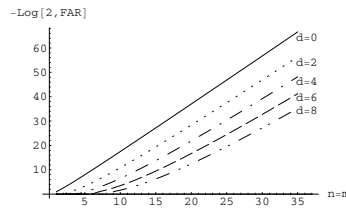
Fig. 3. FAR for $m = n$ and different d values

Table 1. Protocol phases and durations

Protocol phase	Duration
The terminal asks permission to send u'	6 ms
The card prepares to receive u'	8 ms
The terminal sends u'	6 ms
The card compares u and u'	4 ms
The card returns true or false	2 ms

Note that the above calculations rely on two assumptions that appear in the full version of the paper [1]. Moreover, an alternative fingerprint scrambling model is given in the appendix.

4 Implementation and Applications

The protocol was implemented as a Javacard applet on a Gemplus GemXpresso Pro smart card using Ikendi Software AG's minutiae extraction engine.

For the reader terminal emulation, the demonstrator uses a Pentium III at 500 MHz and a Gemplus GemPC Touch 430 smart card reader with an embedded fingerprint sensor (silicon sensor using capacitive technology), shown in figure 4.

The entire fingerprint matching process takes less than a second. The applet's size is about 510 bytes and the card's processing time is 26 ms, that break-down as shown in table 1.

The applications of our scheme are described in the full version of this paper [1].

4.1 Conclusion

The above shows that although extremely economic (from an on-board resource consumption perspective), the protocol presented in this paper provides equivalent functionalities to other match-on-card techniques in all typical use-cases.

Fig. 4. GemXpresso Pro and GemPC Touch 430

References

[1] C.Barral, J. S. Coron and D. Naccache, *Externalized Fingerprint Matching*, Cryptology ePrint Archive, Report 2004/021, year 2004, http://eprint.iacr.org. 313

[2] A. Robert, La Tribune, Les cartes à puce se cherchent une identité, page 59b, October 10, 2003.

[3] D. Maltoni, D. Maio, A. Jain, S. Prabhakar, *Handbook of Fingerprint Recognition*, Springer, New York, 2003. 309

A   Appendix: Simplified Minutiae Scrambling Model

A.1   The New Assumption

In the simplified minutiae scrambling model, we let n be the number of minutiae in $f(D)$ and $k \leq n$ be the total number of minutiae in the resulting template t, that is, true and false minutiae. Again, we let d be a security parameter whose choice is discussed below.

Definition 1. *A Biometric Scrambling Algorithm is an algorithm taking as input a set $f(D)$ of n minutiae and outputting a template t and a randomly distributed k-bit string u, such that the i-th minutia in t belongs to $f(D)$ iff $u_i = 1$.*

Simplified Assumption. There exists an (efficient) Biometric Scrambling algorithm $\mathcal{A}$ with the following two properties:
- Given $f(D)$ and a t generated by $\mathcal{A}$, one can obtain a u' such that $H(u' \oplus u) \leq d$ with probability greater than β.
- Given only t the success probability of an attacker in outputting u' such that $H(u' \oplus u) \leq d$ is $\epsilon_{\text{guess}} + \texttt{negl}$, where $\texttt{negl}$ is a negligible function of the parameters, and

$$\epsilon_{\text{guess}} = 2^{-k} \sum_{i=0}^{d} \binom{k}{i}$$

A.2   The New Protocol

The new enrolment protocol is then the following:

1. The issuer extracts $f(D)$ and sets $t \leftarrow \{\}$.
2. The issuer generates a random k-bit string $u = u_1, \ldots, u_k$.
3. For $i = 1$ to k:
   (a) if $u_i = 1$, the issuer adds to the template t a random (and not already selected) minutia from $f(D)$.
   (b) if $u_i = 0$, the issuer adds to t a random minutia.
4. The issuer signs the data $\{t, u, d\}$. Let σ be the signature of $\{t, u, d\}$.
5. The issuer delivers an identity card containing $\{t, u, d, \sigma\}$. The card allows the free reading of t and d.

   Identification is identical to 3.2.

A.3 Security

The second property of the Simplified Assumption ensures that the success probability of an attacker is only negligibly greater than the success probability obtained by just randomly "guessing" the random string u. The following theorem proves that the identification protocol is secure under the Simplified Assumption.

Theorem 1. *Without the knowledge of $f(D)$, the attacker's success probability is smaller than $\epsilon_{guess} + \texttt{negl}$.*

Proof. Assume that an attacker $\mathcal{F}$ manages to pass the identification protocol without knowing $f(D)$, with probability ϵ'. Then, using $\mathcal{F}$, given a template t, one can obtain u' such that $H(u' \oplus u) \leq d$, without knowing d, with probability ϵ'. By the Simplified Assumption, this can only be done with probability lesser than $\epsilon_{\text{guess}} + \texttt{negl}$, which gives $\epsilon' \leq \epsilon_{\text{guess}} + \texttt{negl}$.   □

A.4 Evaluating the FAR and FRR

The FAR is, by definition, the probability that a wrong fingerprint is declared genuine. Therefore, the FAR is smaller than the attacker's success probability :

$$\text{FAR} \leq 2^{-k} \sum_{i=0}^{d} \binom{k}{i} + \texttt{negl}$$

Neglecting the term $\texttt{negl}$, the various $\{k, d\}$ choices and their corresponding FARs are shown in table 2.

Table 2. FAR for various $\{k, d\}$

$-\log_{10}(\text{FAR})$	2	3	3	4
k	10	20	26	30
d	2	3	5	5

The FRR being the percentage of correct fingerprints that do not pass the identification algorithm, the FRR is equal to $1 - \beta$, where β in the probability introduced in the Simplified Assumption. $1 - \beta$ must be small.

A Robust Fingerprint Minutiae Matching Algorithm Based on the Support Model

Xiaohui Xie, Fei Su, Anni Cai, and Jing'ao Sun

Biometric System Lab
Beijing Univ. of Posts and Telecommunications, China
xie_xiaohui@yahoo.com.cn
{sufei,anni_cai,jingsun}@bupt.edu.cn

Abstract. A novel method to match two minutiae point-sets based on the support model we proposed is presented in this paper. In this method, coarse matching is first performed from a number of seeds, and the results are then fused to obtain a constrained corresponding relationship of minutiae in two point-sets. By using the support degree of the elements in the constrained relations, the one-to-one correspondence is finally determined by comparison of similarity of local structures. Experiments show that this algorithm is robust and can deal with the translation, rotation, distortion and outlier problems well.

1 Introduction

Fingerprint minutiae are commonly employed as the basic features in most of present fingerprint recognition algorithms. In such circumstances, fingerprint recognition can be regarded as a planner point-set matching problem, where the best match with maximum of corresponding point pairs between two point-sets is searched under certain error constraints[1]. Many solutions have been proposed to solve this problem. A.K.Jain et al. used a string matching technique[2], Z.Chen et al. described a fingerprint verification algorithm based on topological structure[3], Isenor and Zaky proposed a method based on graph matching[4], N.K.Ratha et al. used local similarity to match two fingerprints[5], etc.

Although fingerprint minutiae matching algorithms have been studied since last two decades, there still have not a proper way to handle the elastic distortion problem in fingerprint matching. Looking for robust and efficient fingerprint minutiae matching algorithms is still a challenge problem.

In this paper, we first propose the support model(SM) we established for minutiae matching. Based on this model, we then present a novel fingerprint minutiae matching algorithm, and prove its robustness by experiments. No explicit registration step is required in this algorithm and it deals with the elastic distortion and outlier problems quit well. As its computational complexity is $O((C_M^3 + (M-3) \cdot N) \cdot C)$, the algorithm runs fast.

2   The Support Model

Suppose that we have two minutiae point-sets P and Q extracted from the template and the query images respectively, where $P = \{P_i(x_i,\ y_i,\ \theta_i) \mid i \leq M\}$ and $Q = \{Q_i(x_i,\ y_i,\ \theta_i) \mid i \leq N\}$, each minutia point is represented by its location $(x_i,\ y_i)$ and the direction θ of the ridge flow where it resides, and M and N are the number of minutiae in P and Q respectively.

Subset $L(A)$ of set Ω is called the local of point A if it satisfies the condition that the distance between each point in $L(A)$ and point A is less than a threshold Th, that is $L(A) = \{A_j \mid A_j \in \Omega,\ A_j \neq A,\ sqrt((A_j.x - A.x)^2 + (A_j.y - A.y)^2) < Th\}$, where $A.x$ represents the x component of A.

A binary relation $R(A) = \{(A_i,\ A_j) \mid A_i,\ A_j \in (L(A) \cup \{A\})\}$ is then defined to describe the relation between minutiae A and its local $L(A)$. $R(A)$ is also called the constraint relations of point A. Each element $(A_i,\ A_j)$ of R can be represented by a relation vector $V_{ij}(dist,\ angle,\ linea,\ lineb)$, where:

$$\begin{cases} V_{ij}.dist = \sqrt{(A_i.x - A_j.x)^2 + (A_i.y - A_j.y)^2} \\ V_{ij}.angle = \varphi(A_i.\theta - A_j.\theta) \\ V_{ij}.linea = \varphi(\overrightarrow{A_iA_j} - A_i.\theta) \\ V_{ij}.lineb = \varphi(\overrightarrow{A_iA_j} - A_j.\theta) \end{cases} \quad (1)$$

$\varphi(\alpha)$ in Eq. (1) is defined as $\varphi(\alpha) = \alpha \pm k \cdot 2\pi,\ k \in N$, with a range of $[\,0,\ 360)$. All of the relation vectors generated from one minutia's local compose a relation vector set $V = \{V_{ij} \mid (A_i,\ A_j) \in R\}$. The similarity of two relation vectors V_1 and V_2 from different point-sets can be measured by:

$$Similarity(V_1,\ V_2) = -U \cdot F(E) = 1 - U \cdot (f_1(e_1),\ f_2(e_2),\ f_3(e_3),\ f_4(e_4)) \quad (2)$$

where $E(e_1,\ e_2,\ e_3,\ e_4)$ is the distance between the two vectors V_1 and V_2, and its components are defined as below:

$$\begin{cases} E.e_1 = |V_1.dist - V_2.dist| \\ E.e_2 = |\phi(V_1.angle - V_2.angle)| \\ E.e_3 = |\phi(V_1 linea - V_2.linea)| \\ E.e_4 = |\phi(V_1.lineb - V_2.lineb)| \end{cases} \quad (3)$$

$\phi(\alpha)$ in Eq. (3) is defined as $\phi(\alpha) = \alpha \pm k \cdot 2\pi,\ k \in N$, with a range of $[-180,\ 180)$. F in Eq. (2) is a limiting-function cluster and it is defined as $if\ |e_i| < threshold_i,\ f_i(e_i) = e_i;\ else\ f_i(e_i) = \infty,\ i = 1,\ 2,\ 3,\ 4.\ U$ is a normalization factor, where $U = (k_1/threshold_1,\ k_2/threshold_2,\ k_3/threshold_3,\ k_4/threshold_4)$, and $\sum_i k_i = 1$.

Suppose that we have two minutiae point-sets P and Q from two images of the same finger. If minutiae points P_1 and P_2 in set P correspond respectively to minutiae points Q_1 and Q_2 in set Q, and $P_1 \in L(P_2)$ or $P_2 \in L(P_1)$, generally speaking, the similarity degree of $V_{P_1P_2}$ and $V_{Q_1Q_2}$ would have a relatively high value since distortion within a local area commonly is small.

If Θ is a subset of relation set $R(P_i)$, based on the above observation we can define the support set $Sup(P_i)_\Theta$ of minutia P_i under the constraint of Θ as a subset of Q which satisfies the condition below:

$$Sup(P_i)_\Theta =$$
$$\{Q_k \mid \forall j, \ (P_i, \ P_j) \in \Theta, \exists \ l, \ Similarity(V_{P_iP_j}, V_{Q_kQ_l}) > Threshold \} \ (4)$$

$Sup(p_i)_\Theta$ can be simply referred as the constrained support of P_i.

Eq. (4) implies that every element in $Sup(P_i)_\Theta$ is possible to match with minutia point P_i. $1/\|Sup(P_i)\|_\Theta$ is then defined as the minutia P_i's facticity. The higher the facticity, the less the number of candidates for P_i's corresponding point would be. If Θ is empty, that is to say, there is no relation constraint is required, every minutia in set Q then has the possibility to match with point P_i, i.e. $Sup(P_i)_\Theta = Q$ and $\|Sup(P_i)_\Theta\| = N$. If there exits only one minutia Q_k in set Q which corresponds to point P_i in minutia set P, then $Sup(P_i) = \{Q_k\}$ and $\|Sup(p_i)\| = 1$. From Eq. (4) we can also learn that $\|Sup(p_i)_{\Theta_1}\| \geq \|Sup(p_i)_{\Theta_2}\|$, if $\Theta_1 \subset \Theta_2$. That is to say, the number of elements in P_i's constrained support decreases monotonically as the number of relations in Θ increases, and the one-to-one correspondence between P_i and its mate point in Q becomes more clear. This observation provides a way to find the corresponding relationship between two minutiae point-sets P and Q. In other words, it provides a way to obtain the minimal constrained support of a minutiae in set P.

From Eq. (4) we can know that for every element $Q_k \in Sup(P_i)_\Theta$, its local $L(Q_k)$ has a similar structure with P_i's local $L(P_i)$. We then define the support degree of Q_k to P_i as $SD(Q_k \mid P_i)_\Theta$, which describes the similarity degree of the local minutia structure of Q_k to that of P_i:

$$SD(P_i \mid Q_k)_\Theta = \sum_{(P_i,P_j)\in\Theta, \ Q_l\in Sup(P_i)_\Theta} Similarity(V_{P_iP_j}, V_{Q_kQ_l})/\|\Theta\| \quad (5)$$

$Sup(P_i)_\Theta$ contains all minutiae that possibly match with P_i, but the elements of $Sup(P_i)_\Theta$ generally have different support degrees to point P_i. If minutia Q_k is the true match point of P_i, Q_k would have the highest of support degree to P_i among minutiae in $Sup(P_i)_\Theta$, since deformations at a small local area can always be considered very small.

3   Matching Based on the Support Model

Matching based on the Support Model(SM) can be explained as a procedure to first obtain the constrained support with the minimal number of elements for each minutiae in set P by utilizing of the minutiae's local relations, and then to increase the facticity of every point in set P through adding more constraint relations. By analyzing the different support degrees of minutiae in P_i's constrained support, we can finally determine the one-to-one correspondence between point sets P_i and Q.

The algorithm proposed in this paper can be divided into three stages: Coarse matching, C-Seed fusion and fine matching.

The details of the fingerprint minutiae matching algorithm we proposed is described as follows.

3.1 Coarse Matching

If a minutiae set S is chosen as a Seed, it should meet the following two conditions. Firstly, S is a subset of minutiae set P, and $\forall P_i \in S$, $P_i \in \bigcap\limits_{P_j \in S,\ P_i \neq P_j} L(P_j)$.
That is to say, the elements of the seed are distributed over a local area, and every minutia of the seed is a local component of other minutiae in S. Secondly, the constrained support $Sup(P_i)_H$ must not be empty for each P_i, where H is the relation set formed by minutiae of S, i.e., $H = \{(P_i,\ P_j) \mid P_i \neq P_j,\ P_i, P_j \in S\}$.

Generally speaking, a Seed should be chosen with a high probability of $Sup(P_i)_H \supseteq \{Q_j\}$ and with a small value of $\sum \|Sup(P_i)_H\|$, where Q_j is the true match point of P_i in Q. Thus the number of elements in S should not be too large, otherwise it would be hard to find an appropriate seed which satisfies so many relation constraints in H since the number of elements in H increases drastically when S has more components. In addition, the probability of $Sup(P_i)_H \supseteq \{Q_j\}$ decreases as the number of elements in seed increases. Generally $\|S\|$ is chosen as 2, 3 or 4. If the quality of image is good and a large number of minutiae can be extracted from the image, a bigger value for $\|S\|$ can be used, otherwise a smaller one should be taken.

Suppose that we choose $\|S\| = 3$. The seed is expressed as $S = \{P_1, P_2, P_3\}$, and the initial relation set $H_{init} = \{(P_1, P_2), (P_2, P_3), (P_3, P_1)\}$. The procedure of coarse matching is described as below:

1. Find the contained support of seed $S_{p1} = Sup(P_1)_{H_{init}}$, $S_{p2} = Sup(P_2)_{H_{init}}$, $S_{p3} = Sup(P_3)_{H_{init}}$, and initialize the support depth of the seed $depth = 1$;
2. Look for one minutiae $P_i(P_i \notin S)$ that satisfies the condition $\|L(P_i) \cap S\| \geq 2$, and denote the intersection of the local of P_i and the seed as S_{P_i}, i.e. $S_{P_i} = L(P_i) \cap S$;
3. Find the constrained support $Sup(P_i)_{R_{Pi}}$ under the constraints of relation $R_{Pi} = \{(P_i, P_j) \mid P_j \in S_{P_i}\}$;
4. Repeat step 2) to step 3), until all minutiae that satisfy conditions in 2) have been found;
5. Let $S_{depth} = \{P_i \mid \|Sup(P_i)_{R_{Pi}}\| \geq 1\}$ and $S \Leftarrow S \cup S_{depth}$;
6. Update the support depth: $depth \Leftarrow depth + 1$;
7. Repeat step 2) to step 6), until all minutiae in point set P have been searched.

It can be seen from the above steps, that if point P_i of P has merged into the Seed, the constrained support of P_i should contain its match point as long as its true match point exists and the following conditions hold: Firstly, minutiae Q_1, Q_2, Q_3 in set Q are respectively the corresponding points of P_1, P_2, P_3 in seed S; Secondly, there exist $S_{p1} \supseteq \{Q_1\}$, $S_{p2} \supseteq \{Q_2\}$ and $S_{p3} \supseteq \{Q_3\}$; Thirdly, the local deformation does not exceed a certain threshold.

It should be noticed that a term "support depth" has been introduced in the above steps, which describes how far matching can be extended from the seed to its neighborhood regions. It can be expected that the support depth of the seed whose constrained support contains its true match points would be greater than that when the constrained support of the seed does not contain its true match point.

3.2   C-Seed Fusion

If P_i is corresponding to point $Q_i' \in Q$ and $Q_i' \in Sup(P_i)_H$, we call the constrained support of P_i True Support; on the contrary, if $Q_i' \notin Sup(P_i)_H$, we call the constrained support of P_i False Support. There are three possibilities for constrained supports of minutiae in Seed S:

1. The constrained supports of all minutiae in seed S are all True Support, i.e., $Q_i' \in Sup(P_i)_H$ for $P_i \Leftrightarrow Q_i'$;
2. The constrained supports of all minutiae in seed S are all False Support, i.e., $Q_i' \notin Sup(P_i)_H$ for $P_i \Leftrightarrow Q_i'$. False support is mainly caused by reasons, such that the template and the query images are not overlapped on the area where S locates, or there is large deformation on the neighborhood of S, or some features have been lost due to bad image quality, etc.
3. The constrained supports of some minutiae in seed S are True Support, and others are False Support. This can happen when minutiae's true match point has been lost at feature extraction stage, but a substitute point around the lost one exists.

Obviously False Support of the Seed will lead the succeeding matching steps to wrong directions. In order to reduce the possibility of such a case, we introduce a C-Seed fusion method into the matching algorithm.

Initially, we choose C seeds. After performing coarse matching, we obtain the constrained supports of minutiae in each seed. We record the support depth of each seed and arrange the seeds in descending order, S^1, S^2, S^3 $\cdots$ S^C according to their depths.

Assume that the constrained support of P_i generated from seed S^a is $Sup_a(P_i)$ and its constrained support generated from seed S^b is $Sup_b(P_i)$. If the following condition is satisfied:

$$\exists P_i \in P, \quad Sup_a(P_i) \cap Sup_b(P_i) \neq \Phi \tag{6}$$

We defined a combined constrained support below:

$$Sup(P_i) = \begin{cases} Sup_a(P_i) \cap Sup_b(P_i) & if\ Sup_a(P_i) \cap Sup_b(P_i) \neq \Phi \\ Sup_a(P_i) & if\ Sup_b(P_i) = \Phi \\ Sup_b(P_i) & if\ Sup_a(P_i) = \Phi \end{cases} \tag{7}$$

The minutiae's support degrees are also modified during the fusion process:

$$SD(P_i|Q_j) = \begin{cases} max(SD_a(P_i \mid Q_j), SD_b(P_i \mid Q_j)) \\ \qquad\qquad if\ Sup_a(P_i) \cap Sup_b(P_i) \neq \Phi \\ SD_a(P_i \mid Q_j) & if\ Sup_b(P_i) = \Phi \\ SD_b(P_i \mid Q_j) & if\ Sup_a(P_i) = \Phi \end{cases} \tag{8}$$

Using Eqs. (7) and (8), we combine the results of two seeds S^1, S^2 which have the two highest support depth, and then combine the new result with the results of $S^3 \cdots S^C$ sequentially. We obtain the final restricted support of every minutiae.

3.3   Fine Matching

Although after the above steps the facticities of all minutiae become higher than that before C-Seed fusion, some minutiae may still have more than one candidates as their corresponding points. The reason for this is that the relation set so far used in the procedure of calculating the constrained support is only a subset of the minutiae's local relations. Hence we will confirm the matching relationship for each minutiae at the fine matching stage.

we search minutiae whose constrained support meet $\|Sup(P_i)\| > 1$, and choose minutiae Q_j which has the maximal support degree, $SD(P_i \mid Q_j) = \max_k(SD(P_i \mid Q_k))$, as the corresponding point of P_i. Thus, one-to-one matched pairs (P_i, Q_j), $P_i \in P$, $Q_j \in Q$ can be obtained.

In order to guarantee a correct result, we also recheck every matched point pair to make sure that their matching error is less than a certain threshold.

3.4   Similarity Measures

The computational complexity of the algorithm described above is $O(C \cdot (C_N^3 + (M-3) \cdot N))$. Suppose that only M' pairs of matched points have been found during the matching, i.e., (P_i, Q_i), $i \leq M' \leq min(M, N)$. We define a local similarity coefficient $e_1 = \sum_{i \leq M'} SD(P_i \mid Q_i) / M'$ and a global similarity coefficient $e_2 = \sum_{i, j, i \neq j} Similarity(V_{P_i P_j}, V_{Q_i Q_j}) \cdot 2 / (M' \times (M' - 1))$. Then the degree of relative deformation for the two minutiae point-sets can be described by $distortion = e_1/e_2$, and the similarity of the two point-sets can be measured by:

$$score = g \cdot (M' / min(M, N)) \cdot e_1 \cdot distortion \tag{9}$$

where g is a constant.

4   Experiment Results and Performance Evaluation

We tested our algorithm on the first database of FVC2002[6], which contains 800 gray-level fingerprint images. These images are 388×374 pixels with a resolution of 500dpi, captured from 100 different fingers with 8 sample images each. We compare every two fingerprints in the database, and get $100 \times (8 \times 7)/2 = 2800$ times true matching and $100 \times 90 \times 8/2 = 36000$ times false matching. The average matching time is 0.015s per match by using a laptop with a PIII 866 CPU. According to the similarity score level distribution curve of the true matches and the false matches, we can derive the false matching rate curve and the false accept rate curve. These curves are shown in Fig.1 , and in the figure we also
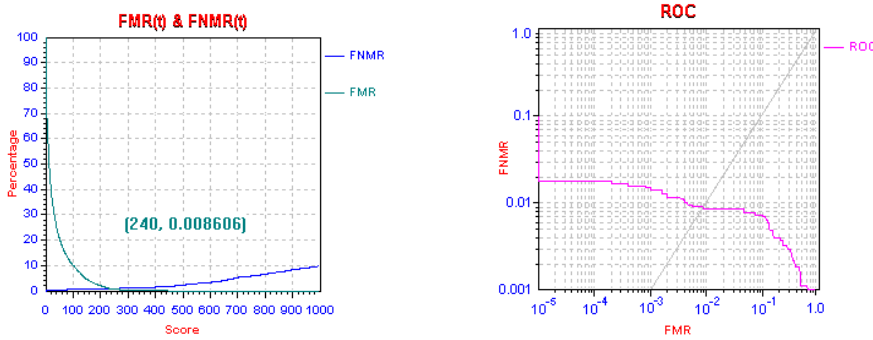
Fig. 1. The result's curves of the algorithm on the first database of FVC2002

Table 1. The influence with different number of seeds

Number of the Seeds (C)	2	5	10	13	16	20
Equal error rate (EER %)	2.1	1.4	0.93	0.85	0.8	0.78

give the ROC curves of the result. The experiment shows that the algorithm has excellent performance both in accuracy and in speed. Table 1 shows the influence to the matching results with different number of seeds. From the table we can see that if the number of seeds used in matching is small, the probability of obtaining a false match would be high because in such a case the possibilities of obtaining False Supports of the seeds are relatively large. The algorithm's performance becomes steady as the number of seeds used in the algorithm is more than a certain value 16.

5   Summary and Future Work

In this paper, we have presented a novel fingerprint minutiae matching algorithm based on the support model. In each matching step the robustness of the algorithm is ensured. Experiments have shown the effectiveness of the support model and the new algorithm based on this model. How to increase the probability of obtaining true support of the seed during the coarse matching procedure, and how to decrease the false matched pairs generated during the C-Seed fusion procedure are key problems in the future works.

References

[1] N. K.Ratha, K.Karu, S.Chen, and A. K.Jain. *A Real-time Matching System for Large Fingerprint Database.* IEEE Trans. on Pattern Analysis and Machine Intelligence. 18 (8): 799-813. Aug 1996.   316
[2] A. K.Jain, L.Hong, and R. M.Bolle. *On-line Fingerprint Verification.* IEEE Trans. on Pattern Analysis and Machine Intelligence. 19 (4): 302-313, April 1997.   316

[3] Z.Chen, C. H.Kou, *A Toplogy-based Matching Algorithm for Fingerprint Authentication.* Security Technology. 1991. Proceedings. 25th Annual 1991 IEEE International Carnahan Conference on , 1-3 Oct. 1991 Page(s): 84-87.   316

[4] D. K.Isenor and S. G.Zaky. *Fingerprint Identification Using Graph Matching. Pattern Recognition.* 19(2): 113-122, 1986.   316

[5] N. K.Ratha, R. M.Bolle, V. D.Pandit, V.Vaish. *Robust Fingerprint Authentication Using Local Structural Similarity.* Applications of Computer Vision, 2000, Fifth IEEE Workshop on., 4-6 Dec. 2000 Page(s) : 29-34.   316

[6] D.Maio, D.Maltoni, R.Cappelli, J. L.Wayman, A. K.Jain. *FVC2002: Second Fingerprint Verification Competition.* Pattern Recognition, 2002, Proceedings. 16th International Conference on., 11-15 Aug. 2002 Page(s): 811-814 vol.3.   321

Fingerprint Classification: An Approach Based on Singularities and Analysis of Fingerprint Structure

Xin Wang and Mei Xie

College of Electronics Engineering
University of Electronic Science and Technology of China, Chengdu 610054, China
mxie@uestc.edu.cn

Abstract. In this paper, we propose a new approach to fingerprint classification based on both singularities and analysis of fingerprint structure. Fingerprints are classified into five categories: arch, tented arch, left loop, right loop and whorl. The directional image is firstly divided into nonoverlapping regions to give a synthetic representation. Then, singularities (delta and core points) are extracted from a fingerprint based on the directional image with a Poincare index method. Combined with the number and location of the detected singularities, the fingerprint structural information can be exploited to classify the fingerprints. In our experiments, we focus on the fingerprint structural information to classify fingerprints. The method is invariant to rotation, translation and small amounts of scale changes. Furthermore, it does not require any iterations or feedback. Experiments have been executed on NIST4 databases.

1 Introduction

Fingerprints have long been used for personal identification. It is assumed that fingerprint matching is considered one of the most reliable techniques of people identification. Fingerprint matching is usually carried out at two different levels. At the coarse level, fingerprint can be classified into five main classes: arch, tented arch, left loop, right loop and whorl, as shown in Fig.1 [1].

The fine-level matching is performed by extracting and comparing two sets of minutiae points, ridge endings and ridge bifurcations, from a fingerprint image. Although the coarse classification does not identify a fingerprint uniquely, it is helpful in determining when two fingerprints do not match [2].

The problem of fingerprint classification has been addressed by a number of authors in the past [1]. In this paper, we develop a method based on both the singularities and analysis of fingerprint structure to classify fingerprints.

Fig. 1. From left to right, we have arch, tented arch, left-loop, right-loop and whorl

We will discuss the details of our fingerprint classification method in the following sections. Section 2 presents the singularities detection method. Section 3 presents the analysis of fingerprint structure. In Section 4, some experimental results on the NIST4 databases are shown. Conclusion is presented in Section 5.

2   Singularities Detection

A captured fingerprint image usually consists of two components, the foreground and the background. The foreground is the fingerprint area and the noisy area at the borders of the image is called the background. The block is considered to be the foreground if its local variance satisfies some predefined requirement, otherwise, the background. If a fingerprint's foreground hasn't reached our requirements, it will be rejected.

2.1 Orientation Field Computation

Since fingerprints are graphical flow-like ridges, it is reasonable to use direction as the representation of fingerprint ridges or valleys. The orientation field that represents an intrinsic property of the ridges or valleys on the fingerprint image is most commonly used in the fingerprint verification systems [3].

In our algorithm, we use a facile method to calculate the orientation estimation filed .The algorithm consists of the following main steps.

1)   Divide the input fingerprint images into blocks of equal size $w \times w$
2)   Quantify the region $[0, \pi)$ into eight directions (shown in Fig.2).

The directions are corresponding to θ_i, $\theta_i = 22.5^o * i$ (i=0,1,2,3,4,5,6,7).

3)   Calculate x $signature$ $x_l(i, j)$ $(l = 0,1,2,3,4,5,6,7)$ for pixel(i,j) in each block. The equation used for computation is

$$x_l(i, j) = \sum_{m=1}^{K} \left| C(i_m, j_m) - \frac{1}{K+1} \sum_{m=0}^{K} C(i_m, j_m) \right| \tag{1}$$

where $C(i_0, j_0)$ indicates the gray value of the pixel(i,j) and $C(i_m, j_m)$ $(m = 1,...,K)$, the gray values of continuing pixels in a direction. K is the number of pixels chosen in each direction, which is determined by the type of images (resolution, line width, etc).

4)   Estimate the local orientation $\theta(i, j)$ using the following formula:

$$\theta(i, j) = Min \sum_{u=1}^{w} \sum_{v=1}^{w} x_l(u, v) \tag{2}$$

where w is the size of the block; $x_l(u, v)$ is the x $signature$ for pixel(u,v).
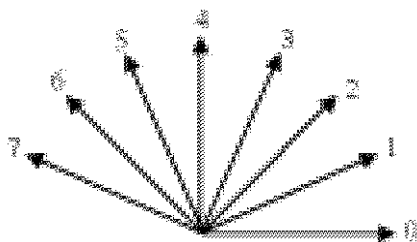
Fig. 2. The eight directions

Due to the presence of noisy regions in the fingerprint image, the estimated orientation field may not always be correct. A postprocessing procedure needs to be applied to modify the incorrect local orientation [4]. A method based on the correlation of blocks is proposed to obtain the smooth orientation field $\theta'(i, j)$.

2.2 Singularities Detection

A lot of methods have been proposed to the detection of singularities, while the Poincare index that derived from continuous curves is the most popular one. And more, the Poincare index on the orientation field can be computed to determine all the possible delta and core points in the fingerprint.

In this paper, we develop a new method to get the Poincare index on the orientation field. A 5×5 mask that is used to calculate and verify the Poincare index are shown below:

First, the Poincare index of the block M is obtained with the following formula:

$$Poincareindex(M) = \sum_{k=0}^{7}(\Delta_k) \tag{3}$$

where

$$\Delta_k = \begin{cases} \theta_k - \theta_{k+1 \mod 8} & where \ \left|\theta_k - \theta_{k+1 \mod 8}\right| =< 4 \\ 8 + \theta_k - \theta_{k+1 \mod 8} & where \ \ \theta_k - \theta_{k+1 \mod 8} < -4 \\ \theta_k - \theta_{k+1 \mod 8} - 8 & where \ \ \theta_k - \theta_{k+1 \mod 8} > 4 \end{cases} \tag{4}$$

P_0		P_1		P_2
	P_0	P_1	P_2	
P_7	P_7	M	P_3	P_3
	P_6	P_5	P_4	
P_6		P_5		P_4

Fig. 3. The 5×5 mask

and θ_k is the direction of the block P_k in the neighboring 3×3 blocks.

Then, the candidate singularities can be detected based on the Poincare index.

    i. If the Poincare index=8, the block M is a candidate core point.
    ii. If the Poincare index=-8, the block M is a candidate delta point.

After this, there will be the verification of such singular points. Only if the Poincare index calculated from the 3×3 and 5×5 around the candidate have the same Poincare value, the candidate is considered as a real singular point [5].

From the number of cores N_{ci} and deltas N_{di}, we can classify the fingerprints as whorl, candidate loop, tented arch and arch.

3   Analysis of Fingerprint Structure

At the former processes, the fingerprints are coarsely classified into arch, tented arch, candidate loop and whorl. But the number and location of cores and deltas are not enough for the detailed classification of the loop and tented arch. Thus, some more information is required to accomplish the fingerprint classification.

Flow-like tracing technique is used to analyze the fingerprint structure. By tracing from the inner side to the outside of a core, we can see that there is a tendency of angle variation. Following a left or right loop ridge, the tendency is distinct. In our algorithm, this feature is employed in distinguishing a fingerprint with one core as left-loop, right-loop or tented arch. Three samples before flow-like tracing are shown in Fig.5.
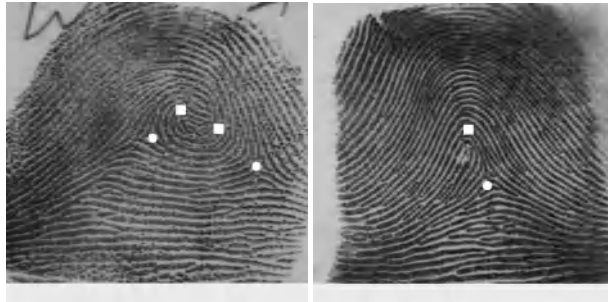
Fig. 4. Singular points found (circle as core and square as delta)

Fig. 5. Samples before flow-like tracing (left-loop, right-loop and tented arch)
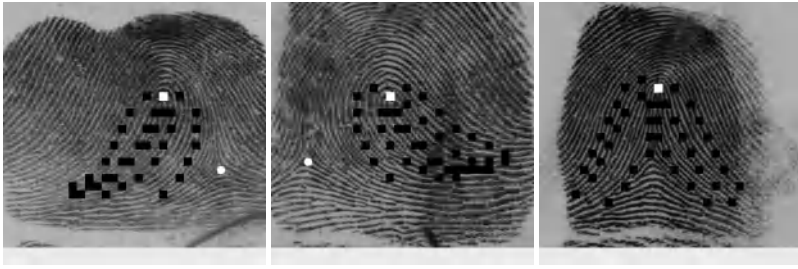
Fig. 6. Samples after flow-like tracing (left-loop, right-loop and tented arch)

With this structural feature, the fingerprints with only one core point can be uniquely classified. The detailed steps are as following:

i. Start from each 3×3 neighboring blocks of the core point, tracing the orientation in two opposite directions with the tracing step is predefined as 2; the tracing stops when it meets any of the following situations:

   1. It traces to the core point or its 3×3 neighboring blocks.
   2. The tendency of angle variation has reverted abruptly.
   3. It reaches the segmented boundary or reached as a predefined value.

   For each of the 3×3 neighboring blocks, if tracing ends with the situation 3, the correspondent neighboring block will be regarded to have a potential left or right tendency.

ii. The numbers of the neighboring blocks with left or right tendency are counted as N_l and N_r, respectively. By comparing the two numbers, the tendency of the fingerprint can be determined and fingerprints with one core point are classified. In our algorithm, with a threshold T_c, we determine the detailed classification. $T_c = (N_l + N_r)*\beta$, where β is an empirical value. Samples after flow-like tracing are shown in Fig.6.

4   Experimental Results

The classification method described above has been carried out on the fingerprints in the NIST4 databases and the 5-class results are shown in the table1. Because a variety of ambiguities occurring in the fingerprint, 17.5% fingerprints in the NIST4 are cross-referenced. So, we consider our classification result to be correct if it matches the assigned class or the cross-referenced class in [6].

Because the tented arch and arch have subtle difference, it is reasonable to combine them into one class and if so, the average error in classifying the fingerprints into five classes is about 18% without rejection in our experiments. With excluding the fingerprints rejected (14.4%), the classification accuracy can rise to 94%.

Table 1. Five-class classification results on NIST4

As.\nTrue	Whorl	Left loop	Right loop	Tented Arch	Arch
Whorl	**612**	12	8	6	3
Left loop	14	**581**	2	28	16
Right loop	14	5	**618**	27	14
Tented Arch	3	23	28	**412**	246
Arch	4	3	3	82	**664**

The other largest source of error is the misclassification between the arch/tented arch and some loops due to the similar ridge pattern or poor image quality. In the paper, we strive to make the best use of fingerprint structure, and haven't applied the delta points in the detailed classification, which is usually a common method to other researchers. Classifying the fingerprints with as few factors as possible is one of our main purposes in the experiments.

5   Conclusions

The classification method we proposed has a good performance when false singular points exist or some true singular points are missing. From the experimental results, we observe that the fingerprint structure is useful in fingerprint classification. However, the method based on fingerprint structure only performs well to fingerprints with distinct angle variation. When applied to distinguish the arch/tented arch from loops, our approach is not very encouraging. Now, we are studying a lot of classification schemes to find a combined method with higher efficiency and less complexity.

Reference

[1]   Jain, A. K., Pankanti, S.: Fingerprint Classification and Recognition: The Image and Video Processing Handbook, A. Bovik (ed), Academic Press (2000).
[2]   Karu, K., Jain, A.K.: Fingerprint Classification: Pattern Recognition, Vol.29, 1996 (389-404).
[3]   Qinzhi, Z., Kai, H., Hong, Y.: Fingerprint Classification Based On Extraction and Analysis Of Singularities and Pseudoridges: Conference in Research and Practice in Information Technology, Vol. 11, 2001(83 - 87).
[4]   Hong L., YiFei, W., Jain, A.K.: Fingerprint Image Enhancement: Algorithm and Performance Evaluation: IEEE Transactions on PAMI, Vol. 20,1998(777-789).
[5]   Byoung-Ho Cho, Jeung-Seop Kim, Jae-Hyung Bae, In-Gu Bae, Kee-Young Yoo: Core-based Fingerprint Image Classification: Proceedings of 15th International Conference on Pattern Recognition, Vol. 2, 2000(859-862).
[6]   Watson, C.L., Wilson, C.L.: NIST Special Database 4, Fingerprint Database, National Institute of Standards and Technology, 1992.

Characteristics of the Identification Algorithm Using a Matching Score Matrix

Takuji Maeda, Masahito Matsushita, and Koichi Sasakawa

Advanced Technology R & D Center, Mitsubishi Electric Corporation
Tsukaguchi Honmachi 8–1–1, Amagasaki, Hyogo, 661–8661 Japan
{maeda.takuji,matsushita.masahito,sasakawa.koichi}@wrc.melco.co.jp

Abstract. Biometrics authentication can be achieved by either verification or identification. In terms of convenience, identification are superior since a user does not have to input his/her ID number. However, it must be capable of searching the database storing user enrollment templates at high speed. To meet this need, we proposed an identification algorithm. In this paper, we describe our proposed method and discuss its characteristics of response speed using some simulation results. It is shown that response speed depends on the way to select enrollment data.

1 Introduction

Biometrics authentication systems can be achieved by either verification (one-to-one matching) or identification (one-to-many matching). In the verification procedure, a quick response can be expected because the matching is executed only once, though the claimer has to enter his/her ID number for every authentication session. In terms of convenience, identification is superior since there is no need for the user to input a personal ID number. However, it must be capable of searching the database storing user enrollment templates at high speed. To meet this need, methods for increasing search speed by reducing the number of match candidates have been proposed[1, 2].

In this paper, we describe our algorithm to achieve identification in a short period of time[3]. The behavior of the algorithm is illustrated by an experimental simulation result. And a method to improve response speed is described with an additional simulation result.

2 Identification Algorithm

2.1 Conventional Methods

There exists some methods to realize identification. Here shows two representative types of conventional identification method. One of them is linear search. It is realized by executing one-to-one matching step by step from the top of a database. The response speed to reach the identity data depends on the position in the database. When an amount of n data is enrolled in the database,

Table 1. Variables

n	The amount of enrolled data
E_i	Enrolled data distinguished by i
V_u	Input data distinguished by u (u is unknown)
$x_{i,j}$	The matching score of an enrolled data E_i and another enrolled data E_j
$y_{u,j}$	The matching score of an input data V_u and an enrolled data E_j
m	A counter indicating the step in an identification process
$r(m)$	Candidate data number one of the enrolled data at the mth step
$\boldsymbol{x_i}(m)$	The subset score vector at the ith row of the matching score matrix at the mth step
$\boldsymbol{y_u}(m)$	The score vector which consists of the matching scores of input data V_u and the candidate enrolled data $E_{r(k)}$ ($k = 1, 2, \cdots, m$) at the mth step
$z_i(m)$	The correlation value of $\boldsymbol{y_u}(m)$ and $\boldsymbol{x_i}(m)$

one-to-one matching of $n/2$ times on average needs to be done until the identity data is finally found. Accordingly when n is very large, this method has the difficulties that the time needed for the one-to-one matching operation grows in proportion to n.

The other method in which the search time is shortened, i.e., by data clustering[1, 2]. Each enrollment data is classified due to its feature or some kind of property and stored in a cluster that the data belongs to in the enrollment phase. In the identification phase, a cluster of input data is first chosen; in most cases, linear search is executed in each cluster. A problem identical to that in linear search occurs because the cluster size gets larger when the number of enrolled data increases. Other problems also present themselves, e.g., the way to classify the data in the enrollment phase has to be defined for each kind of biometrics and any mis-classification[4] in the identification phase can cause a slow response.

2.2   Proposed Method

We proposed a method to identify a person in a database in an instant[3]. The method does not require any kinds of classification or pre-ordering. In the proposed method, a quick response can be achieved by manipulating the search order according to the evaluation value which is calculated using a matching score matrix. This algorithm is described below. The variables used for the explanation are defined in Table 1.

Enrollment Phase. Let's suppose $n - 1$ biometrics data (E_1 to E_{n-1}) have been already enrolled in the database and the new nth biometrics data (E_n) is to be enrolled. A matching score matrix, which consists of matching scores calculated by all combination of $n - 1$ enrolled data, has already been built.

1. In order to update the matching score matrix, the new input data for enrollment is verified with all of the enrollment data and matching scores $x_{n,i}$ are calculated for $i = 1, 2, \ldots, n$.
2. The matching scores calculated at the previous step are added as a last row of the new matching score matrix
3. The nth column is copied as in Eq.(1) due to the symmetry of a matching score matrix.

$$x_{i,n} = x_{n,i} \quad (i = 1, 2, \ldots, n - 1) \tag{1}$$

It takes long time to calculate $x_{n,i}$ in proportion to the number of n. This is a known problem that we should solve.

Identification Phase. Here, the process of the proposed algorithm is described using an example which n data is already enrolled. Let's suppose an unknown claimer inputs his/her biometrics data V_u. Set the counter as $m = 1$, the current candidate data number as $r(m) = 1$, and the threshold as 50 before the identification process is started. Figure 1 illustrates a matching score matrix made in the enrollment phase.

1. V_u is compared with $E_{r(1)}(= E_1)$ and $y_{u,1}$ is calculated to 8. As it is less than the threshold value, the next candidate data is determined as follows. The values $z_i(1)$ for all i excluding already tried ones are calculated as in Eq.(4) using the current matching score vector $\boldsymbol{y_u}(1) = \{8\}$ defined in Eq.(2) and the subset score vector at the ith row of the matching score matrix $\boldsymbol{x_i}(1)$ defined in Eq.(3).

$$\boldsymbol{y_u}(m) = \{y_{u,r(1)}, y_{u,r(2)}, \ldots, y_{u,r(m)}\} \tag{2}$$

$$\boldsymbol{x_i}(m) = \{x_{i,r(1)}, x_{i,r(2)}, \ldots, x_{i,r(m)}\} \tag{3}$$

$$z_i(m) = \frac{2\,\boldsymbol{x_i}(m) \cdot\, \boldsymbol{y_u}(m)}{\boldsymbol{x_i}(m) \cdot\, \boldsymbol{x_i}(m) +\, \boldsymbol{y_u}(m) \cdot\, \boldsymbol{y_u}(m)} \tag{4}$$

The max value among all of the $z_i(1)$ is chosen. In this case, E_4 becomes the next candidate data because $\boldsymbol{y_u}(1) = \{8\}$ best matches $\boldsymbol{x_4}(1) = \{8\}$. m is set to 2 for the next step.
2. V_u is compared with E_4 and a matching score $y_{u,4}$ is calculated. As $y_{u,4}$ is 11 and less than the threshold, $z_i(2)$ against $\boldsymbol{y_u}(2) = \{8, 11\}$ are calculated. $z_i(2)$ is maximized when $i = n$ and the next candidate is E_n because $\boldsymbol{y_u}(2) = \{8, 11\}$ best matches $\boldsymbol{x_n}(2) = \{9, 11\}$. m is set to 3 for the next step.
3. The matching score $y_{u,n}$ of V_u and E_n is calculated as 16. As it does not exceed the threshold, $z_i(3)$ against $\boldsymbol{y_u}(3) = \{8, 11, 16\}$ are calculated. $z_i(3)$ is maximized when $i = 5$ and the next candidate is E_5 because $\boldsymbol{y_u}(3) = \{8, 11, 16\}$ best matches $\boldsymbol{x_5}(3) = \{9, 10, 15\}$. m is set to 4 for the next step.
4. The matching score $y_{u,5}$ of V_u and E_5 is 85 and it is over the threshold. The result therefore shows that V_u corresponds to E_5 in the database.

This algorithm corresponds to the process that the 1D waveform $\boldsymbol{x_i}(n)$ in the matching score matrix that best matches the 1D waveform $\boldsymbol{y_u}(n)$ is searched.

a matching score matrix

	E_1	E_2	E_3	E_4	E_5			E_n
1st ▶ E_1	100	10	25	8	9			9
E_2	10	100	14	15	6			14
E_3	25	14	100	17	18			2
2nd ▶ E_4	8	15	17	100	10			11
Hit! ▶ E_5	9	6	18	10	100			15
3rd ▶ E_n	9	14	2	11	15			100

a matching score vector

candidate data E_1 E_4 E_5 E_n

V_u	8	—	—	11	85		16

1st   2nd Hit!   3rd
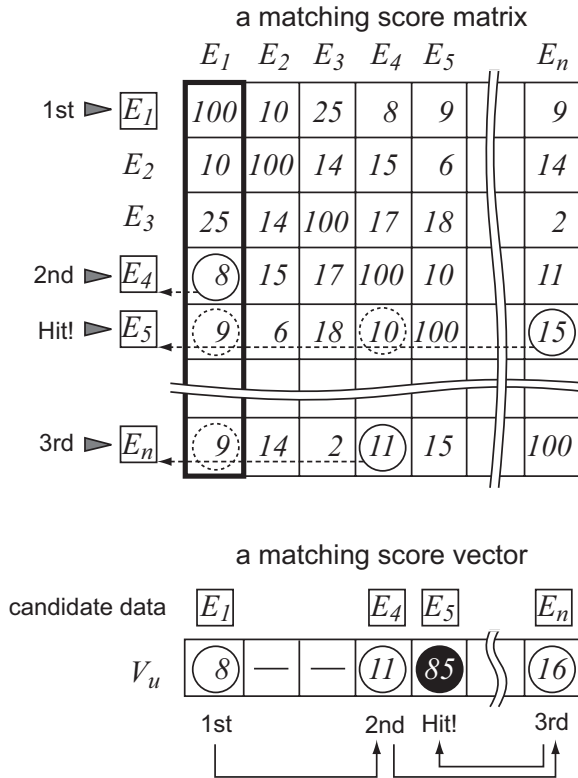
Fig. 1. An example of a matching score matrix and a matching score vector

3   Simulation Results

3.1   Simulation Conditions

We show some experimental computer simulation results in order to examine effects of the proposed algorithm. Common simulation conditions are shown in Table 2.

3.2   Response Speed Characteristics

To evaluate the response speed of proposed algorithm, we examine the distribution of required steps to identity data in a simulation process. It is described as a relative frequency indicating the claimer's template data will appear as a candidate at a certain step in an identification process. We discuss the response speed characteristics using the required steps to identity data because the computational complexity of the sequence to determine the next candidate in Eq.(4) is much less than that of one-to-one matching.

Table 2. Common simulation conditions

Condition	Value
Kind of biometrics	Fingerprint
Sensor product	FPR-DT mk2, Mitsubishi Electric Corp.
Sensor specification	optical type,minutiae-based,320×240 pixels, 500 dpi
Enrolled number of data	2000
Captured data in an enrollment phase	8 data/finger
Data for enrollment	1 data/finger which is selected from captured data
Data for use in an identification phase	7 data/finger
The number of the simulations	14,000 trials = $7 \times 2,000$

At first, a simulation result labeled [1st] is shown in Figure 2. It is under the condition that the first data of 8 captured data is chosen for an enrollment data and to create a matching score matrix. In the figure, the horizontal axis denotes the required steps to identity data and the left vertical axis denotes the relative frequency of the required steps. Those axes are logarithmic. The right vertical axis denotes the cumulative relative frequency which is linear.

The shape of the relative frequency looks compressed to the range of little steps. Its peak is around 10-20 range and 80 % of this distribution extends up to about 400 steps. This result means that the claimer's template data appears
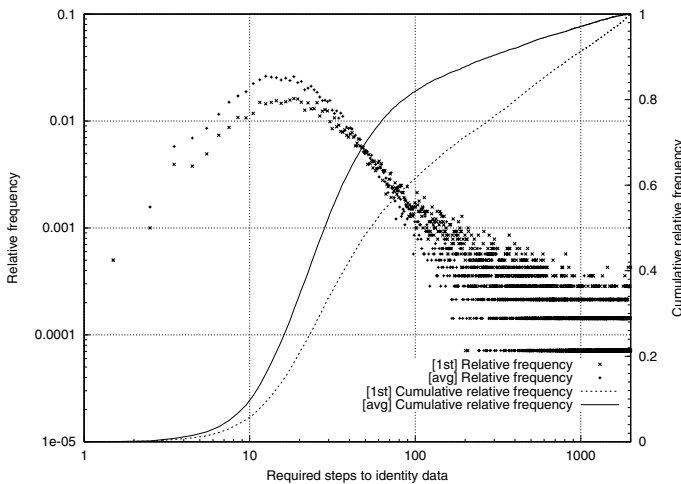
Fig. 2. Distribution of required steps to identity data

early as a candidate in an identification process and this characteristics differ greatly from those of conventional methods.

In addition, as most of data are found in short steps, the identification process can be stopped before the last step to make response time be shorter. As the False Rejection Rate will change according to the timing to stop the identification process, the accuracy of the identification can be better if the search process is not stopped and executed to the last step.

Note that it depends on the threshold or quality of the input data if the candidate template data is accepted as the claimer's data or not. We will discuss this topic in another paper.

3.3   Response Speed Improvement

In the above simulation, the enrollment data is selected simply as the first trial of 8 captured data. For a matching score matrix to be robust, it may be effective to appropriately determine the suitable data for the creation of a matching score matrix in the enrollment phase. To confirm this hypothesis, we select one data producing the highest possible matching score from all captured data. An average matching score is used to select an enrollment data. An average matching score is calculated by verification with all of the other data within the captured data. This condition is distinguished by the label [avg]. A simulation result under the condition [avg] is also shown in Figure 2.

The shape of the relative frequency of [avg] is more compressed to the range of little steps than that of [1st]. It can be seen from the difference of the cumulative relative frequency curves. The cumulative relative frequency curve of [avg] has higher value than that of [1st] and that means the response speed has been improved by the condition [avg].

As the response speed characteristics is affected by the selection way of the enrollment data, the enrollment selection strategy is very important to improve response performance. For example, it is supposable that good quality data causes quick response, on the other hand poor quality data causes slow response. Moreover, it may let the speed performance be better that updating enrolled data in the identification phase.

3.4   The Relation between the Number of Enrolled Data and Response Speed

The relation between the number of enrolled data and the required steps to identity data on average is shown in Figure 3. As an example of conventional methods, linear search curve is also shown for comparison.

The linear search needed $n/2$ steps on average when n data was enrolled. On the other hand, the proposed method needed experimentally about $\sqrt{n}$ steps, a lot fewer than the conventional approaches. So the difference between them will extend according to the value of n. The proposed method has great advantage in proportion to n. It is our future work to analyze the reason why our method needs about $\sqrt{n}$ steps on average.
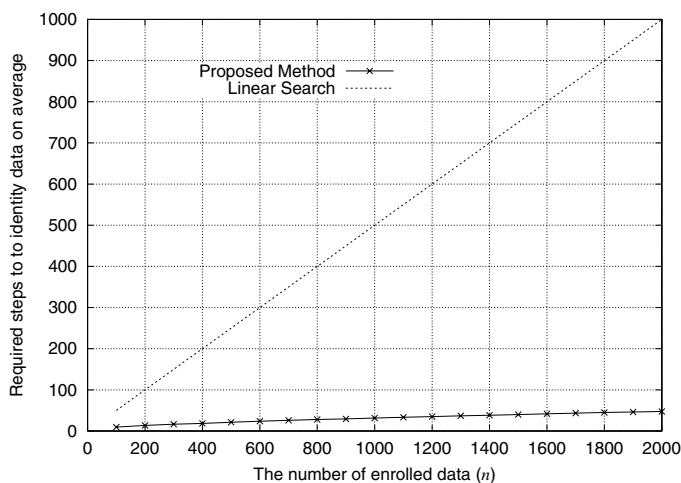
Fig. 3. Relation between the number of enrolled data and the required steps

4   Conclusions

We described our proposed identification algorithm which needs much fewer steps to identity data than the conventional approaches. It is very useful for any kinds of biometrics because it depends on only matching scores that are generally defined in most biometrics.

And also we discussed its response speed characteristics using some simulation results. It is found that the response speed depends on the enrollment data. It is very important how to choose the enrollment data because they are used for making a matching score matrix and the values in the matrix make a great influence to the sequence of an identification process. It is our future work to make a matching score matrix to be more robust for various input data to obtain stable and good performance.

References

[1] T. Kamei and M. Mizoguchi, "Fingerprint preselection using eigenfeatures," Proc. IEEE Computer Society Conference, Computer Vision and Pattern Recognition, pp.918-923, 1998. 330, 331
[2] R. S. Germain, A. Califano, and S. Colville, "Fingerprint matching using transformation parameter clustering," IEEE Computational Science and Engineering, vol.4, Issue 4, pp.42-49, 1997. 330, 331
[3] T.Maeda, M.Matsushita, and K.Sasakawa, "Identification algorithm using a matching score matrix," , IEICE Trans. on Information and Systems, Vol.E84-D, No.7, pp.819-824, 2001. 330, 331
[4] W. Shen, "Automated fingerprint pattern classification error analysis," Lect. Notes Computer Science, vol.1351, pp.24-31, 1997. 331

Fingerprint Matching Integrating the Global Orientation Field with Minutia

Jin Qi, Yangsheng Wang, Zhongchao Shi, Ke Xu, and Xuying Zhao

National Laboratory of Pattern Recognition
Institute of Automation, Chinese, Academy of Sciences, Beijing 100080, P.R.China
jqi@nlpr.ia.ac.cn

Abstract. We define a novel structure for each minutia in the fingerprint based on the global orientation field. Then the structure pattern is used to establish the minutia correspondence between the test fingerprint and template fingerprint. A new technique calculating the matching score is developed based on both the number of corresponding minutia pairs and the similarity level between the template fingerprint orientation field and the aligned test fingerprint orientation field. The results of the experiment conducted on the public data collection, DB3, demonstrate the effectiveness of the method in this paper.

1 Introduction

Fingerprint matching techniques can be broadly classified as being minutiae-based or correlation-based. Several minutiae matching approaches have been proposed in the literature. These include methods based on structure matching [1], [2], [3], [4], alignment matching in [5], [6], [7], non-linear transformation [8], [9]. The key aspect of all these methods is to obtain the minutiae correspondences accurately.

We develop a new method of fingerprint matching incorporating the global fingerprint orientation field with fingerprint minutiae properly. In addition, we present a new technique to compute the matching score based on the number of matching minutia pairs and the similarity level between the two orientation fields one from template fingerprint and another from registered fingerprint, respectively.

The rest of the paper is organized as follows. A detailed definition of our novel minutia structure is presented in the following section. The matching scheme based on the proposed minutia structure is developed in Section 3. This is followed by validation experiments conducted on the public domain collection of fingerprint images, DB3. Finally, concluding remarks are presented in Section 5.

2 Definition of the Novel Structure

In general, a minutia point M_k detected from a fingerprint can be described by a feature vector given by:

$$F_k = (x_k \qquad y_k \qquad \varphi_k)^T \ .\qquad (1)$$

where (x_k, y_k) is its coordinate, φ_k is the local ridge direction.

We define a function $d\phi(t_1, t_2)$ for the difference between two directions or angles t_1 and t_2, $-\pi/2 < t_1 < \pi/2, -\pi/2 < t_2 < \pi/2$, as follows:

$$d\phi(t_1, t_2) = t_1 - t_2 \; . \tag{2}$$

Given a minutia point M_k with orientation φ_k, we define our minutia structure as following procedures:

First, we can get three angles θ_1, θ_2 and θ_3. Let $\theta_1 = \varphi_k, \theta_2 = \varphi_k + 2\pi/3$ and $\theta_3 = \theta_2 + 2\pi/3$. Then we plot three lines l_1, l_2 and l_3 along the angles θ_1, θ_2 and θ_3 with respect to X axis through the minutia point M_k. A sampling step is done starting with the minutia point M_k along each line with sampling interval τ. The sampling step along each line stops till the latest sampling point falls in fingerprint background region, as illustrated in Fig.1.

The sampling pattern consists of three lines l_m ($1 \le m \le 3$) with three positive directions θ_m, ($1 \le m \le 3$), each one of them comprising $N_{l_m}^k$ sampling points p_{i,l_m}^k, ($1 \le i \le N_{l_m}^k, 1 \le m \le 3$), equally distributed along the line l_m. Denoting by ω_{i,l_m}^k the local ridge orientation estimated in p_{i,l_m}^k, the relative direction φ_{i,l_m}^k between minutia M_k and the sampling point p_{i,l_m}^k calculated by

$$\varphi_{i,l_m}^k = d\phi(\varphi_k, \varphi_{i,l_m}^k) \tag{3}$$

is independent from the rotation and translation of the fingerprint. The feature vector F_k of a minutia M_k that describes its structure characteristic with global fingerprint orientation field is given by:

$$F_k = \{\{\varphi_{i,l_m}^k\}_{i=1}^{N_{l_m}^k}\}_{m=1}^3 \tag{4}$$
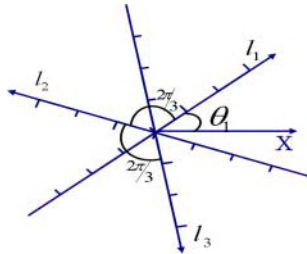
Fig. 1. Sampling points organized on three lines around a minutiae detail

The structure feature vector F_k is invariant to rotation and translation of the fingerprint.

Suppose F_i and F_j are the structure feature vectors of minutia i from input fingerprint and minutia j from template fingerprint, respectively. A similarity level is defined as

$$S(i,j) = \begin{cases} \dfrac{T - \left| F_i - F_j \right|}{T}, & if \left| F_i - F_j \right| < T \\ 0, & otherwise \end{cases}. \qquad (5)$$

where T is the predefined threshold. The similarity level $S(i,j), 0 \le S(i,j) \le 1$, describes a matching certainty level of a structure pair instead of simply matched or not matched. $S(i,j) = 1$ implies a perfect match while $S(i,j) = 0$ implies a total mismatch.

3   Fingerprint Matching

Using the proposed minutia structure, we develop a fingerprint matching algorithm based on point pattern matching. The algorithm receives at the input   two minutia lists and two orientation fields captured from two fingerprint impressions and delivers a matching score that expresses the degree of similarity between the two fingerprint. In order to align two point sets and two orientation fields before calculating the matching score, we need to identify a set of corresponding minutia pairs.

3.1 Corresponding Minutia Identification

The value of the similarity level between minutiae serves to identify corresponding pairs. The best-matched structure pair can be used as a corresponding point pair. Although not all well-matched structures are reliable, our experiments show that the best-matched structure pair of all minutiae structures of template and input fingerprints is very reliable. The best-matched minutiae structure pair (b_1, b_2) is obtained by maximizing the similarity level as:

$$S(b_1, b_2) = \max_{i,j} \left(S(i,j) \right). \qquad (6)$$

3.2 Registration

The registration stage is meant to recover the geometric transformation between the two fingerprint impressions.

In our work, the rigid transformation, i.e., translation vector $\left(t = \left[t_x, t_y \right]^T \right)$ and rotation angle (φ), is recovered by the best-matched structure pair that exhibits the

largest similarity value in equation (6). Let the best-matched minutiae structure pair is denoted by (b_1, b_2), minutia b_1 from the input fingerprint and another b_2 from the template fingerprint. Hence, we have

$$\varphi = D(b_2) - D(b_1), \quad \textit{and} \quad t = P(b_2) - R_\varphi P(b_1) \ . \tag{7}$$

where R_φ denotes the 2×2 operator of counterclockwise rotation with φ and the position and direction of a minutia b are denoted by $P(b) = [x(b), y(b)]^T$ and $D(b)$, respectively. Applying the estimated geometric transformation onto the minutiae from the test fingerprint we obtain the list comprising the registered minutiae. The orientation field from the test fingerprint will be aligned using the estimated transformation.

3.3 Minutia Pairing

Because of various factors that include the presence to local nonlinear deformations and the errors induced by the minutiae extraction algorithm, the corresponding minutiae can't overlap exactly. Consequently, one must allow a certain tolerance between the positions and directions of corresponding minutiae by employing an elastic matching algorithm as proposed in [6], [7], and [10]. Therefore, the matching should be elastic by using a 3-D bounding box Bg in the feature space instead of an exact matching. In our method, the corresponding minutiae pairs are collected among the pairs with largest similarity level values (5), which, also fall in the bounding box Bg .

3.4 Orientation Block Pairing

As the orientation field estimation algorithm proposed in [7], the fingerprint image should be divided into a number of sub-blocks before computing the fingerprint orientation. With the registered orientation field, the procedure to identify the corresponding orientation block pairs is straightforward. Let (B_1, B_2) denote the corresponding orientation block pair, block B_1 from test fingerprint, block B_2 from template fingerprint, respectively. The similarity degree $S(B_1, B_2)$ of the two blocks B_1 and B_2 is calculated as follows:

$$D\phi(B_1, B_2) = |O(B_1) + \varphi - O(B_2)| \ . \tag{8}$$

$$S(B_1, B_2) = \begin{cases} \dfrac{T_1 - D\phi(B_1, B_2)}{T_1}, & \text{if} \quad D\phi(B_1, B_2) < T_1 \\ 0, & \text{others} \end{cases} \tag{9}$$

where φ is computed with equation (7), T_1 is a threshold and the direction of a block B is denoted by $O(B)$.

3.5 Matching Score Computation

With the introduction of our novel minutia structures and registered fingerprint orientation fields the matching score Ms can be determined by both minutiae matching score Mm and orientation field matching score Mo.

Let N_1 and N_2 denote the number of minutiae located inside the intersection of the two fingerprint images for test and template fingerprints, respectively. The minutiae matching score Mm can be calculated according to the following equation

$$Mm = \frac{\sum\limits_{i,j} S(i,j)}{\max\{N_1, N_2\}} \; . \tag{10}$$

where (i, j) is the corresponding minutiae pair, one from test fingerprint and another from template fingerprint, respectively, and $S(i,j)$ is computed according to equation (5).

The orientation field matching score Mo is defined by

$$Mo = \frac{\sum\limits_{B_i, B_j} S(B_i, B_j)}{N} \; . \tag{11}$$

where (B_i, B_j) is the corresponding orientation block pair, one for test fingerprint and another for template fingerprint, respectively, N is the number of overlapped blocks of both fingerprints, and $S(B_i, B_j)$ is determined by equation (9).

The final matching score Ms is computed as follows:

$$Ms = w_m Mm + w_o Mo \; . \tag{12}$$

where (w_m, w_o) is a weight vector that specifies the weight associated with the minutiae matching score Mm and the orientation field matching score Mo.

4   Experimental Results

The experiments reported in this paper have been conducted on the public domain collection of fingerprint images, DB3 in FVC2002. It comprises 800 fingerprint images of size 300×300 pixels captured at a resolution of 500 dpi, from 100 fingers (eight impressions per finger).
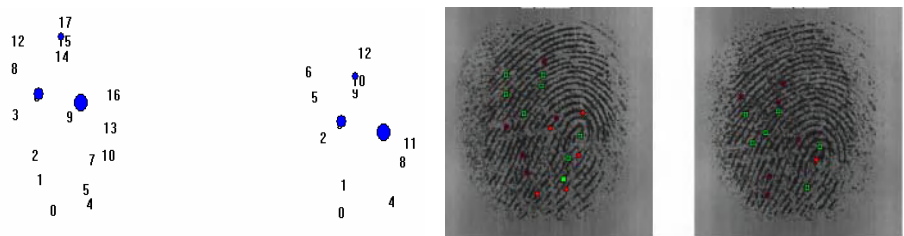
Fig. 2. The 3rd best-matched corresponding minutiae pairs identified by our algorithm. The two point images on the left are the configurations of the minutiae extracted from the two original fingerprint impressions on the right. The more large the size of the corresponding circular regions is, the more similar the corresponding minutiae pair is
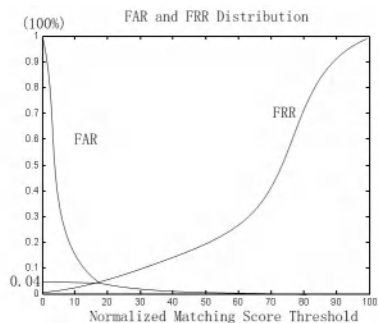
Fig. 3. FAR and FRR distribution with respect to normalized matching score threshold and an ERR of 4% obtained

A set of experiments is conducted in order to evaluate the ability of the proposed algorithm. Fig. 2. shows the 3rd best-matched minutia pairs, where the more large the size of the corresponding circular regions is, the more similar the corresponding minutiae pair is. The two fingerprint images are captured from the same finger 110, labeled as 110_1 and 110_2 in DB3, respectively.

Each fingerprint in the set is matched with the other fingerprints in the set. So, a total of $\frac{1}{2} \times 800 \times 799$ matches have been performed. The distribution of FAR and FRR with respect to the normalized matching score threshold is shown in Fig.4. We achieve an EER of 4%. The processing time for minutiae matching is not fixed. It depends on the number of the minutiae. The average processing time of the $\frac{1}{2} \times 800 \times 799$ matches is 0.012 seconds on a PC Pentium II 450 MHz.

5   Conclusions

In this paper, we describe a novel minutiae structure that allows integration of orientation field information with the minutia details of the fingerprint. The new minutia structure is rotation and translation invariant and captures more global information on fingerprint ridges and furrows pattern. Furthermore, It reduces the

interdependencies between minutia details, which can be missed or erroneously detected by a minutia extraction algorithm. A fingerprint matching algorithm that relies on the proposed minutia structure has been developed, where the matching score is determined by both the matched minutia pairs and the matched orientation block pairs.

The usefulness of the proposed approach is confirmed in the experiments conducted, which show good performance in reliability, accuracy and processing speed.

References

[1]   A.K. Hrechak and J.A. McHugh, "Automatic Fingerprint Recognition Using Structural Matching," Pattern Recognition, vol. 23, no. 8, pp. 893-904, 1990.

[2]   Wahab, S.H. Chin, and E.C. Tan, "Novel Approach to Automated Fingerprint Recognition," IEE Proc. Visual Image Signal Processing, vol. 145, no.3, pp. 160-166, 1998.

[3]   Z. Chen, C.H. Kuo, "a Topology-Based Matching Algorithm for Fingerprint Authentication", Proc. of 25th Annual IEEE International Carnahan Conference on Security Technology, pp. 84-87, 1991.

[4]   X. Jiang and W. -Y. Yau, "Fingerprint Minutiae Matching Based on the Local and Global Structures," Proc. 15th Int'l Conf. Pattern Recognition, vol. 2, pp. 1038-1041, 2000.

[5]   A.K. Jain, L. Hong, S. Pankanti, and R. Bolle, "An Identity-Authentication System Using Fingerprints," Proc, IEEE, vol. 85, no.9, pp. 1365-1388, 1997.

[6]   N.K. Ratha, K. Karu, S.Chen, and A.K. Jain, "A Real-Time Matching System for Large Fingerprint Database," IEEE Trans. Pattern Analysis and Machine Intelligence, vol.18, no.8, pp. 799-813, Aug. 1996.

[7]   A.K. Jain, L. Hong, and R.Bolle, "On-Line Fingerprint Verification," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no.4, pp. 302-313, Apr. 1997.

[8]   Almansa and L. Cohen," Fingerprint Image Matching by Minimization of a Thin-Plate Energy Using a Two-Step Algorithm with Auxiliary variables," Fifth IEEE Workshop on Applications of Computer Vision, pp. 35-40, Dec. 04-06, 2000, Palm Springs, California, USA.

[9]   M. Bazen and S. H.Gerez, "Thin-Plate Spline Modeling of Elastic Deformations in Fingerprints", Proc. 3rd IEEE Benelux Signal Processing Symposium, pp. 1-4, 2002.

[10]   X. Jiang and W.-Y. Yau, "Fingerprint Minutiae Matching Based on the Local and Global Structures," Proc. 15th Int'l Conf. Pattern Recognition, vol.2, pp. 1038-1041, 2000.

[11]   Ross, A. Jain, and J. Reisman, "A Hybrid Fingerprint Matcher," Pattern Recognition, vol.36, pp. 1661-1673,2003.

[12]   H. Chui and A. Rangarajan, "A New Point Matching Algorithm for Non-Rigid Registration," Computer Vision and Image Understanding, vol.89 pp. 114-141, 2003.

A Systematic Approach for Feature Extraction in Fingerprint Images

Sharat Chikkerur, Chaohang Wu, and Venu Govindaraju

Center for Unified Biometrics and Sensors (CUBS)
University at Buffalo, NY, U.S.A
{ssc5,cwu3,govind}@cedar.buffalo.edu

Abstract. Fingerprint image enhancement and feature extraction are the most important stages in fingerprint verification. The robustness of the system depends entirely upon its ability to enhance bad quality images and reliably extract minutiae from it. In this paper we propose two original algorithms for the extraction of global and local features in fingerprint images. A global feature extraction and enhancement algorithm based on Fourier analysis that is simultaneously able to extract local ridge orientation, ridge frequency and ridge quality measure is proposed. A novel algorithm for minutiae feature extraction and post processing based on chain coded contour following is also presented.

1 Introduction

This paper introduces new algorithms for fingerprint image enhancement, and feature extraction from fingerprint images. Fingerprints are characterized by global and local features. The global features include the ridge orientation map, core and delta locations. The minutiae points form the local features. The estimation of local ridge orientation and the local ridge frequency play an important role in all subsequent stages of a fingerprint verification system [14]. The ridge orientation map is used in image enhancement [14,8], detection of singular points [16], minutiae post processing and fingerprint classification [7]. The map of local ridge frequency has been found to be useful in fingerprint image enhancement [8].

There have been several approaches to estimate the orientation map of a fingerprint image. These include the use of gradients [8], template comparison [7], and ridge projection based methods [14]. The orientation estimation obtained by these methods is noisy and has to be processed further. The post processing methods can be based on vector averaging [6], relaxation methods [14], or mathematical orientation models [15]. In particular, the orientation models depend on reliable detection of the core and delta points in the fingerprint image. However, many of the methods used for singular point extraction [7,16,1] depend on a reliable orientation map, therefore requiring that the orientation map be estimated accurately to begin with.

The local ridge frequency indicates the average inter ridge distance within a block. This can be estimated based on the projection sum [8], or variation of gray level in different directions [11]. These two methods also depend upon the reliable extraction of the local ridge orientation.
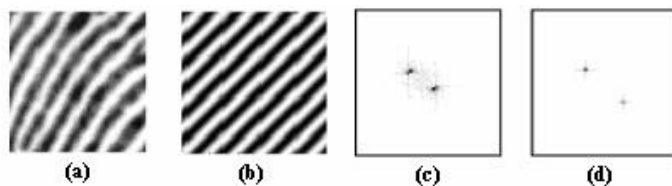
Fig. 1. (a) Local region in a fingerprint image (b) Surface wave approximation (c d) Fourier spectrum of Fig. 2.a, Fig. 2b. The symmetric nature of the Fourier spectrum arrives from the properties of the Fourier transform for real signals [3]

The robustness of the verification system depends on its ability to enhance poor quality images. General-purpose image processing algorithms are not very useful in this regard but serve as a preprocessing step in the overall enhancement scheme [10]. A majority of the techniques are based on the use of contextual filters whose parameters depend on the local ridge frequency and orientation. The filters themselves may be spatial [8,4] or based on Fourier domain analysis [14,18].

Feature extraction algorithms can be categorized as those based on gray level and those dealing with binarized images. Approaches working with binary images gray proceed with detection of the minutiae after thinning the binary image. The binarization approaches include peak detection [13], adaptive and optimal thersholding. Approaches working with gray level images are mostly based on ridge following [12].

The rest of the paper is organized as follows: Section 2 presents a scheme to extract local ridge orientation and frequency using Fourier domain analysis. Segmentation and enhancement schemes are also introduced. Section 3 gives details of the minutiae extraction and post processing methods. Section 4 presents an objective evaluation of the algorithm. The summary of the paper is outlined in Section 5.

2   Global Feature Extraction

2.1 Fourier Domain Analysis

With the exception of the singularities such as core and delta any local region in the fingerprint image has a consistent orientation and frequency. Therefore, the local region can be modeled as a surface wave that is characterized completely by its orientation ϕ and frequency f. The parameters of the surface wave (f, ϕ) may be easily obtained from its Fourier spectrum that consists of two impulses whose distance from the origin indicates the frequency and its angular location indicates the orientation of the wave.

2.2 Directional Field Estimation

In reality the surface wave model is only an approximation, and the Fourier spectrum of the real fingerprint images is characterized by a distribution of energies across all frequencies and orientations. To approximate each block by a single orientation and frequency, a probabilistic approximation is used. We can represent the Fourier

spectrum in polar form as F(r, φ). Using this, we can define a probability density function f(r, φ) and the marginal density functions f(φ),f(r) as

$$f(r,\phi) = \frac{|F(r,\phi)|^2}{\iint\limits_{r\ \phi} |F(r,\phi)|^2\, d\phi dr} \tag{1}$$

$$f(\phi) = \int\limits_r f(r,\phi)dr \ ,\ f(r) = \int\limits_\phi f(r,\phi)d\phi \tag{2}$$

We assume that the orientation φ is a random variable that has the probability density function f (φ). The expected value of the orientation may then be obtained by

$$E\{\phi\} = \int\limits_\phi \phi \cdot f(\phi)d\phi \tag{3}$$

2.3 Ridge Frequency Estimation

The average ridge frequency is estimated in a manner similar to the ridge orientation. We can assume the ridge frequency to be a random variable with the probability density function *f(r)* as in Eqn. (2). The expected value of the ridge frequency is given by

$$E\{r\} = \int\limits_r r \cdot f(r)dr \tag{4}$$

The frequency map so obtained is smoothened by applying a 3x3 gaussian mask.

2.4 Energy Map and Segmentation

The fingerprint image may be easily segmented based on the observation that the surface wave model does not hold in regions where ridges do not exist. In the areas of background and noisy regions, there is very little structure and hence very little energy content in the Fourier spectrum. We define an energy image E(x,y) (Eqn. 5) where each value indicates the energy content of the corresponding block. The fingerprint region may be differentiated from the background by thersholding the energy image. We take the logarithm values of the energy to obtain a linear scale. Fig 4 illustrates the energy map obtained by this method.

$$E = \sum_u \sum_v |F(u,v)|^2 \tag{5}$$

2.5 Enhancement

We propose a Fourier domain based block-wise contextual filter approach for enhancing fingerprint images. The image is divided into 16 x 16 overlapping blocks that is filtered in the Fourier domain by a frequency and orientation selective filter whose parameters are based on the estimated local ridge orientation and frequency. Block-wise approaches have problems around the singularities where direction of the ridges cannot be approximated by a single value. The bandwidth of the directional filter has to be increased around these regions. We propose to use the directional histogram obtained in the estimation of the orientation image for this purpose. We assume that f(ϕ) is unimodal and centered around E{ϕ} and define the bandwidth as the angular extent where P{|ϕ-E{ϕ}|<ϕ_{BW}}= 0.5. Thus, the angular bandwidth of the filter adapts itself in regions of high curvature. The filters are defined as specified in [14] and are given by

$$H_r(r) = \sqrt{\frac{(rr_{BW})^{2n}}{(rr_{BW})^{2n} + (r^2 - r_{BW}^2)^{2n}}}$$

$$H_\phi(\phi) = \begin{cases} \cos^2 \dfrac{\pi(\phi - \phi_C)}{2\phi_{BW}} & \text{if } |\phi - \phi_c| \leq \phi_{BW} \\ \qquad 0 & \text{otherwise} \end{cases} \qquad (6)$$

r_{BW} : radial bandwidth, ϕ_{BW} : angular bandwidth, ϕ_c : mean orientation.

3   Chain Code Based Minutiae Detection and Post Processing

Commonly used minutiae extraction algorithms that are based on thinning are iterative, computationally expensive and produce artifacts such as spurs and bridges. We propose a chain coded contour following method for this purpose. Chain codes are a loss less representation of contours and yield a wide range of information about the contour such as curvature, direction, length etc [5]. As the contour of the ridges is traced consistently in a counter-clockwise direction, the minutiae points are encountered as locations where the contour has a *significant* turn. Specifically, the ridge end occurs as significant left turn and the bifurcation as a significant right turn in the contour. Analytically the turning direction may be determined by considering the sign of the cross product of the incoming and outgoing vectors at each point. The product is right handed if the sign of Eqn.(7) is positive and left handed if the sign is negative. The turn is termed significant only if $x_1 y_1 + x_2 y_2 <= T$. The threshold T is chosen to have a small value. In practice a group of points along the turn satisfy this condition. We define the minutia point as the center of this group.

$$\text{sgn}(\ \vec{P}_{in} \times \vec{P}_{out}\ ) = \text{sgn}(\ x_1 y_2 - x_2 y_1) \qquad (7)$$
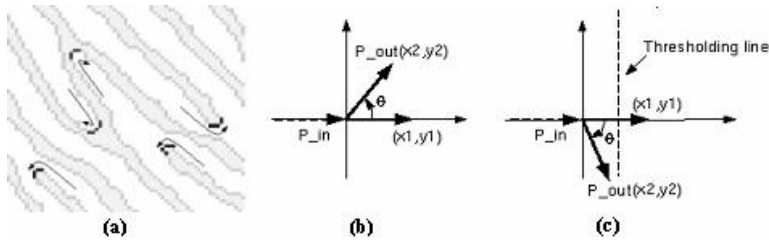
Fig. 2. (a) Minutiae marked by significant turn in the contour (b) Left turn (b) Right turn
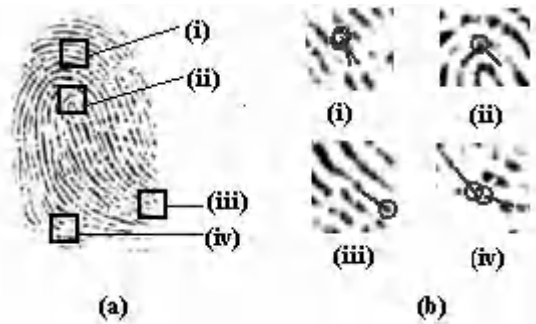
Fig. 3. Post processing: (a) Fingerprint image with locations of spurious minutiae marked (b) Types of spurious minutiae removed by applying heuristic rules (i-iv)

The feature extraction results in two forms of errors. The enhancement process may introduce artifacts that are detected as spurious minutia or may miss some of the genuine minutiae (Fig. 3.). We use heuristics based on the structural properties of the minutiae to eliminate the false features. Similar approaches are outlined in [10]. We use the following rules to eliminate false minutiae (i) We merge the minutiae that are within a certain distance of each other and have similar angles. This is to merge the false positives that arise out of discontinuities along the significant turn of the contour. (ii) If the direction of the minutiae is not consistent with the local ridge orientation, then it is discarded. This removes minutiae that arise out of noise in the contour. (iii) We discard all minutiae that are within a distance of the border of the fingerprint area. The border is determined based on the energy map. This rule removes spurious minutiae that occur along the border of the fingerprint image (iv) We remove the pair of opposing minutiae that are within a certain distance of each other. This rule removes the minutiae that occur at either ends of a ridge break.

4   Testing and Evaluation

The efficiency of the feature extraction can be measured by how well it meets the objective of detecting the minutiae features from the original image. We use the evaluation method proposed in [14]. We measure two quantities namely "Sensitivity" and "Specificity" to evaluate the algorithm. The sensitivity and specificity indicates the ability of the algorithm to detect the true minutiae and reject false minutiae

respectively. We manually identified the minutiae features on a subset of FVC2002 DB1 [2] database to establish the ground truth of the minutiae. The minutiae features were then detected using our algorithm and the sensitivity and specificity were measured over the entire database. The algorithm was executed on a PC with AMD Athlon 1.1GHz processor running Windows XP. The algorithm takes an average of 0.98s over the test images.

4.1 Results
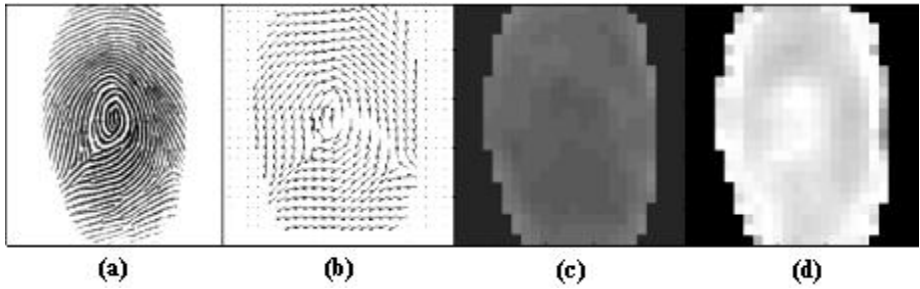
Fig. 4. (a) Original print (b) Orientation Image (c) Energy map (The brighter regions indicate regions with higher energy) (d) Frequency Image (Bright regions indicate higher frequencies)

Fig. 5. Results of the contextual filtering algorithm (a) Good quality print. (b) Medium quality print (c) Bad quality dry print

Table 1. Performance over FVC2002 (DB1) database

	Sensitivity(%)	Specificity(%)
Mean Value	79.40	85.29
Std Deviation	7.86	12.91

5   Conclusion

We have proposed a unified frequency domain analysis based algorithm for the enhancement and extraction of global features from fingerprint images. The enhancement algorithm uses the full context information obtained from the Fourier domain analysis of the fingerprint image. We have also proposed a novel algorithm for the extraction of minutiae points from the fingerprint image using the chain coded contour ridge following method. Heuristic rules specifically suitable for our algorithm have been presented.

References

[1]   Bazen and Gerez, "Extraction of Singular Points from Directional Fields of Fingerprints", Annual CTIT Workshop, Enschede, The Netherlands, 2001.
[2]   Fingerprint Verification Competition, http://bias.csr.unibo.it/fvc2002/.
[3]   Gonzalez R. C and R. E. Woods ,"Digital Image Processing", Prentice Hall, Upper Saddle River, NJ, 2002.
[4]   Gormann L. and J. V. Nickerson, "An Approach to Fingerprint Filter Design", Patt. Recogn., Vol. 22, No. 1, pp. 29-38, 1989.
[5]   Govindaraju V., Z. Shi and J. Schneider, "Feature Extraction Using a Chaincoded Contour Representation", International Conference on Audio and Video Based Biometric Person Authentication", Surrey, UK, 2003.
[6]   Kaas M. and A. Witkin, "Analyzing Oriented Patterns", Comp. Vision Graphics Image Process. Vol. 37, No. 4, pp. 362-385, 1987.
[7]   Kawagoe M., A. Tojo, "Fingerprint Pattern Classification", Pattern Recogntion, Vol. 17, No 3, pp 295-303,1987.
[8]   Ling H.,Y.Wan and A. K. Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluation", PAMI, Vol. 20, No. 8, pp. 777-789., Aug. 1998.
[9]   Madhavanth S., Kim, and V. Govindaraju,"Chaincode Processing for Hand-written Word Recogntion", IEEE Trans. On Pattern Analysis and Mach. Intell 19, pp. 27-40, 1997.
[10]   Maltoni D., D. Maio, A. K. Jain, S. Prabhakar, "Hand book of Fingerprint Recognition", Springer-Verlag, 2003.
[11]   Maio D. and D. Maltoni, "Neural Network Based Minutiae Filtering in Fingerprints″, Proc. 14th International Conf. Pattern Recognition, Brisbane, Australia, pp. 1654-1658, 1998.
[12]   Maio D. and D. Maltoni "Direct Gray Scale Minutiae Detection in Fingerprints", IEEE Trans. On Patt. Analysis and Machine Intell. Vol.19, No.1, 1997.
[13]   Ratha N. K. , S. Y. Chen and A. K. Jain,"Adaptive Flow Orientation Based Feature Extraction in Fingerprint Images", Patt .Recogn. Vol. 28, No. 11, pp. 1657-1672, 1995.
[14]   Sherlock B. G., D. M. Monro, K. Millard, "Fingerprint Enhancement by Directional Fourier Filtering", IEEE Proc.Vis. Image Signal Process., Vol 141., No 2, pp. 87-94, April 1994.
[15]   Sherlock B. G, D. M. Monro, "A Model for Interpreting Fingerprint Topology", Pattern Recognition, Vol. 26, No. 7, pp 1047-1055, 1993.
[16]   Srinivasan V.S. and N. N. Murthy, "Detection of Singular Points in Fingerprint Images", Pattern Recognition, Vol. 25, No. 2, pp. 139-153,1992.
[17]   Viscaya, Gerhardt,"A Nonlinear Orientation Model for Global Description of Fingerprints", Pattern Recognition, Vol.29, No.7, pp 1221-1231, 1996.
[18]   Watson C. I., G. T. Candela and P. J. Grother, "Comparison of FFT Fingerprint Filtering Methods for Neural Network Classification", NISTIR 5493, 1994.

Fingerprint Minutiae Matching
with Orientation and Ridge

Jiangang Cheng[1], Jie Tian[1,2], and Hong Chen[1]

[1] Biometrics Research Group, Key Laboratory of Complex Systems and Intelligence Science
Institute of Automation, Chinese Academy of Science, P.O.Box 2728, 100080, China
[2] Graduate School of the Chinese Academy of Science Beijing, 100039, China
`jie.tian@mail.ia.ac.cn`

Abstract. An important step in automatic fingerprint identification system (AFIS) is fingerprint matching. The task of fingerprint matching is to verify whether two fingerprints are coming from same finger. In this paper we detail and discuss the fingerprint matching algorithm. A minutia matching algorithm is proposed which modified the algorithm presented by Jain et al. In this algorithm, in order to reduce the effect of noise and false minutiae, block orientation and ridge information are introduced into the minutiae-based matching algorithm in a simple but reliable way. Ridge information which is some sampled points in the ridge are used to align two fingerprints, in order to avoid misalign two fingerprints, we use block orientation to correct the ridge alignment. Experiments on the database FVC2002 show the performance.

1   Introduction

Fingerprint recognition as a reliable method among biometric feature recognition technology is widely applied in personal identification for the purposes of high degree of security. Although fingerprint recognition is usually associated with criminal identification and police work, it has become more popular in civilian application, such as access control, personal computer protection, e-commerce. In order to attain high performance of the recognition, many abroad researchers have been proposed many fingerprint recognition algorithms, such as: graph-based or structure-based fingerprint matching algorithms [1,2,3], minutiae-based [4,5,6], special filters [7] or multiple matchers fingerprint recognition algorithms [8,9].

   Graph-based matching [1] treats the fingerprint as an attributed graph, complex algorithms, such as graph isomorphism, have to be adopted in the fingerprint recognition.

   Minutiae-based matching uses the minutiae to describe the fingerprint feature, so matching two fingerprints can be completed by the point pattern matching. This approach is very popular for fingerprint matching. In this method, a fingerprint image is usually processed by a thinning algorithm to obtain feature data, this feature data commonly are the ridge endings and ridge bifurcation. Minutiae-based matching depends on the local feature of the fingerprint and cann't be used as coarse matching such as fingerprint classification. Furthermore, detail matching using this methodology is greatly influenced by the interference of noise and distortion in image

processing. There are some papers to discuss the fingerprint distortion module [13, 14].

Special filter or texture-based method [7] for fingerprint matching uses a bank of Gabor filters to capture both local and global details in a fingerprint, which are called FingerCode by the authors.

In order to attain the high performance, some researchers also use multiple matcher in fingerprint recognition algorithm [8,9,12]. This method used each matcher to get the matching score and then use the probability method to make a final decision whether two fingerprints come from same finger. This approach really improves the performance.

In this paper, we introduce some global information, block orientation and ridges, into the minutiae-based method. The fingerprint is an oriented pattern image. The rest of this paper is organized as follows. In the section 2, is the feature representation. Section 3 is devoted to the methods of fingerprint matching algorithm. The performances of fingerprint identification are listed in section 4. At last, we discuss our algorithm and future work in section 5.

2   Feature Representation

Definition 1 Sampled Ridge**:** The ridge that be sampled is the ridge corresponding to direction of the minutia, and sampled ridge is the collection of all sampled points on the ridge. The interval between two sampled points is S. The maximum number of the sampled points for each minutia is K. The figure 1 illustrates the sampled ridge. For every sampled point, we record the distance $R(d_i) = | MP_i |$ and the angle α_i. d_i is the distance between the sampled point P_i and the corresponding minutia M. $R(\alpha_i)$ is the angle of vector MP_i. The reason we record the ridge is in the section 5.

If the sampled distance is larger, more information about the ridge is abstracted or lost. If sampled distance is small, features will be detected and we will be able to reconstruct them, but redundant information such as noise will be included. So the sampled interval S and K are very important parameters in our algorithm. Our sampled ridge method is similar with the Dario et al [16] method following the ridge that they use the polygonal to represent the ridge.
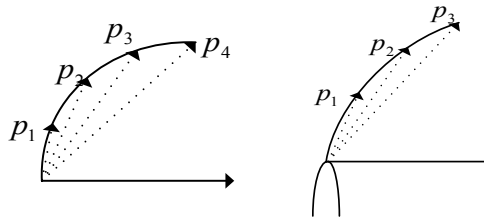
Fig. 1. Sampled Ridges: At the left, for Ridge ending M, the sampled points are $P_1, P_2\ P_3\ P_4$; at the right the sampled points of bifurcation are $P_1, P_2\ P_3$

At last, for a fingerprint image, recorded information is as following:

The minutiae x, y coordinates, orientation θ, d_i, α_i ($i = 1,2,\cdots,K$) associated to every minutia, and the block orientation field **BO**.

3   Fingerprint Matching

The purpose of the fingerprint matching is to determine whether two fingerprints are from the same finger. For this goal, we firstly align the input fingerprint with the template fingerprint that is represented by its minutiae and sampled ridge. After aligning two fingerprints we compute the matching score of two fingerprints. Because we do not have any prior knowledge about what is the relative position of two fingerprints and whether they are similar, a robust aligning two fingerprints method is needed.

Referring to the matching phases in [4], [5], we decompose our algorithm into two hierarchies as follow:

First stage:
      Computing the similarity of two ridges coming from two fingerprints
      Computing the similarity of orientation field
      Getting the rotation parameter and candidates as the reference points
Second stage:
      Computing matching score

3.1 Sampled Ridge Matching

Let

$$P = \{(x_1^P, y_1^P, \theta_1^P), \cdots, (x_M^P, y_M^P, \theta_M^P)\}$$

represent minutiae of the fingerprint in the template databases.

$$Q = \{(x_1^Q, y_1^Q, \theta_1^Q), \cdots, (x_N^Q, y_N^Q, \theta_N^Q)\}$$

represent the minutiae of the input fingerprint.

In this step, we attempt to acquire the best registration parameters of two fingerprints, assuming they are from the same finger. No matter whether the two fingerprints are identical or not, the following similarity transformation is performed:

$$T_{\Delta x, \Delta y, \Delta \theta}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos(\Delta\theta) & -\sin(\Delta\theta) \\ \sin(\Delta\theta) & \cos(\Delta\theta) \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \tag{1}$$

where $(\Delta x, \Delta y, \Delta \theta)$ denotes a set of similarity transformation parameters, x and y are the minutia coordinates. Ratha et al. [4] proposed a general algorithm of minutia-pattern matching by means of Hough transformation. So the problem of the minutia pattern matching was converted to detect the highest peak in the Hough space of the transformation parameters by discretized the set of all possible transformation

parameters and meanwhile computing the corresponding matching score for each transformation. However, due to the presence of noise and deformation, the input minutiae cannot always be aligned exactly with respect to those of the template using the parameters at the peak in the Hough space. In order to reliably align two fingerprints, we introduce the sampled ridge to select reference points. Comparing with the method in [5], our aligned method is simple but effective.

Definition 2 Similarity Measure of two sampled ridge: Let R is the ridge of the minutiae P_i, r is the ridge of the minutiae Q_i, the similarity measurement of R and r is defined following:

$$Diff(P_i, Q_j) = \frac{1}{L}\left(\alpha \sum_{i=0}^{L} |R(d_i) - r(d_i)| + \beta \sum_{i=0}^{L} |R(\alpha_i) - r(\alpha_i)|\right) \qquad (2)$$

L is the number of the sampled points in the ridge, α and β are constants.

In our system, two parameters affect the similarity of two sampled ride, one is the interval between two sampled points S, the other is the count of sampled points K. We discuss the problem in the section 6.

If the two ridges are similar, we get the rotation parameter $\Delta\theta = \theta_i^P - \theta_j^Q$. After $M*N$ times comparing ridges, we get corresponding ridges and the rotation $\Delta\theta$ which corresponds to have maximum number of ridge similarity. Using them we could compute the registration parameters.

In order to avoiding misaligning two fingers, we use the block orientation field BO^P and BO^Q. It is the first step in [4] to detect the core or delta point using block orientation field to classify the fingerprints in the database in order to reduce the retrieval space. Generally, without the phase of classification to fingerprints in an AFIS, there will be more false accepted matching. But the classification based on the core and delta points is affected by the noise and the classification accuracy is not high. So we introduce the block orientation field to reduce the false accepted rate and to avoid the noise effect, at the same time avoid misaligning two ridges.

Block Matching Algorithm(BMA)
For each pair $P_i \leftrightarrow Q_j$ in F

  a. Compute the translation parameters:
$$\Delta x = x_i^P - x_j^Q$$
$$\Delta y = y_i^P - y_j^Q$$

  b. Using the equation (1) to translate the BO^Q according to Δx, Δy and θ_0, the translated block orientation is also noted as BO^Q.

  c. Compute the similarity:
$$\text{For } (m, n) \in BO^P \cap BO^Q$$

$$S(m,n) =$$

$$\begin{cases} \left| \theta_T(m,n) - \theta_{I'}(m,n) \right| & if \ \left| \theta_I(m,n) - \theta_{T'}(m,n) \right| < 90 \\ 180 - \left| \theta_T(m,n) - \theta_{I'}(m,n) \right| & other \end{cases} \quad (3)$$

Compute the similarity of RO^P and RO^Q

$$S_{BO}(P_i, Q_j) =$$

$$\begin{cases} \left(C - \dfrac{\sum\limits_{(m,n)\in O} S(m,n)}{|O|} \right) \exp\left(\left(1 - \dfrac{|O|}{N} \right) \Big/ \sigma \right) & if \ \dfrac{\sum\limits_{(i,j)\in O} S(i,j)}{|O|} < C \\ 0 & other \end{cases} \quad (4)$$

$O = BO^P \cap BO^Q$, $N = \max(|BO^P|, |BO^Q|)$, σ and C are two constants.

d. If $S_{BO}(P_i, Q_j) > Threshold$, delete the pair $P_i \leftrightarrow Q_j$ from set F.

From *BMA* algorithm, we get the rotation parameter and matching minutiae set F as the aligned candidates.

3.2 Computation Matching Score

After getting registration candidate points, the number of matched pair minutiae can be obtained. If two minutiae fall into a same tolerance box after registration, they are defined as matched minutiae. In [4], Ratha *e. al.* defines a fixed tolerance box. In our system we use a changeable tolerance box to make our algorithm more robust to nonlinear deformation between fingerprint images. The changeable tolerance box definition could be found in our published paper [10]-[11].

Now we compute the matching score:

1. For each pair $P_i \leftrightarrow Q_j$ in F, if F is empty, go to 5

2. Take P_i and Q_j as reference minutia. Convert each minutia point in the template minutiae set and the input minutiae set to the polar coordinate system with respect to the corresponding reference minutia and rotation parameter θ_0

3. Represent the template and the input minutiae in the polar coordinate system as symbolic strings by sorting each minutia in the increasing order of radial angles:

$$P_i^s = \left((r_1^p, e_1^p, \theta_1^p)^T, \dots, (r_M^p, e_M^p, \theta_M^p)^T \right)$$

$$Q_j^s = \left((r_1^Q, e_1^Q, \theta_1^Q)^T, ..., (r_N^Q, e_N^Q, \theta_N^Q)^T \right)$$

4. Decide which pair minutiae are located in same tolerance box , if they are consistent or similar:

<div align="center">score[i][j] += 1;</div>

5. Find the maximum value of *score[i][j]* and use it as the matching score of the input and template minutiae set.

6. Let the maximum score be $score[i_0][j_0]$, compute the $S_{BO}(P_{i_0}, Q_{j_0})$, the final matching score is :

$$\text{Score} = \frac{100 * score[i_0][j_0]}{\max(N, M)} * S_{BO}(P_{i_0}, Q_{j_0}) \tag{5}$$

If the matching score is higher than a threshold value, the input image is considered to come from the same finger as the template image, else we would consider these two images as coming from different fingers.

4   Experimental Results

The performance of the algorithm taking part in the FVC2002 competition is evaluated on images from four different fingerprint databases, three of these databases are acquired by various sensors, low-cost and high-quality, optical and capacitive.

Before we submit our algorithm, we must estimate two important parameters: S, K. FVC2002 provides 80 fingerprints in each database to be available to the participants to allow parameters tuning before the submission of the algorithm. So we use statistical method to estimate the parameters on the 80 fingerprints for each database. At last, we approximate S about 8, and K to about 10. So in our algorithm, two parameters are fixed.

Based on above algorithm we have developed a software system including enrolling fingerprints and matching fingerprint, to take part in the FVC2002 competition. The results are tested by organizer of the FVC2002.

5   Conclusion

In this paper, we introduce our automatic fingerprint verification algorithm. The orientation is our algorithm basement. Our method integrates the advantages of minutiae-based and texture-based methods, and can improve the algorithm performance and reliability. As we use orientation field to simulate the texture of the image, and the orientation is the middle result enhancing image, it is faster than the gray value texture of the image in the literature. But we must point out our algorithm also have some disadvantages comparing with other algorithm, for example we

occupy much memory due to the block orientation field. From the results on the FVC2002 database, our recognition performance is not very ideal. From our experience, our algorithm can attain higher recognition performance than the results on the FVC2002 databases.

Acknowledgement

Reference

[1]   1 D.K. Isenor, S.G. Zaky, Fingerprint Identification Using Graph Matching, Pattern Recognition, Vol. 19, No.2, pp.113-122, 1986.
[2]   2 S.Gold, A.Rangarajan. A Gradulated Assignment Algorithm for Graph Matching, IEEE Trans on Pattern Analysis and Machine Intelligence, Vol.18, No.4, pp.377-388, 1996.
[3]   3 A.K.Hrechak, J.A.McHugh, Automated Fingerprint Recognition Using Structural Matching, Pattern Recognition, Vol.23, No.8, pp.893-904, 1990.
[4]   4 N.K.Ratha, K.Karu, S.Chen, and A.K.Jain, A Real-time Matching System for large Fingerprint Databases, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.18,No.8,pp.799-813, 1996.
[5]   A.K.Jain, L.Hong, and R.,Bolle, On-Line Fingerprint Verification, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.19, No.4, pp.302-314, 1997.
[6]   Z.Miklós, Vajna, A Fingerpri nt Verification System Based on Triangular Matching and Dynamic Time Warping, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.22,No.11, pp.1266-1276, 2000.
[7]   A.K.Jain, S.Prabhaker, L.Hong and S.Pankanti, Filterbank-based Fingerprint Matching, IEEE Trans. Image Processing,Vol.9, No.5, pp.846-859, 2000.
[8]   A.K.Jain, S. Prabhakaker, S.Chen, Combining Multiple Matchers for a High Security Fingerprint Verification System, Pattern Recognition Letters, Vol.20, No.11-13, pp.1371-1379, 1999.
[9]   A.K.Jain, A.Ross, S.Prabhakar, Fingerprint Matching Using Minutiae and Texture Features. International Conference on Image Processing, Thessaloniki, Greece, Oct., 2001.
[10]   Luo Xiping, Tian Jie, Image Enhancement and Minutia Matching Algorithm in Automated Fingerrpint Identification System, Journal of software, vol. 13 No5.,2002.
[11]   Luo Xiping, Tian jie, A Minutia Matching algorithm in Fingerprint Verification, proceeding of the 15th International Conference on Pattern Recognition(ICPR 2000), V.4, P833-836, Barcelona, Spain.
[12]   Gian luca Marcialis, Fabio Roli, Paolo Loddo, Fusion of Multiple Matchers for Fingerprint Verification, Proceeding of the Workshop on Machine Vision and Perception, Siena, Italy, 2002.

[13]   R. Cappelli, D. Maio and D. Maltoni, Modelling Plastic Distortion in Fingerprint Images, in proceedings Second International Conference on Advances in Pattern Recognition (ICAPR2001), Rio de Janeiro, pp.369-376, March 2001.
[14]   Asker M. Bazen, Sabih H. Gerez, Elastic Minutiae Matching by Means of Thin-Plate Spline Models, proceeding of the 16th International Conference on Pattern Recognition (ICPR 2002), pp985-pp988, Quebec, Canada, 2002.

A Hierarchy Approach for Singular Point Detection in Fingerprint Images

Furong Wang, Xuan Zou, Yuan Luo, and Jiaxi Hu

Center of Communication Software & Switching Technology
Department of Electronics and Information Engineering
Huazhong Univ. of Sci. & Tech, Wuhan, 430074, P. R. China
wangfurong@mail.hust.edu.cn

Abstract. In this paper we present a hierarchy approach for singular points (SPs) detection in fingerprint images. In the coarse location stage those image blocks containing SPs are located by calculating Poincare Index in the block orientation field, the types of the SPs are determined at the same time, and orientations of cores are computed in a novel way. Then SPs are extracted according to local intensity gradient in a pixel level in the precise location stage. Experiment results show that the utilization of both high-level info and local info in this hierarchy approach leads to not only robust performance on SP detection but also good time efficiency.

1 Introduction

Although many other biometrics features, such as face, voice, ear, gait and etc, have been studied in recent years, fingerprint has been the most widely used feature in biometrics-based personal verifications/authentications. The advantages of fingerprint technology include the low cost for fingerprint acquisition, and the relatively good performance of fingerprint biometrics. However, efforts are being made to achieve even better performance.

A fingerprint image consists of flow-like oriented pattern, and the orientation field (in a block level, or in a pixel level) forms the basic structure of the fingerprint image in different scales. SPs (core and delta, see Fig 1) denote the discontinuities in the orientation field in fingerprint images, and are thus considered important features of the structure. Therefore SPs have been used in fingerprint classification [1][10], fingerprint matching [2], and even serve as the "origins" of an intrinsic coordinate system for minutiae description [3].

The methods for detecting SPs have been widely exploited in literature. G. A. Drets and H. G. Lijienstrom presented an approach in which sliding neural networks are used [4]. P. Perona detected the location of SPs by local energy [5]. M. Tico and P. Kuosmanen described a method based on multi-resolution representation of both orientation field and certainty level in [6]. While most of the above methods provide a continuous output indicating how local orientation field resemble SP [7], method based on Poincare Index (PI)[1][10] can produce a Boolean output indicating the

existence/non-existence of SP. In [1], PI was computed in a 2*2 rectangle with the interested point at the upper left corner. And in a later work [10], a closed square curve with an empirically optimal length of 25 Pixels is considered when computing PI.

Asker M. Bazen and Sabih H. Gerez [7] proposed a PCA-based pixel directional field computation method and proved that it provided same result with traditional "average square-gradient" method. The coherence of the pixel directional field is computed to segment fingerprint region from background. SPs are detected in the high-resolution directional field, with their orientations estimated. However, it is a computationally expensive task to locate SPs in pixel resolution by calculating PI, because one has to compute the orientations for all pixels in the fingerprint image, then calculate PI for every pixel.

Fig. 1. Fingerprint images with different SPs (Cores marked by squares, Deltas by triangles)

P. Ramo, M. Tico and etc [8] proposed an SP detection algorithm, in which SPs are located by detecting those points with both zero horizontal gradients and zero vertical gradients. There is no need of calculating the orientation field, so SPs can be detected much faster than using algorithms in [1]. However, since this algorithm in [8] is greatly based on local information, the detection results often include SPs with false location or type in a noisy area. As a remedy P. Ramo and etc discard improper SPs if another SP has been found in the adjacent area, but such a remedy is not good enough because SPs with different types may exist in an adjacent area as the results of noise influence. In this situation, the discard of SPs without selection may cause type problem. We avoid this by determine the SP type in an interested area before we detect its precise location. See Fig 5.

We proposed in this paper a hierarchy approach for SPs detection, in which both the information in a large scale and local information are used. The proposed approach will be described in detail in section 2, and experiment result will be shown in section 3. In section 4 a summary of the whole paper is presented.

2   SPs Detection in a Hierarchy Approach

Before SPs detection, the original image is normalized to an image with a given mean and variance. SPs detection consists of two stages. In the first one, we locate every block containing a SP by calculating PI in block orient field, the type of SP in each area is determined and if it's a core, the orientation of it will be calculated. The use of

a high level info (block orientation) instead of local info ensures both the robust detection and the time efficiency. Then, in the second stage, an improved version of SPs detection algorithm described in [8] is employed to detect SPs with pixel resolution in the above areas. The detection of SPs in much smaller areas can be completed much faster than the detection in a whole image.

The two stages are described below in details:

2.1 Stage 1: Coarse Location, Type Determination, and Orientation Estimation

Block Orient Field Computation. To achieve more accuracy, the Least Squared Estimation (adopted in [7] [9] [10] [11]) of orientation vector is used to obtain a block orientation field. Subsequently the foreground of the fingerprint is determined in a block level.

Poincare Index Computing. PI is calculated for each block in the foreground of the image. PI stands for the accumulation of orientation changes in a counter-clockwise close contour around an interested area (pixel or block). The value of PI will be π if there exists a core in this area, $-\pi$ if there is a delta, and 0 if no SP exists. The size of block is 9*9 pixels; ordinarily two cores can't exist in one block with such a size. If block is too large, such as a 30*30 block, it may contain a core-delta pair, then the PI around such a block will be zero, and these two cores can't be detected.

SPs Orientation Estimation. We also get the estimation of the orientation of the SP if it is a core. We only consider the orientations of cores because cores appear much more frequently than deltas, and only one angle is necessary to describe the orientation of a core. In contrast three angles are needed to describe the orientation of a delta.

The orientations of the ridges in this block with a core are with great variation, so the block orientation, which is the Least Square Estimation of all those orientations of ridges in a block [9], is still far from accurate to describe the orientation of this core. See Fig 3. The estimation method we used is described here:

Fig. 2. Fingerprint Image and Block Orientation Field of foreground
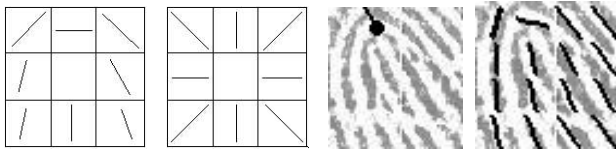
Fig. 3. (a) Typical Block Orientations in 3*3 Blocks with a core in the center block (b) Block Orientation Template (c) Image patch with a core (d) Block orientations near core
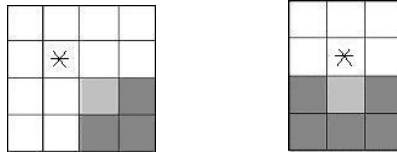
Fig. 4. Blocks involved in computing the orientation of core. The block containing a core is marked with' *', MCB is filled with gray, and those adjacent blocks considered are filled with black. (a) MCB is in the 8-neighbohood but not in the 4-neighborhood of the block containing a core (b) the case when the MCB is in 4-neighborhood

1) Find the Most Coherent Block (MCB) from the 8 neighbors of the considered block containing a core. The neighbor block with the least difference with the orientation in the corresponding position of a Block Orientation Template (BOT) is named MCB. The orientation of MCB is a coarse estimation of the orientation of the core. See Fig 3, Fig 4.
2) Calculating the orientation of the core using weighted average of orientation info in both the MCB and some of its adjacent blocks. Different adjacent blocks will be involved in the computation depending on whether the MCB is in the 4-neighborhood of the block containing a core or not. See Fig 4a, 4b.
3) Normalizing the orientation of the core to the range (-PI/2,PI/2).

Both the orientations of the MCB and those of some of its neighbors are considered because the orientations of ridgelines in them are more consistent than those in the block containing a core. A more robust estimation can therefore be obtained.

2.2 Precise Location

In this stage, the pixel where both directional gradients equal zero will be extracted as SP. The area to search each SP has been limited to a block area containing a SP by stage 1,while in [8] the whole image should be considered. In addition, since the type of SP has been determined in stage 1, the addition steps for type decision in [8] is not needed. The precise location stage consists of the following steps:

1) Calculation of two gradient relevant matrices *PX* and *PY*, where

$$PX(i, j) = \left|\partial_x(i, j)\right| - \left|\partial_y(i, j)\right|$$
$$PY(i, j) = \partial_x(i, j)\partial_y(i, j)$$

(1)

The range of (i,j) is within a 3 *3 blocks area centered at the interested block. We do not limit (i,j) to a block because the PI computation based on block field is performed in a coarse level. The loop to calculate PI surrounds not only the central block, but also some part of the neighbor blocks around it.

2) Gaussian filtering for PX and PY to alleviate the influence of noise.

3) Zero-point detection: A zero point is found in PX and PY, if in the 3*3 window centered at this point the maximum value and the minimum value are of different sign. If a zero point exists in the same location in both PX and PY, then the SP is detected in this position.

4) SPs post-processing: The gradient info of the adjacent pixels around SPs to discard improper SPs that often exist around the true one.

3   Experimental Results

In a computer with Celeron 500MHZ CPU, experiments have been done on the second FVC2000database.Comparison is made among the performances of algorithm in [7], algorithm in [8] and the approach proposed by us. We implemented algorithm in [8], and for algorithm in [7] we use the result reported in [7]. Method 3 and 4 in [7] are based on different foreground segmentation algorithms. See table1.

It can be shown that our algorithm is with the lowest Average Number of False SPs, and the performance of our algorithm on the Ratio of Fingerprints with False SPs is close to the best among all four algorithms. The best performance on both Ratio of Fingerprints with False and Missed SPs are arrived by method 4 in [7]. Relatively more SPs are missing in the result of our algorithm than in that of [7], which should be caused by the high level location in block orientation field. We found most missing SPs are near the foreground boundary. For those boundary blocks we are not able to compute the PI because the close loop for computing it includes background. To achieve even better performance, one can consider using other high-resolution approaches to deal with those boundary blocks as a further step of our approach.

However, we believe our approach should be more efficient in time expense than method 4 in [7], in which both point orientation field and covariance field need to be computed. The average time expense for SP detection using our algorithm is 0.98 second (including 0.42s for fingerprint segmentation, 0.05s for image normalization, 0.27s for block orientation field computation, and 0.24s for the two stages of SP extraction), while it is 1.66 second if algorithm in [8] is adopted. So we can see both robust performance and good time efficiency is achieved by the proposed hierarchy approach.

Table 1. Performance comparison among algorithms in [7],[8] and the proposed algrithm

Algorithm	Average No. of False SPs	Ratio of Fingerprints with False SPs	Ratio of Fingerprints with Missed SPs	Average Proc Time per image
[7] Method 3	0.8	0.2	0.02	N/A
[7] Method 4	0.5	0.13	0.05	N/A
[8]	2.67	0.73	0.55	1.66
Our Alg.	0.42	0.15	0.15	0.98

Fig. 5. Comparison of SPs detection results based on algorithm in [8] (left image) and the approach proposed by us (right image)

4   Conclusions

Both high-level info and local info of the fingerprint image are employed in the proposed a hierarchy approach for robust and efficient SP detection in this paper. Poincare Indexs in a block orient field are computed to locate blocks containing SPs and determine the types of the existing SP. The orientation estimation for the orientation of core is performed in a novel and efficient way. Local gradient info is used within interested blocks to detect the SP in a pixel level. The comparison among results from the proposed approach and some existing approaches show that the proposed one works with good accuracy and efficiency.

Reference

[1]   Kalle Karu, Anril K. Jain, "Fingerprint Classification", Pattern Recognition, Vol.18, No.3, 1996, PP. 389-404.
[2]   Weiwei Zhang; Yangsheng Wang; "Core-based structure matching algorithm of fingerprint verification", 2002, Proceedings. 16th International Conference on Pattern Recognition, Volume: 1, 2002 Page(s): 70 -74.
[3]   Asker M. Bazen and Sabih H. Gerez, "An Intrinsic Coordinate System for Fingerprint Matching", 3rd International Conference on AUDIO- and VIDEO-Based Biometric Person Authentication, Halmstad, Sweden, June 6-8, 2001.
[4]   G. A. Drets and H. G. Liljienstrom, "Fingerprint sub-classification: A neural network approach", Intelligent Biometric Techniques in Fingerprint and Face Recognition, L.C. Jain, U. Halici, I. Hayashi, S. S. Lee, and S.Tsutsui, Eds., PP. 109-134, CRC Press, Boca Raton, 1999.
[5]   P. Perona, "Orientation diffusion", IEEE Transactions on Image Processing, Vol. 7, No. 3, PP. 457-467, Mar. 1998.
[6]   Marius Tico, Pauli Kuosmanen, "A multi-resolution method for singular points detection in fingerprint images", Proceedings of the 1999 IEEE International Symposium on Circuits and Systems, vol.4, Page(s): 183 -186, Jul 1999.
[7]   Asker M.Bazen, Sabih H.Gerez, "Systematic Methods for the Computation of the Directional Fields and Singular Points of Fingerprints", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.24, No.7, July 2002.

[8]   Ramo, P.; Tico, M.; Onnia, V.; Saarinen, J.; "Optimized singular point detection algorithm for fingerprint images", Proceedings. International Conference on Image Processing, 2001, vol.3, Page(s): 242 -245.

[9]   L. Hong, Y. Wan, A. K. Jain, "Fingerprint image enhancement: algorithm and performance evaluation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume: 20 Issue: 8, Aug. 1998, Page(s): 777 –789.

[10]   L. Hong and A. K. Jain, "Classification of Fingerprint Image", Proceeding of 11th Scandinavian Conference on Image Analysis, June 1999.

[11]   Michael Kass, Andrew Witkin, "Analyzing Oriented Patterns", Computer Vision, Graphics, and Image Processing, vol.37, no.4, pp.362-385, 1987.

Compensatory Algorithm for Fingerprint Recognition

Hong Hui [1] and Jian-hua Li[2]

[1]School of Information Security Engineering
[2] Department of Electrical Engineering
Shanghai Jiaotong University, Shanghai, China
{huih,lijh888}@sjtu.edu.cn

Abstract. Present methods for fingerprint (FP) recognition have limited tolerance to distortion and noise in FP images. They usually employ larger threshold value of distance between the vector of a given FP and the stored templates. This will reduce the FRR, but increase the FAR concurrently. A novel compensatory algorithm for FP recognition is purposed in this paper to enhance algorithm robustness to distortion and noise and improve performance systematically. Compensatory algorithm contains three processing modules: simulated FP image generation module; Gabor feature extraction and coding module; ANFIS based matching module. The experimental result proves that this algorithm can achieve a higher accuracy.

1   Introduction

Automatic FP identification is the most convenient and comparatively mature among biometrics, and has been successfully used in many fields of security, such as Internet transactions, ATM, smartcard, e-banking, e-commerce, LAN/NT system entrance and access control, etc.. With the availability of solid-state sensor for FP capture, it has shown increasing usage in civilian domain. To match the stricter request of accuracy, the compensatory algorithm for FP recognition is purposed. The proposed algorithm is robust in the distortion and noise, and offers improve performance. The flow chart of the system using compensatory algorithm is shown in Fig. 1.

FP recognition system involves in two processes: registration process and recognition process. The registration process is shown above the dotted line, the image sensor for the acquisition of FP is shown as 1 while 2 is the captured FP image which is used as input template for the simulation of FP process module in 3; 4 is the series of FP images generated by distortion and noise addition algorithm; These FP images are transformed to corresponding Gabor vectors X_{in} from 6 through module 5 for Gabor feature extraction and coding to be trained by the ANFIS network; 8 indicates the output which denotes the model of FP X_i which is stored in the database (DB). Steps 1 and 2 are similar on the lower side of the dotted line. When the corresponding Gabor vector is transformed through module 5, the remaining task is to match the model stored in the DB and generate the result 4.
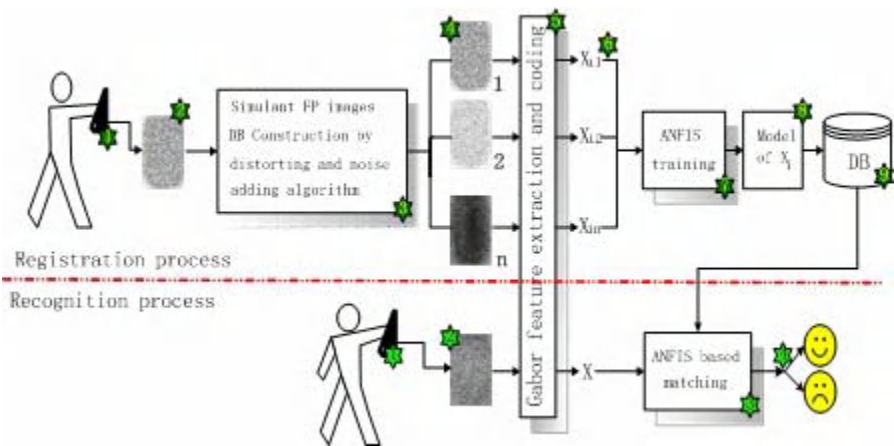
Fig. 1. Flow chart of the FP recognition system using compensatory algorithm: The registration process is shown on the upper side of the dotted line and the recognition process appears on the lower side (the numeral denote the steps of shown process)

2   Compensatory Algorithm

So-called "compensatory" denotes that the algorithm can compensate the influence of distortion and noise in FP images, thus enhancing the robustness of the system. Compensatory algorithm contains three processing modules: construction of simulant FP images; Gabor feature extraction and coding; ANFIS based matching.

2.1 Construction of Simulant FP Images

An FP database, generated by Cappelli's demo program (SfinGe2.51), has shown its usefulness for the algorithms evaluation similar to other databases [3]. So the training set of ANFIS can be constructed by way of this kind of simulated FP to increase algorithm's robustness to distortion and noise. The FP set is generated from the FP captured, which acts as the template.

The FP simulation algorithm [7] involves in several parameters, which is influenced the distortion and noise of FP images. Although the FP sets look like actually gathered when randomly chooses the parameters, the method is adopted to choose the parameters all-alone according to each level. For example, for the one distortion parameter such as the size of touch area of finger to sensor surface, 9 levels are given to generate the FP directly from the FP template captured on the basis of the mathematical models. The FP set generated by this way can also use for precisely evaluating the robustness of the algorithm to a certain parameter, which is convenient for improving the system performance pertaining to the poor element. This is the special characteristic of simulant FP database.

2.2 Gabor Feature Extraction and Coding

The structure-based algorithm uses Gabor filters with proper parameters, capturing both local and global details [1]. According to local parallel of ridges and furrows of FP image and making use of orientation selection property of Gabor filters, we defined rectangular tessellation with the equal-sized cells on the FP image ($w \times w$). Thus the information of specific orientation of each subimage is enhanced, while the information of non-specific orientation and non-specific frequency is suppressed.

The Gabor filters we designed can be defined by following equation:

$$h_r(x, y) = g(x', y') \cos(2\pi F x') \tag{1}$$

where $h_r(x, y)$ is the real part of Gabor function; (x', y') is the rotated form of the basis coordinate (x, y) in the spatial domain, i.e.,

$$(x', y') = (x \cdot \cos\theta_k + y \cdot \sin\theta_k, y \cdot \cos\theta_k - x \cdot \sin\theta_k) ;$$

and $g(x, y)$ the Gaussian function:

$$g(x, y) = \frac{1}{2\pi\sigma_g^2} \exp\left\{-\frac{x^2 + y^2}{2\sigma_g^2}\right\} \tag{2}$$

In order to achiever best performance, the parameter must be judiciously selected. We have done much work on this, and some result is reported [4], [5]. In this paper we select $\{\sigma_g, F, w\} = \{9.42, 0.1, 32\}$. The filtered image is the convolution of input image $i(x, y)$ with the filter $h_r(x, y)$:

$$f_h(x, y) = i(x, y) * h_r(x, y) \tag{3}$$

Fig 2 shows the Gabor feature extracting procedure of a sample FP image. Fig.3 is the block diagram of the whole steps of Gabor Vector generating module.

2.3 ANFIS based Training and Matching

In order to improve the accuracy, we have also proposed a matching algorithm using an Adaptive Neuro-Fuzzy Inference Systems (ANFIS) network, which is trained to identify the Gabor features of fingerprints. The subtractive clustering algorithm and the least-squares estimator are used to identify the fuzzy inference system. The training process is accomplished by using the hybrid-learning algorithm. ANFIS has much less tunable parameters than traditional neural networks so that it has the advantage of being significantly faster and more accurate than many pure neural network-based methods [6], [8].

There are two steps involved to construct the ANFIS. First, a fuzzy model (Fuzzy Inference System) is to be identified. Second, the model using ANFIS is trained.

Generally, it becomes very difficult to describe the rules manually in order to reach the precision needed with the minimized number of Membership Functions when the number of rules is large than three[6]. Therefore, an automatic model identification method becomes a must, which is often realized by means of a training set of r input-output pairs, $\{x_i, y_i\}, i = 1, \cdots, r$. The subtractive clustering algorithm is an

attractive approach [8] to the synthesis of ANFIS networks, which is based on clustering the training set. This method is used for structure the premise parameters identification.
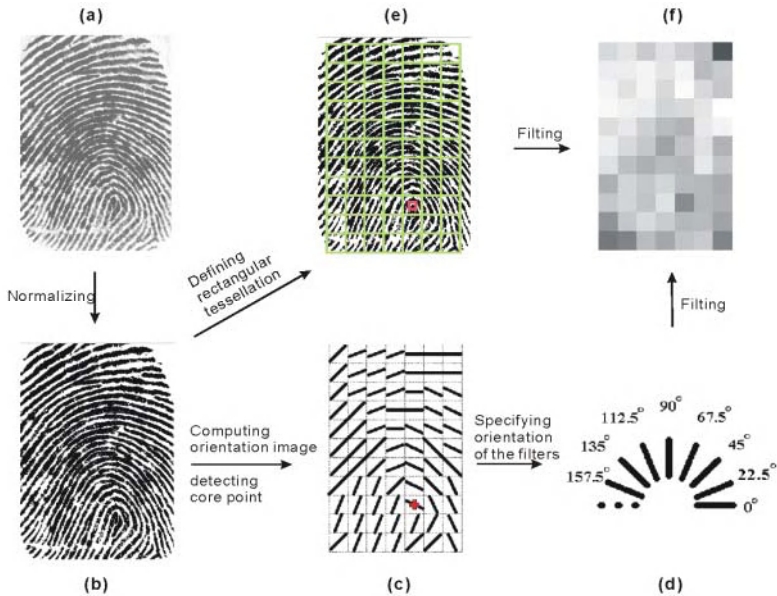
Fig. 2. The sketch diagram of Gabor feature extracting. Where (a) the input image; (b) the normalized image; (c)the orientation image with the core point; (d) denotation of 8 orientation filters; (e) image divided by rectangular tessellation; (f) the Gabor feature image

The training of ANFIS is carried out using a hybrid learning algorithm which uses a least square algorithm to identify the consequent parameters (assuming the premise parameters are fixed) and then the error signals are propagated backwards and the premise parameters are updated using a gradient descent algorithm (assuming the consequent parameters are fixed). The hybrid-learning algorithm is used to identify the consequent parameters. The consequent parameters identified this way are optimal (under the condition that the premise parameters are fixed) [6]. This means that the hybrid leaning algorithm will converge faster since it reduces the search space of the pure back propagation.

3   Experimental Results

For testing the proposed FP recognition system, we used the FP images given by FVC2002, and choose one of each 8 FPs of one finger as the template of simulant algorithm. There are 80 FP images generated for each template. So the total number of images used is 30 (fingers)×80=2400, in which 30×50=1500 FP images are stored for the training purpose, the others are for testing purpose. For totally 30 fingers, we built

models for each of them. The size of subimage is 32×32 pixels, and then the length of feature vector is 70 (the image has been cropped to fit the integral number of filter window). Each model is trained by the sets of the input/output pairs. The inputs are the feature vectors of fingerprints and the output equal to the value of 1 if the input vector is associated with the model, otherwise equal to 0.
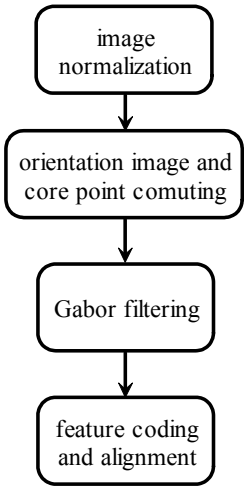
Fig. 3. Block diagram of Gabor Vector generating module

The training process is offline, the testing process is online that takes about 0.1 seconds of total 2400 FPs Gabor feature on a PentiumⅢ 866 PC.

The metric functions we used to evaluate the performance of our system are FRR (False Rejection Rate) and FAR (False Acceptance Rate). As for the testing process, the total number of training vectors as the input data for each model is separated into two parts: one is for genuine, the other is for imposer. Genuine is the FP corresponding to the model, its output for ANFIS is equal to the value of 1, whereas imposer does not present the model, its output is 0. So do the testing vectors. FRR and FAR are defined by:

$$FRR = \frac{\text{Number of Genuine ANFIS\_Output} \leq t}{\text{Number of Genuine Recognition Attempts(NGRA)}} \tag{4}$$

$$FAR = \frac{\text{Number of Imposter ANFIS\_Output} > t}{\text{Number of Imposter Recognition Attempts(NIRA)}} \tag{5}$$

where t is a threshold predefined, ranging from 0 to 1.

For 30 fingers, we built 30 models for each of them; and each model can be regarded as the template for a corresponding fingerprint. For each of 30 models, there are totally 8×30 FP images arranged to match it in the first 3 databases of FVC2002, which called "attempts", in which 8 genuine is corresponding to the model, the expected output is 1; and the other 8×29 fingerprints are imposer, expected output is 0. Table 1 shows the

experiment result for our system and the first three algorithms in FVC2002 for comparison. Hereinto, the Equal Error Rate (EER) is computed as the point where $FAR(t) = FRR(t)$.

Table 1. Experiment result and comparison with other algorithms

Algorithm		EER (%)
Compensatory algorithm		0.378
FVC2002	PA15	0.19
	PA27	0.33
	PB15	0.41

4   Conclusion and Future Work

In this paper, a novel compensatory algorithm for FP recognition is purposed to enhance algorithm robustness to distortion and noise and improve performance systematically. The simulant FP database used in compensatory algorithm can also use for precisely evaluating the robustness of the algorithm to a certain parameter, which is convenient for improving the system performance pertaining to the certain element. The experimental demonstration is reported, which proves that this algorithm could achieve a high accuracy.

Although the proposed method looks very computationally expensive, it is still an available method because the time cost procedure is offline and the online time is a little. Moreover, along with the development of hardware technology, the time cost of the algorithm will be drop largely by implemented on DSP or other embedded system.

The experimental demonstration proved that this algorithm could achieve relative high accuracy. To improve the system performance, the unsupervised clustering is still a challenge in our future researches.

References

[1]   A.K. Jain, S. Prabhakar, L. Hong and S. Pankanti: Filterbank-based FP Matching, IEEE Transactions on Image Processing, Vol. 9, No.5, (2000): 846-859
[2]   A.K. Jain, L. Hong, S. Pankanti and R. Bolle: An Identity Authentication System Using Fingerprints, Proc. IEEE, Vol. 85, No. 9, (1997) 1365-1388
[3]   D.Maio, D.Maltoni, R.Cappelli, J.L.Wayman, A.K.Jain: FP Verification Competition, IEEE Trans. PAMI, Vol.24, No.3, March (2002): 402-412
[4]   H. Hui, H. Zhou, L.Y. Wang: Optimal designed Gabor filters for FP recognition, Proceedings of SPIE Annual Meeting 2002 Seattle, Washington, USA, July 7–11(2002)
[5]   H. Hui, H. Zhou, L.Y. Wang: Structure-based FP matching using optimal Gabor filters, Proceedings of SPIE, Vol.4925, Photonics Asia 2002 Shanghai, China, October 14-18 (2002) 22-28
[6]   J.-S. R. Jang, C.-T. Sun, and E. Mizutani: Neuro-Fuzzy and Soft Computing, New Jersey: Prentice Hall, USA, (1997) 239-240

[7]   R. Cappelli: SFinGe: Synthetic FP Generator, proceedings 12th CardTech/SecurTech (CTST2002), April 2002. Invited: http://bias.csr.unibo.it/research/biolab/bio_tree.html

[8]   S.L.Chiu: Fuzzy model identification based on cluster estimation, Journal of Intelligent and Fuzzy Systems, Vol.2, No.3, (1994) 267-278

[9]   H.J. Yang, D.O. Liang, Y. Tian: Singular point detection of FP based on derectional field characteristics, ACTA Automatic Sinica, Vol. 27, No.2 (2001) 272-275

A Hierarchical Hough Transform
for Fingerprint Matching

Chaoqiang Liu[1], Tao Xia[2], and Hui Li[1]

[1] Temasek Laboratories and Centre for Wavelets
Approximation and Information Processing
National University of Singapore,
{tslliucq,tsllh}@nus.edu.sg,
[2] Centre for Wavelets, Approximation and Information Processing
National University of Singapore
iesxiat@nus.edu.sg

Abstract. This paper addresses the improvement on the matching accuracy and speed of generalized Hough transform for the biometric identification applications. The difficulties encountered in generalized Hough transform are investigated and a new hierarchical Hough transform algorithm is proposed, which is faster and more accurate compared to conventional generalized Hough transform.

1 Introduction

The biometrics is a technology that (uniquely) identifies a person based on his physiological or behavioral characteristics. Nowadays, biometrics identification is commonly in use for example at banks, airports, and government agencies. For identification signatures at present around the world, the popular methods are face images, fingerprint, voice print and iris pattern. Among them, the fingerprint is the most favorite one for its uniqueness for the individual. Accurate automatic personal fingerprint identification is becoming more and more important to the operation of our increasingly electronically interconnected information society.

An automatic fingerprint authentication system has four main design components: acquisition, representation (template), feature extraction, and matching. Among them the feature extraction including the fingerprint image processing problem is well studied [1]–[3]. However, no perfect feature extraction method to guarantee all fingerprint features are extracted exactly, especially for the latent fingerprints with poor quality and incompleteness. Therefore, the better matching algorithm for the fingerprint authentication is a big problem in this society.

Fingerprint matching has been approached from several different strategies, such as image-based [4], ridge pattern matching, graph-based fingerprint matching [5] and structural matching [6]. Also, given the minutiae representation of fingerprints, matching a fingerprint against a database converts to the problem of point matching [7]. In recent years, the Hough Transform and generalized Hough

Transform [8], GHT, has become an extremely popular method of 2D shape detection. The most exciting properties of Hough transform are its immunity to noise. The GHT-based approach converts point pattern matching [9] to a problem of detecting peaks in the Hough space of transformation parameters. The best transformation parameters of matching two point sets corresponding the peak is attained by accumulating the evidence in the discretized Hough space. Several novel approaches have been proposed to improve its computational efficiency and practicability [10],[11].

However, the traditional GHT used in fingerprint matching may lead to some errors due to the coarse quantization in the rotation parameter space. Based on the analysis of this kind of errors, the GHT in the translation parameter space is modified to ensure our candidate optimal solutions in the coarser level includes the real optimal solution in finer level. Therefore we develop a novel hierarchical Hough transform (HHT).

This paper is organized as follows. The fingerprint matching problem is briefly introduced in Section 2. Our method is described in Section 3. The experimental results and discussions are presented in Section 4. Conclusions are drawn in Section 5.

2   Fingerprint Alignment and Matching

As mentioned above, an automatic fingerprint authentication system is composed of four main design components. Although there are other representations such as pore [12], common representations of fingerprints are still minutiae (fingerprint features of ridge ending and ridge bifurcation [13]) based, and each minutia is described by its location (x,y coordinates) and the orientation. A feature extractor detects the minutiae for automated feature extraction and matching. In order to match one against another, the two points sets must be registered with respect to each other. The rotation, scale, and translation parameters of alignment transform are estimated using GHT. The actual transformation parameters are obtained by detecting peak in Hough space. The transformations $F_{s,\delta_x,\delta_y,\delta_\theta} : R^2 \to R^2$ given below is the most common one for alignment.

$$F_{s,\delta_x,\delta_y,\delta_\theta} \begin{pmatrix} x \\ y \end{pmatrix} = sR_{\delta_\theta} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \delta_x \\ \delta_y \end{pmatrix} \tag{1}$$

where $R_{\delta_\theta} = \begin{pmatrix} \cos\delta_\theta & \sin\delta_\theta \\ -\sin\delta_\theta & \cos\delta_\theta \end{pmatrix}$ is the rotation transform. s, (δ_x, δ_y), and δ_θ are the scale, shift, and rotation parameters, respectively.

In the fingerprint identification/authentication application, the scale difference is mainly caused by the dpi for different kind of sensors and it is supposed to be known in advance. Therefore in this paper we deal with other two parameters only, namely rotation and translation.

In practice, the parameter space is partitioned into the uniform cubic according to the specified precision of each parameters. Therefore, the accuracy

of alignment transformation depends on the quantization of the Hough space. Apparently, finer precision will cause more cubics and it will lead to the huge burden of computation cost. On the other hand, coarser precision sometimes will lead to the wrong result, especially for the fingerprint authentication applications. Therefore, the proper trade-off between computation cost and accuracy is indeed an important problem for the real application.

Mainly, there are two kinds of errors/difficulties of the existing fingerprint alignment and matching,

- Missing the matched pair due to the different tolerance bounds for different alignment precision and minutiae location.
- Duplication count of matched minutiae, i.e. there are more than one minutiae are matched against one minutia in the template and vice versa.

In this paper, we proposed a hierarchical Hough transform (HHT) to overcome these two problems. For simplicity, in the following contents, the denotation $v_1 \leq v_2$ for two vectors means for every element, this inequality holds and sometimes we use v to represent its x, y components. Also, when we state $\theta_1 \leq \theta_2$ for two angles, we always compare θ_1, θ_2 under the modular 2π.

3   Hierarchical Hough Transform

The accuracy of alignment transform $F_{\delta_x,\delta_y,\delta_\theta}$ plays an important role in fingerprint matching. To achieve higher accuracy, the precision of all parameters should be high even though that will lead to burden of computation cost. In this application, the parameter space consists of the transform of rotation and translation by which the two coordinate systems will be aligned.

Let U be the feature set of template fingerprint saved in database, and V be the feature set of latent fingerprint. The element in a feature set is a triplet consisting of coordinate and orientation of one minutia. Suppose the template feature set has M minutiae, i.e. $U = \{u_i, i = 1, \cdots, M\}$ and the latent feature set has N minutiae, i.e. $V = \{v_i, i = 1, \cdots, N\}$. Let $M_{xy} = (\max(|v_x|), \max(|v_y|))^T$ be the maximum value of the fingerprint image of set V. $\triangle_{xy} = (\triangle_x, \triangle_y)^T, \triangle_\theta$ are the expected precisions of alignment parameters of the translation and rotation respectively. $\triangle_{xy}, \triangle_\alpha$ is the tolerate bound of location and orientation difference of two matched minutiae respectively. $[x_{\min}, x_{\max}], [y_{\min}, y_{\max}]$ and $[\theta_{\min}, \theta_{\max}]$ are the intervals of possible translation and rotation angles for the latent fingerprint against the template fingerprint in database.

One typical error in popular GHT is that it will miss some matched pairs of minutiae. This is caused by the coarse rotation precision of alignment transform. The following theorem will give the necessary condition for such case.

Theorem 1 The Missing Pair of the Matched Minutiae
Suppose $u \in U, v \in V$ and the tolerance bounds for matching two fingerprints are $\triangle_x, \triangle_y$, the rotation angle for alignment transform of V against U is θ, if the transform imposed on V is $R_{\theta'}$, i.e. $v \xrightarrow{R_{\theta'}} v^1$ then, $\|v^1 - u\| \leq \|(\triangle_x, \triangle_y)^t\| + 2\sin(\frac{|\theta' - \theta|}{2})\|v\|$.

Theorem 1 shows that the actual error bounds should be different for different rotation precisions and different minutia in V. Same error bound for all cases will lead to the missing pairs. For the sake of simplicity, only alignment precision is considered here and $M_{xy,\rho} := 2\sin(\rho/2)M_{xy}$ is used to replace $2\sin(\frac{|\theta'-\theta|}{2})\|v\|$, in Theorem 1, where ρ is the rotation precision, i.e. $|\theta' - \theta| \leq \rho$. That will lead to algorithm 1.

Given an array of precision sequence $(\rho_1, \rho_2, \cdots, \rho_K)$, where $\rho_1 > \rho_2 > \cdots > \rho_K = \triangle_\theta$ and the threshold T of the minimum number of matched minutiae of two matched fingerprints.

Algorithm 1 *Hough Transform for orientation angle*

1. *for $j = 1$ to K,*
2. *$\Theta = \{[\theta_{\min}, \theta_{\max}]\}, \rho = \rho_j, \Theta_r = \emptyset, n{=}0$.*
3. *For each interval $[\beta, \beta'] \in \Theta$*
   (a) Partition $[\beta, \beta'] = \bigcup_{i=0}^{L-1} [\beta_i, \beta_{i+1}]$ with step size $\rho = \beta_i - \beta_{i-1}, \forall i,$.
   (b) for $i = 0$ to $L - 1$,
      i. $Q = \{(k,l)| |(u_{k,\theta} - v_{l,\theta}) - (\beta_i + \frac{\rho}{2})| \leq \rho + \triangle_\alpha, u_k \in U, v_l \in V\}$
      ii. If $j < K$
            using algorithm 2 to compute $m = H_1(Q, \triangle_{xy} + M_{xy,\frac{\rho}{2}}, \beta_i + \frac{\rho}{2})$
            If $m \geq T$ then $\Theta_r = \Theta_r \bigcup \{[\beta_i, \beta_{i+1}]\}$
         else
            Using algorithm 3 to compute $m = H_2(Q, \triangle_{xy} + M_{xy,\frac{\rho}{2}}, \beta_i + \frac{\rho}{2}, n)$
      iii. If $m > n$, then $m \rightarrow n$,
4. *if $\Theta_r = \emptyset$ or $n < T$ These two fingerprint can not be matched, end.*
5. *$\Theta_r \rightarrow \Theta$.*
6. *These two fingerprint are matched with n pairs of minutiae, corresponding score is computed accordingly.*

Remarks

1. For different angle precisions, we keep all possible candidates to avoid local maximum solution.
2. If $K = 1$, then this algorithm will be the generalized Hough transform.
3. The precision sequence can be chosen arbitrarily to meet the requirement of trade-off between the computation cost and the accuracy.

The algorithm proposed here is a hierarchical one in the sense of generating different parameter spaces for the translation in algorithm 1. The approximation of the exact number of matched pairs of two fingerprints U and V will be evaluated by coarse matching algorithm 2 and fine matching algorithm 3 respectively depending on where the rotation angle precision is coarse or not.

Algorithm 2 *Coarse Matching Algorithms of $H_1(Q, D, \beta)$*

1. *Quantizing $[x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}]$ into an accumulator $P(i,j) = 0$, $i = \lceil \frac{x_{\min}}{q_x} \rceil, \cdots, \lceil \frac{x_{\max}}{q_x} \rceil, j = \lceil \frac{y_{\min}}{q_y} \rceil, \cdots, \lceil \frac{y_{\max}}{q_y} \rceil$, q_x, q_y are quantization sizes.*
2. *$Q' = \{(x,y)|(x,y) = u_i - R_\beta v_j, (i,j) \in Q\}$*

3. *For each point $(x, y) \in Q'$, Let $\delta_x = (D_x - q_x)/2$, $\delta_y = (D_y - q_y)/2$,*
   $i_0 = \lceil \frac{x - \delta_x}{q_x} \rceil, \quad i_1 = \lceil \frac{x + \delta_x}{q_x} \rceil; \quad j_0 = \lceil \frac{y - \delta_y}{q_y} \rceil, \quad j_1 = \lceil \frac{y + \delta_y}{q_y} \rceil$
   $P(i, j) + 1 \rightarrow P(i, j), \forall i = i_0, \cdots, i_1, j = j_0, \cdots, j_1$
4. *$(i_{\max}, j_{\max}) = \underset{i,j}{\arg\max}(P(i, j))$.*
5. *return $P(i_{\max}, j_{\max})$.*

In algorithm 2, $(q_x, q_y)^T$ may differ to $\triangle_{xy}$. If $(q_x, q_y)^T = \triangle_{xy}$, it will be the most common matching algorithm; if $(q_x, q_y)^T = \triangle_{xy}/2$ as we used here, it is the more accurate matching algorithm.

Algorithm 2 is used for the rough rotation precision to speed up the matching process. For the last fine rotation precision case in algorithm 1 algorithm 3 is used. The exact number of matched pair between two fingerprint is impossible to obtained but the following theorem will tell us when the exact number of matched pair can be obtained.

Theorem 2 The Duplicate Count of the Matched Pair
Suppose two minutiae v_j and v_k match to u_i simultaneously with the error bound $\triangle_{xy}$ then $\|v_j - v_k\| \leq 2\|\triangle_{xy}\|$

The Theorem 2 shows that if $\|u_j - u_k\| \leq 2\|\triangle_{xy}\|$, and $\|v_j - v_k\| \leq 2\|\triangle_{xy}\|, \forall j, k$, then there is no duplicate count of the matched pair, in other words, no minutiae is matched to more than one minutiae in the other set.

Algorithm 3 *Fine Matching Algorithms of $H_2(Q, D, \beta, n)$*

1. *Quantizing $[x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}]$ into an accumulator $P(i, j) = 0$,*
   $i = \lceil \frac{x_{\min}}{q_x} \rceil, \cdots, \lceil \frac{x_{\max}}{q_x} \rceil, j = \lceil \frac{y_{\min}}{q_y} \rceil, \cdots, \lceil \frac{y_{\max}}{q_y} \rceil$.
2. *$V' = \{v'|v' = R_\beta v, v \in V\}$, $Q'' = \{(x, y)|(x, y) = u_i - v'_j, (i, j) \in Q\}$*
3. *For each point $(x, y) \in Q'$,*
   $i_0 = \lceil \frac{x - (D_x - q_x)/2}{q_x} \rceil, i_1 = \lceil \frac{x + (D_x - q_x)/2}{q_x} \rceil; j_0 = \lceil \frac{y - (D_y - q_y)/2}{q_y} \rceil, j_1 = \lceil \frac{y + (D_y - q_y)/2}{q_y} \rceil$
   $P(i, j) + 1 \rightarrow P(i, j), \forall i = i_0, \cdots, i_1, j = j_0, \cdots, j_1$
4. *$(i_{\max}, j_{\max}) = \underset{i,j}{\arg\max}(P(i, j))$ and $\delta_x = (i_{\max} + \frac{1}{2})q_x, \delta_y = (j_{\max} + \frac{1}{2})q_y$,*
5. *if $P(i_{\max}, j_{\max}) \leq n$, return 0;*
6. *Generating the matching matrix $M(U, V') = \{m_{ij}\}_{ij}, i = 1, \cdots, M, j = 1, \cdots, N$,*
   where $m_{ij} = 0$, if $|u_i - v'_j - (\delta_x, \delta_y)^T| \leq \triangle_{xy}$, otherwise $m_{ij} = 1$.
7. *return $m = \min(\sum_i (1 - \prod_j m_{ij}), \sum_j (1 - \prod_i m_{ij}))$.*

Remarks

1. The actual number of matched pairs m_0 satisfies $rank(M) \leq m_0 \leq m$.
2. When the application prefers the strict criteria for the matched fingerprint, $m = rank(M)$ could be used in last step.
3. The theorem 2 shows that if $\|u_j - u_k\| \leq 2\|\triangle_{xy}\|$, and $\|v_j - v_k\| \leq 2\|\triangle_{xy}\|, \forall j, k$, then $rank(M) = m_0 = m$. That also implies $m_0 = P(i_{\max}, j_{\max})$, therefore if the two fingerprints U, V satisfies the condition in theorem 2, we will use algorithm 2 for all rotation angle precisions.

The three algorithms, Hough Transform for orientation angle, coarse matching algorithm and fine matching algorithm in HHT provide a solution to overcome the problems caused by two kind of difficulties of GHT.
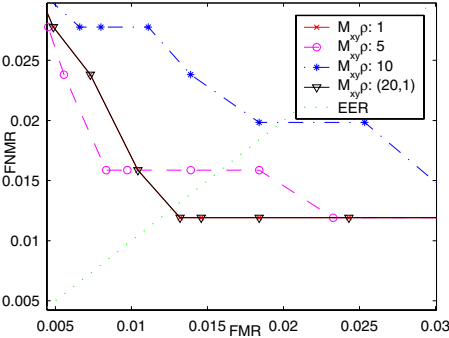
Fig. 1. The matching accuracy for different $M_{xy}\rho$ when $(q_x, q_y)^T = \triangle_{xy}/2$

4   Experimental Results

The performance of the HHT can be convinced by the comparison with the GHT. The test set DB1 of Fingerprint Verification Competition [14] is used for the simulation. The image resolution is 300×300, which means the $M_{xy} = 150$ and maximum rotation is in the range $[-15°, 15°]$, $\triangle_x = \triangle_y = 7$, $\triangle_\alpha = 15°$.

We conduct the simulation result for different rotation precisions $(\rho_1, \cdots, \rho_K)$ for algorithm 1 and different $(q_x, q_y)^T$ in algorithm 2. Obviously, for the larger ρ, the speed is faster, and if $K = 1$, HHT will be the conventional GHT. Instead of using ρ, we list $M_{xy}\rho$ as $M_{xy,\frac{\rho}{2}} \approx \frac{1}{2}M_{xy}\rho$, that means when $M_{xy}\rho = 1$, corresponding possible distortion of the error bound using in algorithm 1 will be less than 0.5 and can be negligible. For $K = 1$, $M_{xy}\rho$ ranges from 1 to 10 and for $K = 2$, $M_{xy}(\rho_1, \rho_2) = (20, 1)$ is the result of HHT.

It shows that the accuracy of HHT equals to the accuracy of the finest precision case for conventional GHT and the speed of matching is around $6 - 7$ times faster. The EER (equal error rate, defined in [14]) row shows that if we do not consider the problems encountered by GHT, the matching accuracy will degrade a lot. Also the use of $\triangle_{xy}/2$ in algorithm 2 improves the accuracy range from 10% to 50%. The result is listed in table 1 and figure 1 shows FNMR (False Non-Match Rate) and FMR (False Match Rate), also defined in [14], the intersection points of $y = x$ to the curves for different $M_{xy}\rho$ are the EER points.

Table 1. Comparison between HHT and GHT

$M_{xy}\rho$	(20, 1)		1		5		10	
$(q_x, q_y)^T$	$\frac{\triangle_{xy}}{2}$	$\triangle_x$	$\frac{\triangle_x}{2}$	$\triangle_x$	$\frac{\triangle_x}{2}$	$\triangle_x$	$\frac{\triangle_x}{2}$	$\triangle_x$
EER	**1.29%**	1.98%	1.29%	1.98%	1.59%	2.20%	1.98%	2.13%
time(ms)	**4.86**	4.00	30.37	28.26	6.39	5.90	3.37	3.10

5   Conclusions

In this paper we analysis two kinds of errors in fingerprint matching, namely missing matched pair and duplicate matching problem and proposed a new algorithm to overcome these difficulties encountered by conventional GHT algorithm. The proposed algorithm can achieve the accurate matching result in an efficient way. Experiment results show that the error will degrade the matching performance and our hierarchical Hough transform will overcome this problem in the efficient way.

References

[1] N. Ratha, S. Chen, and A. K. Jain, "Adaptive flow orientation based feature extraction in fingerprint images", *Pattern Recognition*, vol. 28, no. 11, 1657–1672, 1995. 373

[2] B. M. Mehtre, "Fingerprint image analysis for automatic identification", *Machine Vision and Applications*, vol. 6, 124–139, 1993.

[3] Dario Maio and Davide Maltoni, "Direct gray-scale minutiae detection in fingerprints", *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, no. 1, pp.27–40, 1997. 373

[4] R. Bahuguna, "Fingerprint verification using hologram matched filterings", *Proc. Biometric Consortium Eighth Meeting,* San Jose, CA, June 1996. 373

[5] M. Eshera and K. S. Fu, "A graph distance measure for image analysis", *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, no. 3, 1984. 373

[6] K. Hrechak, and J. A. McHugh, "Automated fingerprint recognition using structural matching", *Pattern Recognition*, vol. 23, no. 8, pp. 893–904, 1990. 373

[7] J. Ton and A. K. Jain, "Registering landsat images by point matching", *IEEE Trans. Geosci. Remote Sensing*, vol. 27, no. 5, pp. 642–651, 1989. 373

[8] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary patterns", *Pattern Recognition*, vol. 13, no. 2, pp. 111–122, 1981. 374

[9] G. Stockman, S. Kopstein and S. Benett, "Matching images to models for registration and object detection via clustering", *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 4, no. 3, pp. 229-241, 1982. 374

[10] N. K. Ratha, K. Karu, S. Chen, and A. K. Jain, "A real-time matching system for large fingerprint database", *IEEE Pattern Anal. Machine Intell.*, vol. 18, no. 8, pp. 799–813, 1996. 374

[11] S. H. Chang, F. H. Cheng, W. H. Hsu, and G. Z. Wu, "Fast algorithm for point pattern matching:invariant to translations, rotations and scale changes", *Pattern Recognition*, vol. 30, no. 2, pp. 311–320, 1997. 374

[12] Andrea R. Roddy and Jonathan D. Stosz, "Fingerprint Features–Statistical Analysis and System Performance Estimates", *Proc. IEEE*, vol. 85, no. 9, pp. 1390–1421, 1997. 374

[13] U. S. A. Government Publication, "The Science of Fingerprints: Classification and Uses", *Washington, D. C.:U. S.Dept. of Justice, FBI*, Government Printing Office, 1985. 374

[14] Fingerprint Verification Competition 2000, http://bias.csr.unibo.it/fvc2000. 378

Fingerprint Alignment Using Line Segments

Cristiano Carvalho and Hani Yehia

Center for Research and Development on Electrical Engineering (CPDEE)
Federal University of Minas Gerais (UFMG)
Belo Horizonte–MG Brazil
crc@cpdee.ufmg.br
hani@cefala.org

Abstract. In this paper, it is provided statistical evidence that supports fingerprint minutiae matching algorithms which use line segments formed by pairs of minutiae as candidates for pivots. This pivots are used to superimpose the analyzed fingerprint templates. Also in this work an algorithm to improve the matcher performance for uncontrolled acquisition systems is proposed. This algorithm employs a technique to sort the minutiae list in templates increasing the chances that corresponding line segments in two templates are tested in the early algorithm iterations. The analysis and the proposed algorithm are validated with data from FVC2000 and FVC2002 databases.

1 Introduction

"Because of its well-known distinctiveness and persistence properties over time, fingerprints are the most widely used biometric characteristics" [1]. "These are imprints of impressions of patterns formed by friction ridges of the skin in the fingers" [2]. "The two most prominent fingerprint features, called minutiae, are ridge termination and ridge bifurcation" [1].

In several works on minutiae based fingerprint recognition, as in [3], [4] and [5], fingerprint matching is the last link in the chain of tasks that make it possible to tell the difference or similarity between two fingerprints. In this process, the objective is to measure the correspondence among minutiae in one input fingerprint and those in previously enrolled templates stored in a database.

The problem of aligning fingerprints is not a trivial task since we do not have a robust reference point to act as a guide to the superimposition of the templates. No minutia in the template can be safely used as an absolute reference because there is a probability that it will not be detected in future samples.

Udupa, Garg and Sharma [6], inspired in a Computer Vision alignment approach for model based recognition of objects, proposed an algorithm in which given two templates **D** and **I** that should be verified for matching, segments formed by pairs of minutiae in **D** and **I** are retrieved and a hypothesis of correspondence is made for the segments. Given this correspondence, the displacement and rotation parameters for a rigid transformation can be obtained and the templates can be superimposed to test the hypothesis validity. This test constitutes
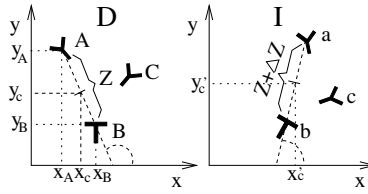
Fig. 1. Principle of line segments alignment

the kernel of the algorithm and must be avoided because of its computational load. So, the authors in [6] propose to filter the candidates using the differences in segment length, in the orientation of the ridge associated with the minutiae and in ridge count.

Figure 1 shows a typical situation for fingerprint recognition in which the minutiae in **I** are displaced and rotated with respect to their corresponding minutiae in **D**. In this figure, minutiae A and B correspond, respectively, to minutiae a and b. The test $(A - B) \rightleftarrows (a - c)$, for instance, can be discarded using the length discrepancy with no need to superimpose the templates. The parameter Δz in Fig. 1 is a tolerance to plastic distortions.

In this work we provide statistical evidence that supports the algorithm based on line segments for fingerprint alignment and introduce a modification to reduce the expected number of superimposition operations.

In our tests we use the software FingerLab [7] to process the fingerprints and the databases from FVC2000 [8] and FVC2002 [9]. The software and the databases are publicly available. As the used version of the software does not retrieve the ridge count between minutiae, our analysis is based only in the minutiae position (x, y) and the associated ridge orientation (θ).

2   Statistical Analysis of Fingerprint Data

Let

$$z_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} = \sqrt{\Delta_x^2 + \Delta_y^2} \tag{1}$$

be the length of the segment formed by the pairs of minutiae i and j where $\Delta_x = (x_i - x_j)$ and $\Delta_y = (y_i - y_j)$. The probability density function of the random variables Δ_u (u = x or u = y) is given by

$$f_{\Delta_u}(\Delta_u) = \int_{-\infty}^{\infty} f_u(\Delta_u + u) f_u(u) du \ . \tag{2}$$

According to (2) $f_{\Delta_u}(\Delta_u)$ tends to be a normal distribution and the closer $f_u(u)$ is to a normal distribution the more this assumption is valid. Figure 2 shows $f_X(x)$ and $f_Y(y)$ obtained for the fingerprints from FVC2000 and FVC2002, where it can be observed that the real distributions for these variables tend to be normal distributions.
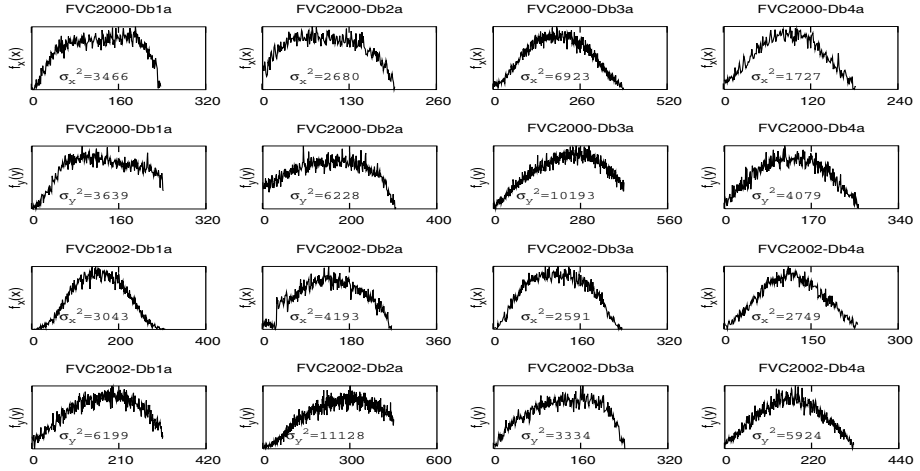
Fig. 2. Distributions $f_X(x)$ and $f_Y(y)$ for fingerprints from FVC2000 and FVC2002

This analysis and (2) lead to the conclusion that the random variables Δ_X and Δ_Y tend to be normal and independent with zero mean. So, we can get an approximation for the distribution of the random variable z with

$$f_Z(z) \approx \frac{z}{\hat{\sigma}^2} e^{-z^2/2\hat{\sigma}^2} U(z), \tag{3}$$

where

$$U(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{otherwise} \end{cases}. \tag{4}$$

Equation (3) shows that $f_Z(z)$ can be approximated by a Rayleigh distribution [10]. One way to estimate $\hat{\sigma}$ is to obtain $f_Z(z)$ from the data, compute its mean, $E\{z\}$, and use

$$\hat{\sigma} = \frac{E\{z\}}{\sqrt{\pi/2}}. \tag{5}$$

The knowledge of the distribution for the random variable z is important because with this information we can predict the expected performance for a given system. In the next section we present some tests to validate the hypothesis that (3) is true.

2.1 Validation of the Theoretical Model

Figures 3(a) and 3(b) show, respectively, the distributions found in the FVC2000 and FVC2002 databases with the plots of the estimated distributions modeled by (3) and using (5) to estimate the parameter $\hat{\sigma}$. Pearson's test statistic sum [10]
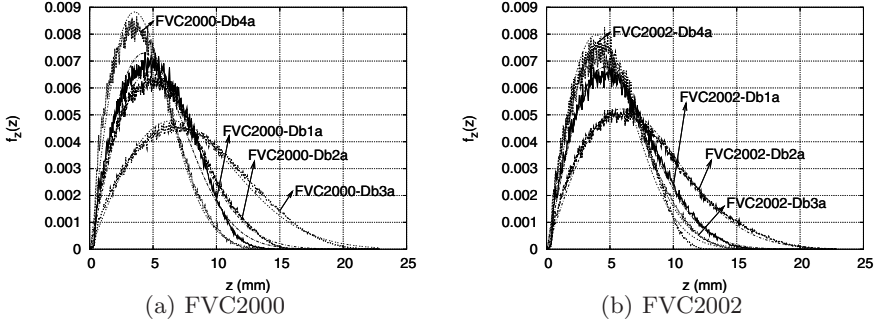
Fig. 3. The distribution functions for segment lengths in databases from FVC2000 and from FVC2002 superimposed by the theoretical distributions

confirmed the null hypothesis so we can claim that the distribution for the segment lengths formed by all pairwise combinations of minutiae in a fingerprint database is probably a Rayleigh distribution.

Udupa, Garg and Sharma in [6] claim that the number of segments to be tested can be reduced, without compromising too much the system performance, by selecting only those segments for which the lengths fall in a pre-defined range $[z_{\min}, z_{\max}]$. They suggest $z_{\min} = 2.54$mm and $z_{\max} = 7.62$mm (50 to 150 pixels for a resolution of 500 dpi) as typical values. We can compute the probability $\overline{p}$ that two templates do not match due to this constraint using

$$\overline{p}(z_{\min}, z_{\max}) = (1 - p_\alpha)^M + (1 - p_\alpha)^N, \tag{6}$$

where M and N are the number of minutiae in templates **I** and **D**, respectively, and p_α is given by

$$p_\alpha = e^{-z_{\min}^2/(2\hat{\sigma}^2)} - e^{-z_{\max}^2/(2\hat{\sigma}^2)}, \tag{7}$$

which was obtained integrating (3) from $z_{\min}$ to $z_{\max}$. As an example, for two hypothetical templates from database FVC2000-Db1_a ($\hat{\sigma} \approx 4.3$mm), say with $M = 30$ and $N = 35$, the probability $\overline{p}(2.54\text{mm}, 7.62\text{mm}) = 7.6 \times 10^{-14}$. Using $\Delta\theta = 22.5^o$ with $p(min(|\theta'-\theta|, 360^o-|\theta'-\theta|) \leq \Delta\theta) = 0.267$ as suggested in [11] and a fixed tolerance of 1mm for segment length differences, we can expect to make about 153 tests (0.06% of the total number of candidates).

With the experimental tests we have confirmed the validity of the proposed theoretical model. In the next section we analyze the expected number of tests and an algorithm is introduced to improve the system performance.

3   Expected Number of Tests

Udupa, Garg and Sharma in [6] suggest the use of the top ten alignments – those with the larger number of mates – to compute a final score for the matching. In

our analysis we do not consider this final consistency check. Instead, we assume that if we have enough mates in an alignment, the probability that the two templates come from different fingerprints is negligible. Thus, the algorithm stops when it first detects K minutiae mates ($K = 10$ in our tests).

The probability that two segments from different templates match in length and orientation is expected to be smaller for templates coming from different fingers than for those from the the same finger. So, the expected number of tests, or superimpositions, $E\{N_t\}$, is larger for a genuine test. This is visible in Fig. 4(a) where we present the empirical cumulative distribution function of N_t obtained for FVC2000-Db1_a database while running one algorithm implementation. The expected number of tests for genuine was 18.1 while for impostor it was 4.4. As an example, ten tests were sufficient to correctly reject 92% of the false template candidates and to correctly authenticate 60% of the genuine.

As the FVC2000 databases acquisitions where controlled to limit the difference in rotation of fingerprints to a small range ($[-15^o, +15^o]$), the probability p_{mn} that the first m minutiae in template $\mathbf{I}$ match the first n minutiae in $\mathbf{D}$ is higher for this database than it would be for a situation where minutiae position in lists are random, e.g. with more freedom to rotate the finger or with more false detected minutiae. To estimate how the system would behave for a situation like this we repeated the test after randomly permuting minutiae in the template list. Results are presented in Fig. 4(b) where a performance degeneration can be noticed. The expected number of tests for genuine rose from 18.1 to 30 while the expected number of tests for impostor rose from 4.4 to 7.2.

To deal with the loss of performance for uncontrolled acquisitions and based on the statistical model devised in (3) we propose the following algorithm:

let T denote a fingerprint template with M minutiae represented by coordinates x and y; let $Info$ be a structure with size $M(M-1)/2$ and O, P and $F_{\mathbf{z}}$ denote fields in this structure; let $\Delta z(z)$ be the tolerance for length z; let $\hat{\sigma}$ denote the parameter for the distribution of the random variable z for the system; let T' be the output template;
$c = 1$;
for $i = 1$ to M **do**
    for $j = i + 1$ to M **do**
        $z = \sqrt{(x(i) - x(j))^2 + (y(i) - y(j))^2}$;
        $Info[c].F_{\mathbf{z}}(c) = \left(e^{-[z-\Delta z(z)]^2/2\hat{\sigma}^2} - e^{-[z+\Delta z(z)]^2/2\hat{\sigma}^2}\right)$;
        $Info[c].O = i$; $Info[c].P = j$;
        $c = c + 1$;
    end for
end for
sort $Info$ using $F_{\mathbf{z}}$ as a key in descending order;
$m = 1$; $c = 1$;
while $c <= M(M-1)/2$ and $m <= M$ **do**
    if $T(Info[c].O)$ is not in T' **then**
        $T'(m) = T(Info[c].O)$;

$m = m + 1;$
end if
if $T(Info[c].P)$ is not in T' **then**
$\quad T'(m) = T(Info[c].P);$
$\quad m = m + 1;$
end if
$c = c + 1;$
end while
The output is the template T' with minutiae sorted.

This algorithm sorts the minutiae lists in both templates using the probability of the lengths formed by minutiae pairs as a key to the sorting operation. We do this to increase p_{mn} and then approximate the results to those obtained for controlled acquisition. Note that this algorithm needs to be run only once for each template after the minutiae detection has taken place. Results, which are close to those obtained for the controlled situation shown in Fig. 4(a), are presented in Fig. 4(b). The expected number of tests dropped from 30 to 20.5 for genuine and from 7.2 to 4.5 for impostor. The fact that these values are close to those observed in the controlled situation, is an evidence that our algorithm performs the task of sorting the minutiae lists in a coherent and robust manner.

4   Conclusion and Remarks

In this paper we analyzed the statistical properties of fingerprint data to show the efficacy of the line segments algorithm for fingerprint alignment. Based on assumptions about the fingerprint statistics we devised the hypothesis that the distribution for the lengths of segments in fingerprint templates can be approximated by a Rayleigh distribution. This hypothesis was confirmed with tests in the public available databases from FVC2000 and FVC2002. We verified that the parameter $\hat{\sigma}$ for the Rayleigh distribution depends on the fingerprint nature as well as on the sensor used to capture the fingerprint.

(a) N_t for controlled acquisition   (b) N_t for uncontrolled acquisition
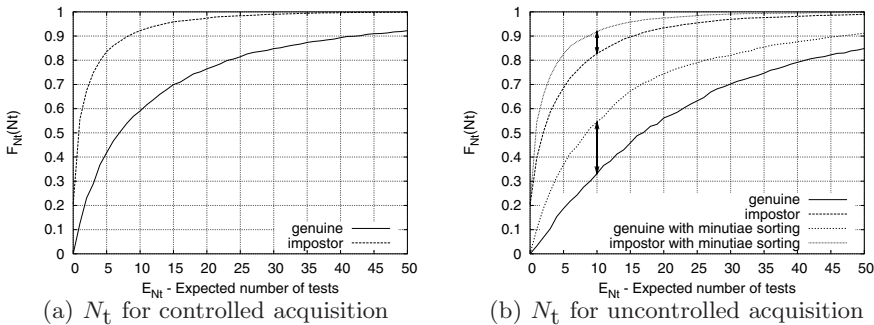
Fig. 4. Distribution for the number of tests spent by the algorithm, N_t, for genuine and impostor for controlled (a) and uncontrolled acquisition (b)

With empirical tests we confirmed that using the filter techniques proposed by Udupa, Garg and Sharma [6], the expected time spent to reject an impostor is shorter than the expected time to authenticate a genuine. We also show that the algorithm can suffer on cases where the minutiae relative position in the template list changes substantially, e.g. for uncontrolled acquisition or high false minutiae detection rate, and proposed an algorithm to sort the minutiae list based on the probability of the segments formed by minutiae pairs. For the cases where the minutia position changed randomly we concluded that our algorithm was able to reduce the average number of tests by 31% for genuine and by 37% for impostor matching. The algorithm complexity is $O(M \log M + M(M - 1))$, where M is the number of minutiae in the template, however it needs to be run only once after the minutiae are detected.

Acknowledgements

References

[1] Maltoni D., Maio D., Jain A., Prabhakar S.: Handbook of fingerprint recognition. Springer (2003) 2–2, 30–30 380
[2] Willis A., Myers L.: A Cost-effective Fingerprint Recognition System for Use with Low -quality prints and Damaged Fingertips. The Journal of the Pattern Recognition Society (2001) 255–270 380
[3] Jain A., Hong L., Bolle R.: On-line Fingerprint Verification. IEEE Transactions on Pattern Analysis and Machine Intelligence (1997) vol. 19, number 4 302–313 380
[4] Luo X., Tian J., Wu Y.: A Minutia Matching Algorithm in Fingerprint Verification. in Proc. Int. Conf. on Pattern Recognition(15th), (2000) vol. 4 833-836 380
[5] Wayman J.: Error rate equations for the general biometric system. IEEE Robotics and Automation Magazine (1999) vol. 6 35–48 380
[6] Udupa R., Garg G., Sharma P.: Fast and Accurate Fingerprint Verification. Audio- and Video-Based Biometric Person Authentication, Third International Conference, AVBPA 2001 Halmstad, Sweden, June 6-8, 2001, Proceedings. Lecture Notes in Computer Science 2091 Springer 2001, ISSN: 0302-9743 380, 381, 383, 386
[7] Carvalho C.:(2004) FingerLab–Fingerprint Laboratory. Version pre-alpha 0-0-1. Available from <http://sourceforge.net/projects/fingerlab> [Accessed 03/15/2004] 381
[8] Maio D., Maltoni D., Cappelli R., Wayman J., and Jain A.: FVC2000: Fingerprint Verification Competition. 15th IAPR International Conference on Pattern Recognition, Barcelona, Spain, Sep. 3-7, 2000 381
[9] Maio D., Maltoni D., Cappelli R., Wayman J., and Jain A.: FVC2002: Second Fingerprint Verification Competition. in Proc. Int. Conf. on Pattern Recognition (16th), vol. 3, pp. 811-814, 2002 381

[10] Papoulis A.: Probability Random Variables, And Stochastic Processes. Electrical Engineering. Communications and signal processing, McGraw-Hill International ed. 3 (1991) 66–79, 139–140, 173–178, 249, 273–275   382

[11] Pankanti S., Prabhakar S., Jain A.: On the Individuality of Fingerprints. IEEE Transactions on Pattern Analysis and Machine Intelligence vol. 24 (2002) 1010-1025   383

Private Fingerprint Verification
without Local Storage

Florian Kerschbaum[1], Mikhail J. Atallah[2], David M'Raïhi[1], and John R. Rice[2]

[1] Arxan Technologies
720 Market Street
San Francisco, CA 94102
`fkerschbaum,dmraihi@arxan.com`
[2] Dept. of Computer Science
Purdue University
West Lafayette, IN 47906
`mja,jrr@cs.purdue.edu`

Abstract. We describe a protocol to solve the problem of comparing fingerprints without actually exchanging them. We extend this to private verification where the verifier does not obtain fingerprint information. We show that no substantial information is leaked to the other party even during multiple protocol runs and present evaluation results from a prototype implementation.

1 Introduction

There are specific security concerns regarding fingerprint systems [11], but fingerprints have distinct advantages, because they cannot be lost or forgotten and they can be measured cheaply and non-intrusively. The disadvantages of fingerprints are that they cannot be changed or re-issued. Their disclosure is permanent. Large databases linking fingerprints to private information exist. It is therefore beneficial to keep fingerprints private.

We present a protocol where two parties each have a fingerprint image. They want to compare them for a match, i.e. that both images are from the same finger, but they do not want to reveal them, unless they are a match. We extend this a protocol where one party does not learn anything about the fingerprint, but can use it to verify the other's party identity.

Fingerprints are approximate, i.e. small changes happen every time one measures it. Cryptographic methods, such as encryption or hashing are not distance-preserving. Small input changes lead to outputs that cannot be compared for proximity any longer.

2 Related Work

We present work from three related areas of research. First, we review literature combining cryptography and biometrics, second, fingerprint matching algorithms and third, secure multi-party protocols.

2.1   Private Biometrics

[4] presents an algorithm combining error correcting codes and hash functions such that biometric templates can be compared using Hamming distance. [10] extends that algorithm, such that the biometric template is no longer treated as only the information part of the code, but as a corrupted code word itself. Our protocol provides fingerprint verification using hamming distance and homomorphic encryption.

[2] describes an algorithm to make biometric templates non-transferable. The biometric template is XOR-ed with a random number, or simply permutated before enrolling. Although this achieves non-transferability, it does not protect the template during comparison.

[1] introduces a very nice system combining fingerprints with cryptography. A one-way function is applied to each fingerprint image's Fourier transform that results in a bit pattern that is mapped into a cryptographic key. The comparison itself reveals the fingerprint, so it has to be done in a secure device.

[16] presents another technique to achieve the ability to re-issue biometrics. It disturbs the biometric image using a random image transformations generating a large number of templates, but it does not guarantee the user's biometric privacy, because it could be undone using external knowledge or important parts might not have been modified.

2.2   Fingerprint Matching

Many algorithms for electronic fingerprint matching have been developed, to list a few [9, 14, 17]. A common type of algorithm matches fingerprints based on minutiae. Several clever methods have been developed to extract minutiae from the fingerprint image, e.g. [15].

The set of minutiae is the fingerprint template. It contains all information necessary to participate in a fingerprint comparison protocol, but less than a fingerprint image.

Our implementation follows the extraction algorithms from [9]. We distinguish two types of minutiae – forks and endings – and extract the ridge orientation with basic texture extraction algorithm. We do not improve upon those algorithms, but use them in our protocol.

2.3   Secure Multi-Party Computation

General secure multi-party computation [7, 18] can construct protocols for any joint computation where the parties do not want to reveal their input. The resulting protocols are not optimized to efficiently solve specific problems.

Many specific protocols have been developed. We will mention only some closely related ones. [5] studies approximate matching for databases access. [6] examines the security of privately computing approximations.

In the next section we will give an overview of the building blocks of our protocol, then we will present the fingerprint comparison protocol, private fingerprint verification, its security and practical evaluation.

3   Assumptions and Model of Computation

In the Fingerprint Verification Protocol both parties, Alice and Bob, have a fingerprint image. We compose the Fingerprint Comparison Protocol from two building-block protocols. Then we show how to modify these protocols, such that Bob will not obtain fingerprint information, nevertheless can still verify Alice's identity using an enrolled template.

We use the honest-but-curious model for the communicating parties, i.e. we assume that the participating parties will follow the protocol, but compute information about the other party's secret data. Our security analysis shows that no substantial information is leaked to the other party in any part of the protocol even if it is run multiple times between the same parties.

Our performance results conclude that it is possible to match fingerprints using our protocol by evaluating the matching results from the prototype implementation on a generated fingerprint database.

4   Fingerprint Comparison Protocol

Each party starts with a fingerprint image. Minutiae extraction can be done without the other's party input. We denote the fingerprint template as $S_A = (a_1, ..., a_n)$ and $S_B = (b_1, ..., b_m)$ for Alice and Bob, respectively. Each minutia is a four tuple (x, y, t, ϕ), where x and y are x- and y- coordinates, t is the type and ϕ the angle of the ridge orientation at coordinate (x, y).

Fingerprint Alignment translates and rotates the two templates such that matching minutiae have approximately the same absolute location. Fingerprint Comparison counts the number of those minutiae with the approximate same absolute location as matches. The final score of our protocol is the number of matching minutiae.

4.1   Fingerprint Alignment

Our alignment protocol uses two minutiae P and Q to align the templates. These two minutiae will be rotated and translated into two common locations P' and Q'. The templates are aligned, if Alice and Bob have picked matching minutiae. Then most matching minutiae will have approximately the same absolute location.

Fingerprint Alignment Protocol

Input: Alice has a fingerprint template S_A and Bob has a fingerprint template S_B.
Output: Alice has an aligned fingerprint template S'_A, Bob has an aligned fingerprint template S'_B, such that if $a_i = (x_a, y_a, \phi_a, t_a)$ matches $b_j = (x_b, y_b, \phi_b, t_b)$ then $x_a \approx x_b, y_a \approx y_b, \phi_a \approx \phi_b$.

1. Alice chooses a random value r as a cryptographic hash of her fingerprint image I: $r = H(I)$,
2. Alice chooses a random permutation Γ_r based on r and permutes her n minutiae $S_A = (a_1, \ldots, a_n)$: $\boldsymbol{a'} = (a_{\Gamma_r(1)}, \ldots, a_{\Gamma_r(n)})$.
3. Alice forms $\lfloor \frac{n}{2} \rfloor$ element-distinct pairs $\boldsymbol{p} = (p_1, \ldots, p_{\lfloor \frac{n}{2} \rfloor})$ where $p_j = (a'_{2j-1}, a'_{2j})$ for $j = 1, \ldots, \lfloor \frac{n}{2} \rfloor$.
4. Alice randomly selects a pair $A_{r,r'} = (a_r, a_{r'})$ from $\boldsymbol{p}$.
5. Alice rotates and translates S_A into S'_A, such that:
   - the location of a_r is at the origin: $a_r = (0, 0, t_a, \phi_a)$;
   - the location of $a_{r'}$ is on the positive x-axis: $a_{r'} = (x'_a, 0, t'_a, \phi'_a), x'_a > 0$.
6. Alice sends the pair $A_{r,r'}$ to Bob.
7. For each $i \in \{1, \ldots m\}$ and each $j \in \{1 \ldots m\}$ Bob performs the following operations:
   (a) Bob forms the pair $B_{i,j} = (b_i, b_j)$.
   (b) Bob rotates and translates $B_{i,j}$, such that $b_i = (0, 0, t_b, \phi_b)$ and $b_j = (x'_b, 0, t'_b, \phi'_b)$.
   (c) If $t_a = t_b$ and $t'_a = t'_b$, Bob computes a score $s_{i,j}$ for the minutiae pair as $s_{i,j} = (x'_a - x'_b)^4 + (\phi_a - \phi_b)^2 + (\phi'_a - \phi'_b)^2$.
8. Bob picks the pair B_{min} with the minimum score and accordingly rotates and translates S_B into S'_B.

4.2   Fingerprint Comparison

The number of minutiae per aligned template varies between the two parties and the minutiae are not ordered, such that their indices restrict the number of possible matches. Alice and Bob perform the following steps before they proceed to compare their fingerprint templates.

Fingerprint Comparison Preparation

Input: An aligned template S'.
Output: A rasterized template S''.

1. Each aligned template S' will be divided into equal-sized squares $\boldsymbol{Squ} = (squ_1, \ldots, squ_\sigma)$.
2. Each square squ_i is mapped to a bit of S":

$$S''_i = \begin{cases} 0 \text{ if } squ_i \text{ does not contain any minutiae} \\ 1 \text{ if } squ_i \text{ contains at least one minutia} \end{cases}$$

Alice computes her rasterized template S''_A from S'_A and Bob computes S''_B from S'_B. The number of matching minutiae n_{match} can be computed from the Hamming distance d_{Ham} between S''_A and S''_B:

$$n_{match} = \frac{|S_A| + |S_B| - d_{Ham}(S''_A, S''_B)}{2}$$

The Hamming distance can computed privately using the semantically secure, homomorphic, public-key encryption scheme by [8]. For brevity of discussion details are left to the reader.

5   Private Fingerprint Verification

We present an extension of the protocol where Bob does not obtain fingerprint information. He stores a hidden template and uses it to verify Alice's identity. Alice only needs to remember a short PIN.

Enrollment Phase

Input: Alice has her fingerprint template S_A, the minutiae pairs $\boldsymbol{p}$ and PIN pin.
Output: Bob has an enrolled template for Alice.

1. Alice generates $r_1, \ldots, r_n$ where $r_i = H(pin, i)$.
2. For each $i \in 1, \ldots, n$ Alice performs the following steps:

    (a) Alice aligns S_A to p_i as in step 5 of the Fingerprint Alignment Protocol obtaining aligned template S_i'.
    (b) Alice prepares $\boldsymbol{v}_i$ from S_i' using the Fingerprint Comparison Preparation algorithm.
    (c) Alice computes the XOR of r_i and v_i: $h_i = r_i \oplus v_i$.

3. Alice sends $p_1, \ldots, p_n$ and $h_1, \ldots, h_n$ to Bob.

During verification Alice enters her PIN and has her fingerprint captured by a sensor. Bob sends the stored minutiae pair information to Alice and Alice aligns her fingerprint before she applies the hiding procedure. Then Bob and Alice engage in the Hamming distance protocol using the hidden bit vectors.

6   Evaluation

6.1   Privacy and Security

In this section we will show that for an attacker who is honestly participating in the protocol, it is not possible to reconstruct enough fingerprint information from his view of the protocol to successfully forge a matching template, even if the protocol is run multiple times.

In the Fingerprint Alignment Protocol one party sends pairs of minutiae to the other party. These pairs have been randomized by rotating and translating them. Therefore the information leaked is only their distance, the difference of their ridge orientation and their type. The protocol ensures that no two pairs have any minutiae in common. The Fingerprint Verification Protocol reveals the same pairs of minutiae to the identifying party each time preventing the combination of pairs from multiple runs. An attacker needs to guess the absolute position and ridge orientation of half of the minutiae.

The Fingerprint Comparison reveals the Hamming distance between two rasterized templates and the number of minutiae in the other party's template. An attacker can try to obtain a rasterized template by guessing the bit vector and
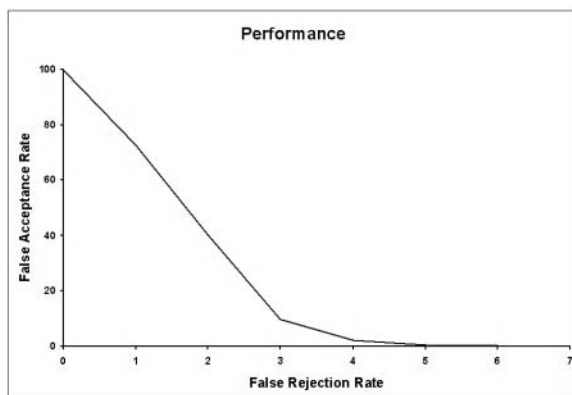
Fig. 1. Matching Performance

in the Fingerprint Verification Protocol also the user's PIN. Let l be the number of bits in the rasterized template, δ the maximum Hamming distance for a successful match and k the number of digits of the PIN, then the number of bits $b_{verification}$ he needs to guess is:

$$b_{verification} = \log_2 \frac{\binom{l}{n} \cdot 10^k}{\sum_{i=0}^{\delta} \binom{l}{i}}$$

For 36 minutiae, a 512-bit rasterized template, a maximum Hamming distance of 31 and a 5 digit PIN, this equals to about 46 bits.

6.2 Performance

We have implemented a prototype version of the comparison protocol in 2001. Since then results and data of performance competitions have been published [12, 13]. This section summarizes our performance evaluation showing the practicality of our matching algorithm.

We generated 108 fingerprint images – 27 "fingers" with 4 prints each – using an automated tool [3]. We used the minimum distance of 10 protocol runs. 1000 non-matching and 100 matching randomly selected pairs were privately compared.

Figure 1 shows our results. The false rejection rate (FRR) is the percentage of fingerprint pairs that have been detected as not matching although they were from the same finger. The false acceptance rate (FAR) is the percentage of pairs that have been incorrectly detected as a match. We varied the maximum Hamming distance to gather the plotted measurements. Our prototype performed closest to the equal error rate at nine matching minutiae for a match.

7   Conclusion

We have presented a Private Fingerprint Verification protocol analyzed its security and experimentally evaluated its matching performance. We plan to modify and extend this protocol and its implementation to address problems of active attacks, live fingerprint capture and comparing other biometrics.

References

[1] C. Soutar, D. Roberge, A. Stoianov, R. Gilroy, and B. V. K. V. Kumar "Biometric Encryption" In ICSA Guide to Cryptography, eds. R. K. Nichols, McGraw-Hill Publishers, 1999 388

[2] J. L. Cambier, U. C. von Seelen, M. Braithwaite, R. Moore, R. Glass and I. Scott "Application-Dependent Biometric Templates" BC14, The Biometric Consortium Conference, 2002 388

[3] R. Cappelli, A. Erol, D. Maio and D. Maltoni "Synthetic Fingerprint-image Generation" ICPR 2000, Proceedings International Conference on Pattern Recognition, 2000 392

[4] G. I. Davida, B. J. Matt, R. Peralta and Y. Frankel "On the relation of error correction and cryptography to an off line biometric based identification scheme" WCC99, Workshop on Coding and Cryptography, 1999 388

[5] W. Du and M. J. Atallah "Protocols for Secure Remote Database Access with Approximate Matching" ACMCCS 2000, 7th ACM Conference on Computer and Communications Security, The First Workshop on Security and Privacy in E-Commerce, 2000 388

[6] J. Feigenbaum, Y. Ishai, T. Malkin, K. Nissim, M. J. Strauss, and R. N. Wright "Secure Multiparty Computation of Approximations" (Extended Abstract) ICALP 2001, 28th International Colloquium on Automata, Languages and Programming, pp. 927-938, 2001 388

[7] O. Goldreich "Secure Multi-Party Computation" Manuscript, http://www.wisdom.weizmann.ac.il/ oded/pp.html, 2002 388

[8] S. Goldwasser and S. Micali "Probabilistic Encryption" Journal of Computer and System Sciences, Vol. 28(2), 1984 390

[9] A. K. Jain, L. Hong and R. Bolle "On-line Fingerprint Verification" IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19(4), pp. 302-314, 1997 388

[10] A. Juels and M. Wattenberg "A Fuzzy Commitment Scheme" CCS'99, Proc. of the 6th ACM Conference on Computer and Communications Security, 1999 388

[11] T. Matsumoto, H. Matsumoto, K. Yamada and S. Hoshino "Impact of Artificial "Gummy" Fingers on Fingerprint Systems" SPIE Vol #4677, Optical Security and Counterfeit Deterrence Techniques IV, 2002 387

[12] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman and A. K. Jain "FVC2000: Fingerprint Verification Competition" IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24(3), pp. 402-412, 2002 392

[13] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman and A. K. Jain "FVC2002: Second Fingerprint Verification Competition" ICPR 2002, Proc. of International Conference on Pattern Recognition, 2002 392

[14] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar "Handbook of Fingerprint Recognition" Springer Verlag, June 2003 388

[15] S. Prabhakar, A. K. Jain, and S. Pankanti "Learning Fingerprint Minutiae Location and Type" Pattern Recognition, Vol. 36, No. 8, pp. 1847-1857, 2003   388

[16] N. Ratha, J. Connell and R. Bolle "Enhancing security and privacy in biometrics-based authentication systems" IBM Systems Journal, Vol. 40(3), pp. 614-634, 2001   388

[17] A. Ross, A. K. Jain and J. Reisman "A Hybrid Fingerprint Matcher" Proc. of International Conference on Pattern Recognition (ICPR), Vol.3, pp. 795-798, 2002   388

[18] A. Yao "Protocols for Secure Computations" FOCS '82, Proc. of the Twenty-third IEEE Symposium on Foundations of Computer Science, pp. 160-164, 1982   388

Multi-unit Biometric Fusion
in Fingerprint Verification

Kangrok Lee[1], Kang Ryoung Park[2], Sanghoon Lee[1], and Jaihie Kim[1]

[1] Department of Electrical and Electronic Engineering, and of BERC
Yonsei University
134, Sinchon-dong Seodaemun-gu, Seoul 120-749, Korea
{plusu,hoony,jhkim}@yonsei.ac.kr
[2] Division of Media Technology
SangMyung University,
7, Hongji-dong, Chongro-gu, Seoul 110-743, Korea
{parkgr}@smu.ac.kr

Abstract. The fingerprint verification system using many fingers is more efficient and reliable than using a single finger. Thus, the multiple use of several fingers and multiple impressions of the same finger can lead to an increase in accuracy. However, it will cause further inconvenience to the user and increase the overall verification time. Therefore, we combine only two fingers (multi-unit fusion) to improve the performance of the fingerprint verification system considering the processing time. We show that the performance can be improved by selecting a better quality fingerprint image of two fingerprints at a preprocessing stage. Also, we propose a new quality checking algorithm based on the directional field. Our experimental results show that when the better fingerprint selection is performed using the quality checking algorithm, there is a significant improvement in the performance of the fingerprint verification system.

1 Introduction

Fingerprint verification is one of the most popular personal identification methods[1]. The fingerprint verification system has the advantage that fingerprint verification is much more reliable than other kinds of personal identification methods based on signature, face, and speech[2]. However, due to the small size of solid state sensors, only a part of the finger tip is captured in the image, and as a result the system performance is diminished. In addition, because of the various sources of noise associated with image acquisition and the vulnerability of the feature extractor to noise and distortion in the fingerprint image, it is very difficult to achieve a desirable false rejection rate when the specified false acceptance rate is very low. To solve those problems, it is necessary to combine more than two fingers or multiple impressions of the same finger for an increase in accuracy. Jain et al. proposed a classifier combination at the decision level and emphasized the importance of classifier selection in the classifier combination[3]. J. Daugman insisted that a biometric system using only a strong biometric in
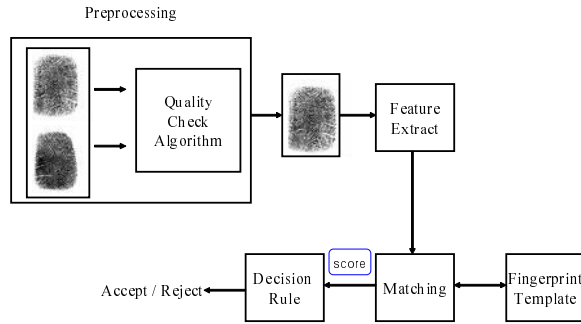
Fig. 1. Fingeprint verification system using multiple fingerprint units

the classifier level is superior to using biometrics in the respect of both the FAR(False Acceptance Rate) and the FRR(False Rejection Rate)[4]. However, he does not consider the cases of the bad condition of input data. The biometric identifiers are not always sensed by a practical fingerprint verification system. For instance, a small fraction of the population may have fingerprints which are not easily captured by a sensor adopted by the system. Therefore we propose that the performance of the verification system can be improved by selecting a better quality fingerprint image of two fingerprints at the preprocessing stage. Also, we prove theoretically and empirically that the proposed method is more accurate than the method using an AND rule or an OR rule and a single finger in the respect of both the FAR and the FRR.

2   New Method for the Performance Improvement in the Fingerprint Verification

First of all, we evaluate the performance by using fingerprint images of two different fingers. In Fig. 1, we use two fingerprint (a right thumb and a left thumb) images acquired by a sensor sequentially. In real-time fingerprint verification systems, it does not take much time to obtain two fingerprint images. Next, we try to select the better fingerprint image of two fingerprint images using the quality checking algorithm. Then feature extraction using the good fingerprint image is followed by matching in the fingerprint verification system. At the matching stage, we compare the minutiae points extracted from the good fingerprint image with those in the fingerprint template, and calculate a matching score. Occasionally, we use decision rules to evaluate the performance improvement. This method can be proven as follows. If biometric tests will be considered as F1 and F2, each biometric test is characterized by its own pair of error rates at a given operating point, denoted as the error probabilities $P_{F1}(FA)$, $P_{F1}(FR)$, $P_{F2}(FA)$, $P_{F2}(FR)$. That is, the fingerprint images of the left and the right thumbs are captured at the first and the second trial, respectively. At the first trial, $P_{F1}(FA) = P_{F1}(FR) = P$ and at the second trial, $P_{F2}(FA) =$

$P_{\mathrm{F2}}(FR)$ = P in case of good data, $P_{\mathrm{F2}}(FR)$ = Q, $P_{\mathrm{F2}}(FA)$ = P in case of bad data. Where $P < Q, P < 1, Q \leq 1, Q = tP$ (t > 1). In upper cases, the bad data mean the case that the input fingerprint image is not good enough to be identified due to the excessive humidity of the sensor surface or the incomplete fingerprint input by a beginner, etc. If there are genuine tests of numbers (the number of good data: X, the number of bad data: $M - X$) and imposter tests of numbers (the number of good data: Y, the number of bad data: $M - Y$), then we can calculate a total of errors at each case. The Total Error Cases (TEC) is followed by,

Case (1) Using the only left thumb at the first trial:

$$\mathrm{TEC} = M \times P + M \times P = 2MP$$

Case (2) Using the only right thumb at the second trial:

$$\mathrm{TEC} = X \times P + (M - X) \times Q + M \times P = (M + X)P + (M - X)Q$$

Case (3) Using an OR rule:

$$\mathrm{TEC} = XP^2 + (M - X)PQ + M(2P - P^2)$$

Case (4) Using an AND rule:

$$\mathrm{TEC} = X(2P - P^2) + (M - X)(P + Q - PQ) + MP^2$$

Case (5) Using the proposed method:

$$\mathrm{TEC} = M \times P + M \times P = 2MP$$

We prove that the TEC of the proposed method (case(5)) is smaller than the TEC of the other method (case(1) - case(4)). To compare case(2) with case(5), we get the difference function $(f(P, Q, M, X))$ between the TEC of case(2) and that of case(5) like Eq. (1).

$$
\begin{aligned}
f(P, Q, M, X) &= \mathrm{TEC\ of\ case\,}(2) - \mathrm{TEC\ of\ case\,}(5) \\
&= (M + X)P + (M - X)Q - 2MP \\
&= P(1 - t)(X - M)
\end{aligned}
\tag{1}
$$

Here, from the conditions of $f(P, Q, M, X) = P(1 - t)(X - M) \geq 0$ and $XP + (M - X)Q \geq MP$, the proposed method is superior to the method using a single fingerprint image. Only when all the fingerprint images show good quality at genuine tests $(X = M)$, the performance using a single fingerprint is equal to that of the proposed method. However, in a real fingerprint verification system, it is impossible. Due to the excessive humidity, dust or sweat of the sensor surface, the case of bad quality images happens frequently $(M - X > 0)$. Therefore, we conclude that the proposed method is superior to the method using a single fingerprint image. Continuously, to compare case(3) with case(5), we get the difference function $(f(P, Q, M, X))$ between the TEC of the case(3) and that of the TEC of the case(5) like Eq. (2).

$$
\begin{aligned}
f(P, Q, M, X) &= \mathrm{TEC\ of\ case\,}(3) - \mathrm{TEC\ of\ case\,}(5) \\
&= XP^2 + (M - X)PQ + M(2P - P^2) - 2MP \\
&= (M - X)P^2(t - 1)
\end{aligned}
\tag{2}
$$

Here, from the conditions of $f(P, Q, M, X) = (M - X)P^2(t - 1) \geq 0$ and $XP^2 + (M - X)PQ + M(2P - P^2) \geq 2MP$, the proposed method is superior to the method using an OR rule. Also, when we compare case(4) with case(5), we know the proposed method is superior to the method using an AND rule. Therefore, we conclude that the proposed method using the good fingerprint image at the preprocessing stage is more efficient and accurate than the method using an AND or an OR rule.

3   Quality Checking Algorithm

In this section, we explain the quality checking algorithm in the context of a local and global fingerprint image. First of all, we use the directional field to examine the quality of a fingerprint image. Also, we use the gradient-based methods to estimate the directional field from a fingerprint. The gradient-based methods are more accurate and efficient than any other method[5]. The gradient vector $[G_x(x, y) \ G_y(x, y)]^T$ is defined as:

$$\begin{bmatrix} G_x(x, y) \\ G_y(x, y) \end{bmatrix} = \nabla I(x, y) = \begin{bmatrix} \frac{\partial I(x,y)}{\partial x} \\ \frac{\partial I(x,y)}{\partial y} \end{bmatrix} \tag{3}$$

where $I(x, y)$ represents the gray-scale image.

3.1   Quality Checking of a Local Fingerprint Image

In the fingerprint image, the difference of local characteristics by the quality of the fingerprint image is showed by Fig. 2(b), (d). In Fig. 2(d), a block of good quality has uniform direction of the ridge structure. But a block of bad quality has irregular direction of the ridge structure in Fig. 2(b). Bazen et al. used the squared gradients for computation of the strength of the orientation[5]. The coherence is a measure for the local strength of the directional field [6]. This measurement, which is called the coherence (Coh like Eq.(4)), presents how well all squared gradient vectors share the same orientation. Using those characteristics, we examine the quality of the fingerprint image by calculating the coherence of a pixel gradient within a block of image pixel (16 * 16 pixels) in Fig. 2(c), (e). Coherence is the quality measurement of a local fingerprint image (Q_C)[5].

$$Coh = \frac{\sqrt{(G_{xx} - G_{yy})^2 + 4G_{xy}^2}}{G_{xx} + G_{yy}} \tag{4}$$

where

$$G_{xx} = \sum_W G_x^2 \ , \ G_{yy} = \sum_W G_y^2 \ , \ G_{xy} = \sum_W G_x G_y$$

are estimates for the variances and crosscovariance of G_x and G_y, averaged over the window W.
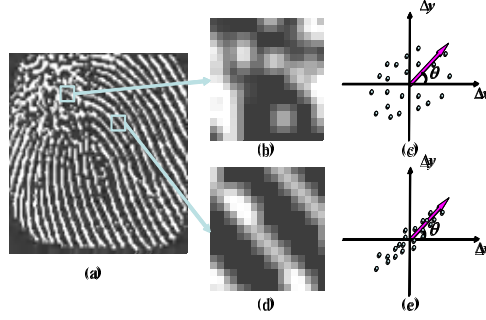
Fig. 2. Quality checking method using local characteristics: (a) input fingerprint image, (b) a block of bad quality, (c) gradient distribution of the block (b), (d)a block of good quality, (e) gradient distribution of the block (d)
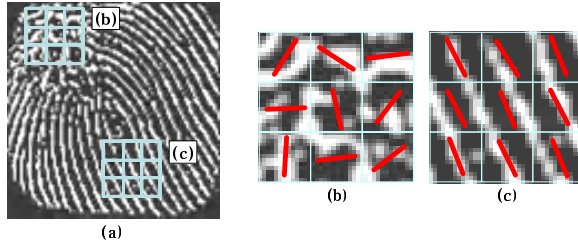
Fig. 3. Quality check using direction variance of the adjacent block: (a) input fingerprint image, (b) an area of bad quality, (c) an area of good quality

3.2   Quality Checking of a Global Fingerprint Image

Generally, a fingerprint has the characteristic that the flows of a ridge vary slowly. No variance of orientation among adjacent blocks exists in the area of good quality in Fig. 3(c). On the contrary, a large variance by no coherence ($Coh = 0$) among adjacent blocks exists in the area of bad quality in Fig. 3(b), Using these characteristics, we calculate the quality of a global fingerprint image by circular variance among adjacent blocks[7]. The method calculating the quality measurement (Q_O) of a global fingerprint image is followed by Eq.(5),

$$Q_O = 1 - V \tag{5}$$

where,

$$V = 1 - \sqrt{C^2 + S^2} : Circular\ variance$$
$$\bar{C} = \frac{1}{W}\sum_{j=1}^{W}\cos\theta_j\ ,\ \ \bar{S} = \frac{1}{W}\sum_{j=1}^{W}\sin\theta_j$$

3.3   Classifying Method of a Local and Global Fingerprint Image

To consider the local and global fingerprint characteristics, we examine whether a relation between the local characteristic and the global characteristic is independent or not. To do this, we calculate the covariance matrix. Diagonal components of this matrix are not bigger than off-diagonal components of it. Thus, the relation between a local characteristic and a global characteristic is not independent. It is assumed that both a local quality measurement and a global quality measurement have two dimensional Gaussian random distributions.

$$p(X|W_i) = \frac{1}{2\pi \times |\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(X-\mu)^t \Sigma^{-1}(X-\mu)\right] \tag{6}$$

where $X = \begin{bmatrix} Q_C\, Q_O \end{bmatrix}^T$, $\mu = \begin{bmatrix} \mu_C\, \mu_O \end{bmatrix}^T$ and Q_C is local quality measurement, Q_O is global quality measurement, μ_C is local mean of quality, μ_O is global mean of quality. W_0 is the class of the good quality fingerprint images and W_1 is the class of the bad quality fingerprint images. We use the same classification method as in the Bayesian Theorem in (7) to minimize errors, when a block in the fingerprint image judges the quality of a fingerprint image by using each quality measurement.

$$\text{If } p(W_i|X) > p(W_j|X) \quad \text{then} \quad \text{select } W_i \tag{7}$$

If the number of a good block is higher than the threshold value (Th = 304) out of 352 blocks in the fingerprint image, we decide that the input fingerprint image is a good quality image.

4   Experimental Results

A set of fingerprint images was acquired through the fingerprint sensor manufactured by Testech, Inc. and 1,140 (20 fingerprints for 57 people) fingerprint images with various image qualities. Two of the 10 fingerprints were enrolled while others were tested. From that, we had 912 authentic tests and 58,026 impostor tests. In Fig. 4(a), the proposed method shows better performance than using a single finger. In Fig. 4(b), the proposed method is superior to that using an AND rule or an OR rule. Also, we assure that the fingerprint verification system using the proposed method showed better performance in comparison with others in terms of both the FRR and the FAR as shown in Table 1.

Table 1. False rejection rates on the test with different values of false acceptance rates

FA(%)	FR1(%) (left finger)	FR2(%) (right finger)	FR3(%) (AND rule)	FR4(%) (OR rule)	FR5(%) (new method)
1.0	6.3	6.9	14.5	5.7	5.8
0.1	11.6	11.8	20.6	10.0	7.2
0.01	15.8	16.2	25.8	14.1	10.4
0.001	18.5	20.6	31.7	19.1	18.3

(a)   (b)
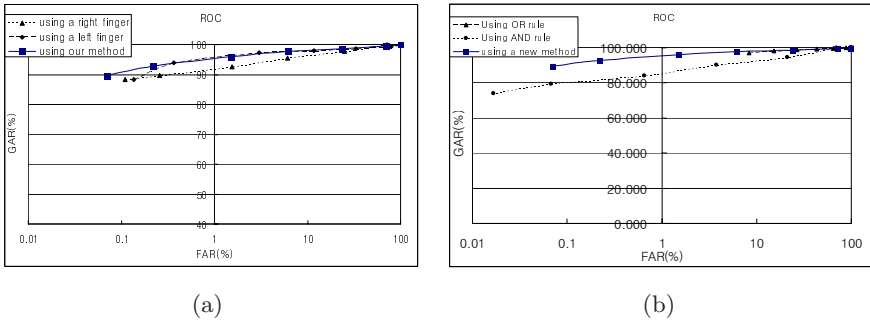
Fig. 4. ROC curves for the proposed method, a single finger, AND, OR rule: (a) Using the proposed method vs. a single finger, (b) Using the proposed method vs. AND, OR rule

5   Conclusion and Future Work

We have designed and implemented the fingerprint verification system which operates by two fingerprint images. We show that a considerable improvement in fingerprint verification performance can be achieved selecting a better image of two fingerprint images. In addition, we prove theoretically and empirically that the proposed method is better than the method using an AND rule or an OR rule and a single finger. Future experiments will include developing biometric fusion at the feature extraction level or at the matching score level.

Acknowledgements

References

[1] A. Jain, L. Hong, and R. Bolle: On-line fingerprint verification. IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 19, no. 4, pp.302-314, April 1997. 395

[2] H. C. Lee and R. E. Gaensslen: Advances in Fingerprint Technology. Elsevier, New York, 1991. 395

[3] S. Prabhakar and A. K. Jain: Decision-level Fusion in Fingerprint Verification. Pattern Recognition, Vol. 35, no. 4, pp.861-874, 2002. 395

[4] J. Daugman: Biometric decision landscapes. Technical Report No. TR482, University of Cambridge Computer Laboratory, 2000. 396

[5] Asker M. Bazen and Sabin H. Gerez: Systematic Methods for the Computation of the Directional Fields and Singular Points of Fingerprints. IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 24, no. 7, pp.905-919, July 2002. 398

[6] M. Kass and A. Witkin: Analyzing Oriented Patterns. Computer Vision, Graphics, and Image Processing, Vol. 37, no. 3, pp.362-385, March 1987.   398

[7] K. V.Mardia and P. E.Jupp: Directional Statistics. John Wiley Sons Ltd, 2000. 399

Fast Separable Gabor Filter
for Fingerprint Enhancement

Vutipong Areekul[1], Ukrit Watchareeruetai[1],
and Sawasd Tantaratana [2]

[1] Kasetsart Signal and Image Processing Laboratory (KSIP Lab)
Department of Electrical Engineering
Faculty of Engineering
Kasetsart University
50 Phaholyothin Road, Jatujak, Bangkok 10903, Thailand
`vutipong.a@ku.ac.th,w_ukrit@yahoo.com`
[2] Department of Electrical Engineering
Sirindhorn International Institute of Technology (SIIT)
Thammasat University
Patumthani 12121, Thailand
`sawasd@siit.tu.ac.th`

Abstract. Since two-dimensional Gabor filter can be separated into one-dimensional Gaussian low pass filter and one-dimensional Gaussian band pass filter to the perpendicular, a new set of separable Gabor filters are implemented for fingerprint enhancement. This separable Gabor filtering consumes approximately 2.6 time faster than the conventional Gabor filtering with comparable enhancement results. This alternative fingerprint enhancement scheme is very promising for practical fast implementation in the near future.

1   Introduction

Goal of fingerprint enhancement is to make ridges clear and genuine as much as possible. The successful enhancement process is used in conjunction with feature extraction process resulting in genuine minutiae and other important features. These original features are very important for fingerprint matching performance. However, fingerprint enhancement process could introduce some artifacts and noise, which leads to fault and missing minutiae, degrading verification performance.

The most popular fingerprint enhancement technique employs contextual filters [1] which their characteristics adapted depending on local context. These filters are designed to make ridges clearly differentiated from each another. Moreover, they also connect broken ridges, fill pores and remove noise and dust. Basically, these filters provide a low-pass along the ridge direction in order to link small ridge gabs and reduce some pores and noise. On the other hand, these filters perform band-pass in the direction orthogonal to the ridges in order to separate ridges from each another. One of the most used contextual filter for fingerprint enhancement is Gabor filter, which proposed by Hong, Wan, and Jain [2].

In this paper, Hong, Wan, and Jain's work [2] was extended by introduced the less computational complexity for Gabor filtering process. In section 2, the separable Gabor filter and realization is presented. Then experimental results and performance comparison between separable Gabor filter and conventional Gabor filter are shown in section 3.

2   Separable Gabor Filter and Practical Implementation

Gabor filters have both frequency-selective and orientation-selective properties. They also have optimal joint resolution in both spatial and frequency domains [1]. Equation (1) shows the 2-Dimensional (2-D) Gabor filter form [1] as follows,

$$G(x, y, \theta, f_0) = \exp\left\{-\frac{1}{2}\left(\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right)\right\} \cos(2\pi f_0 x_\theta),$$

(1)

$$\begin{bmatrix} x_\theta \\ y_\theta \end{bmatrix} = \begin{bmatrix} \sin\theta & \cos\theta \\ -\cos\theta & \sin\theta \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix},$$

where θ is the orientation of the filter, f_0 is the ridge frequency, $[x_\theta, y_\theta]$ are the coordinates of $[x, y]$ after a clockwise rotation of the Cartesian axes by an angle of $(90°-\theta)$, and σ_x and σ_y. are the standard deviations of the Gaussian envelope along the x- and y-axes, respectively.

Consider 2-D Gabor filter in equation (1), if θ is equal to 90°, equation (1) is simplified as follows;

$$G(x, y, f_0) = \exp\left\{-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right\} \cos(2\pi f_0 x)$$

(2)

The Gabor filter at orientation $\theta = 90°$ of equation (2) can be separated into two independent functions as:

$$G(x, y, f_0) = G_{BP}(x, f_0)G_{LP}(y) = \exp\left\{-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}\right)\right\}\cos(2\pi f_0 x)\exp\left\{-\frac{1}{2}\left(\frac{y^2}{\sigma_y^2}\right)\right\}$$

(3)

where $G_{BP}(x, f_0) = \exp\left\{-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}\right)\right\}\cos(2\pi f_0 x)$, and $G_{LP}(y) = \exp\left\{-\frac{1}{2}\left(\frac{y^2}{\sigma_y^2}\right)\right\}.$

G_{BP} in Equation (3) is only a band pass Gaussian function of x axis, and G_{LP} in equation (3) is only a low pass Gaussian function of y axis. Because f_0 can be determined by ridge frequency, f_0 can be obtained as a constant. Therefore equations (3) are only functions with one variable (x or y variable) or 1-Dimensional (1-D) Gabor filters. Clearly, this Gabor filter at $\theta = 90°$ can be implemented as a separable transform [3]. By first apply 1-D band pass Gabor filter (G_{BP}) convolution along each

row of a fingerprint block and then follow by 1-D low pass Gabor filter (G_{LP}) convolution along each column of this fingerprint block image.

In general, the first step of applying conventional 2-D Gabor filter for fingerprint enhancement is to partition a fingerprint image into square sub-blocks, size $M{\times}M$, where $M = 16$ in our experiment. Each block was searched and calculated Gabor filter parameters such as its orientation (θ) and its ridge frequency (f_0). The orientation and ridge frequency of any fingerprint block were found by implementing Hong's method [2] for our approach. Once orientation and ridge frequency were found, the Gabor filter was set and designed to perform 2-D convolution operation. The window $N{\times}N$ of conventional 2-D Gabor filter was convolved along rows and columns of fingerprint image; i.e. multiplied all pixels in the overlapped windows and summed to obtain a window center pixel in the enhanced image. Clearly, this process needs high computational complexity in general.

Instead of applying 2-D convolution, a separable 1-D band pass Gabor filter, size N, can be applied to convolve along the ridge direction θ for all pixels in the block. Then, a 1-D low pass Gabor filter, size N, is convolved along the perpendicular to the ridge direction θ for all pixels in the block. Unfortunately, 1-D pixel sequence along the ridge direction θ is not well-defined due to square sampling grid pattern. For example, if a straight line in any chosen direction is drawn on an image, it is difficult to pick consecutive pixels in order to form a straight line in that particular direction. Hence, separable approach needs resampling process in order to get a precise sequence of pixels. This makes the separable approach is more complicate.

Fortunately, in practical conventional Gabor enhancement, ridge direction is quantized into 8 directions; i.e. 0°, 22.5°, 45°, 67.5°, 90°, 112.5°, 135°, 157.5° [1],[2]. For practical separable Gabor Enhancement, direction of filter is shaped by sampling grid and tessellated pixels, resulting in approximate eight directions i.e. 0°, 26.6°, 45°, 63.4°, 90°, 116.6°, 135°, 153.4°. Obviously, for these eight particular directions, locations of pixels in a sequence are well-defined and tessellated as shown in Fig.1, hence it is possible to implement a separable 2-D Gabor transform without additional computational complexity. However, since a fingerprint image is sampled within square grid as shown in Fig.1, a space between sample to the next sample in any direction may be varied. For example, assume that distance between sample to sample, "a", for 1-D filter with orientation 0° and 90° is equal to 1 unit. Then, a distance between two samples in direction of 45° and 135° is equal to $\sqrt{2}$ unit and a distance between two samples in direction of 26.6°, 63.4°, 116.6°, and 153.4° is equal to $\sqrt{1.2}$ unit. Hence equation (3) should be generalized by adding a distance parameter "a" as the following equation,

$$G_{BP}[n]=\exp\left\{-\frac{1}{2}\left(\frac{(an)^2}{\sigma_x^2}\right)\right\}\cos(2\pi f_0 an) \quad \text{and} \quad G_{LP}[n]=\exp\left\{-\frac{1}{2}\left(\frac{(an)^2}{\sigma_y^2}\right)\right\} \tag{4}$$

where n represents discrete pixel in 1-D direction, and a represents a distance between pixel to pixel in particular direction; i.e. $a = 1$ for 0° and 90°, $a = \sqrt{2}$ for 45° and 135°, and $a = \sqrt{1.2}$ for 26.6°, 63.4°, 116.6°, and 153.4° orientations.
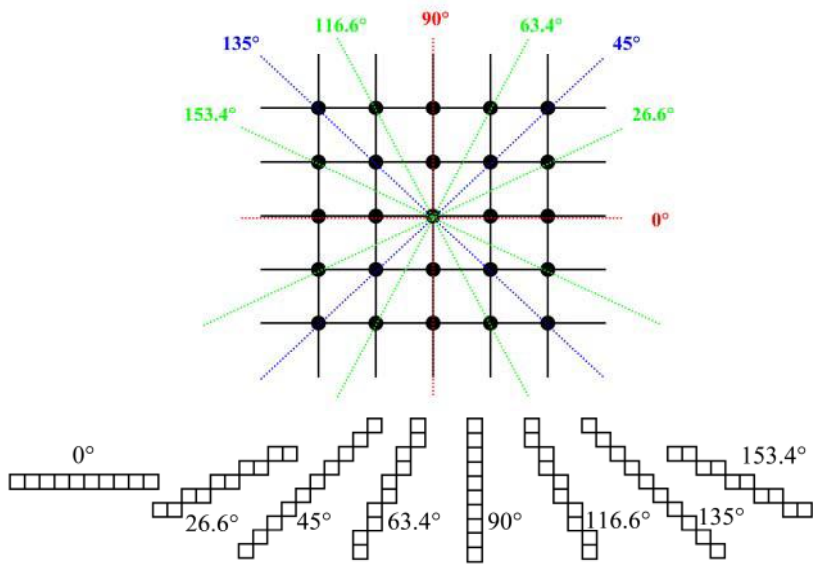
Fig. 1. The square grid samples of an image with eight selected orientations, and the example of practical eight-direction 1-dimensional filter

The example of the practical separable Gabor filter implementation with 0° orientation is shown in Fig.2. For this example, 8×8 block size is used (*M*=8) and 1-D Gabor filter length is 13 (*N*=13). To perform a separable filtering, the convolution process is divided into two phases. At the first phase, a 1-D Gabor band pass filter is convolved along the columns of a extended block, size (*M*+*N*-1)×(*M*+*N*-1), resulting in all gray pixels in Fig.2 (a). Note that the convolution result or gray pixels must be larger than 8×8 block size in order to obtain the correct results in the second phase. The second phase performs convolution with 1-D Gabor low pass filter along the rows of 8×8 block. The gray pixels in Fig.2 (b) are the final separable filtering result.
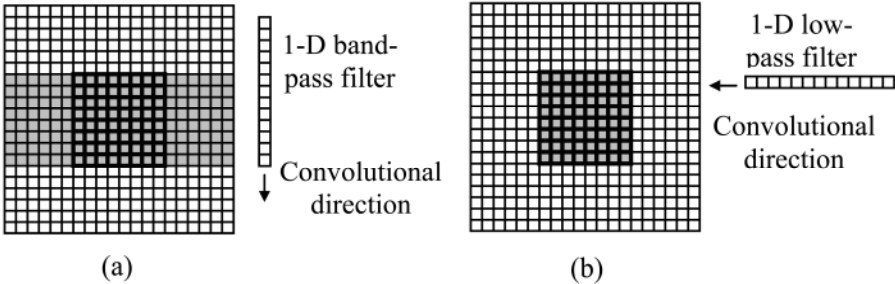
Fig. 2. Separable Gabor filter implementation performing on 0° direction with (a) column operation with band pass filter, and (b) row operation with low pass filter
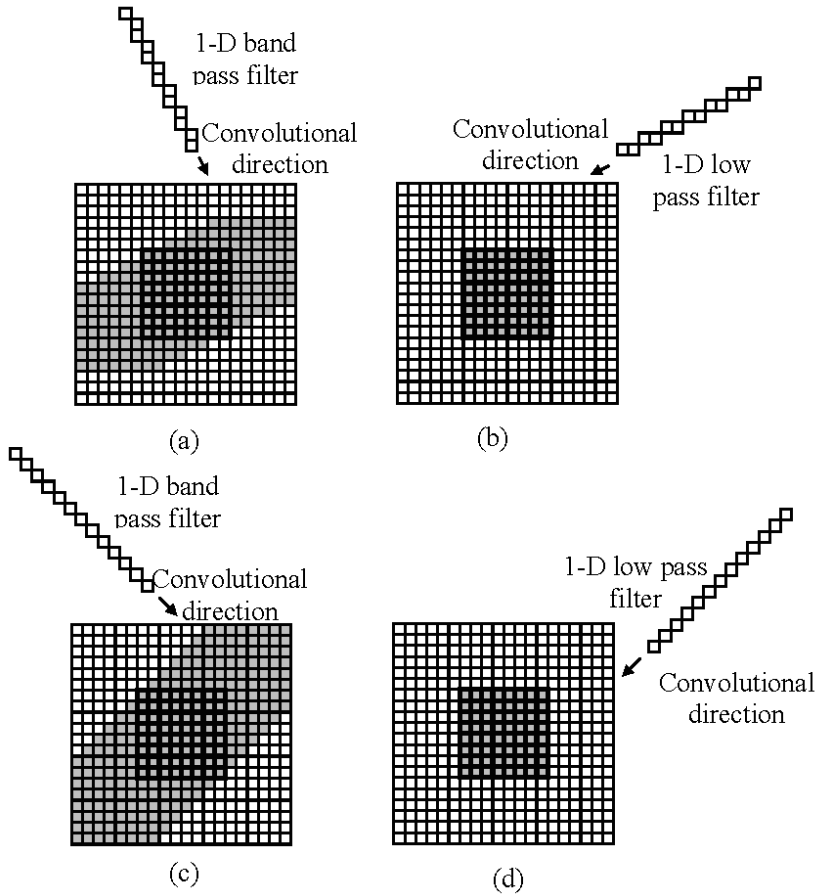
Fig. 3. Separable Gabor filter implementation performing on 26.6° and 45° direction with (a) & (c) column operation with band pass filter, and (b) & (d) row operation with low pass filter

The other examples of the practical separable Gabor filter implementation with 26.6° and 45° orientations are shown in Fig.3. Note that the gray pixels in Fig.3 represent the effective area that must be kept for the next 1-D convolution in perpendicular direction.

3   Experimental Results and Performance Comparison

The experimental results reported in this paper have been conducted on public fingerprint database, DB2A of FVC2000 [4]. The DB2A comprises 800 fingerprint images from 100 fingers, 8 impressions per each finger, with 500 dpi resolution. Three major experiments are conducted in order to compare performance between conventional 2-D Gabor filter and separable Gabor filter such as effect on computational complexity, effect on minutiae, and effect on matching results.

Table 1. The 1st Experiment: average execution time on a Pentium 4 (1.8A GHz) comparison

Average time (second) per a fingerprint image (364×256)	
2-D Conventional Gabor filter	Separable Gabor filter
0.574047 second	0.219285 second

The 1st experiment is the advantage of separable Gabor filter over conventional method. The experiments were tested on a Pentium 4 (1.8A GHz) personal computer with 256MB at 333MHz. Table 1 shows the corresponding execution times comparison between   two methods. The 2-D conventional Gabor filter was implemented on the fly using a set of priori created and stored filters [1]. Moreover, symmetric of 2-D Gabor filter was also exploited in this process. Clearly, separable transform gave less computational complexity approximately 2.6 times. Both programs were not optimized to a commercial application level and these times are shown just to give an indication of the method's speed. Gabor parameters searching algorithm such as ridge frequency f_0 and ridge orientation θ are also included and both programs shared the same routines.

The 2nd experiment related to the minutiae extraction process. The dropped minutiae, spurious minutiae, and exchanged minutiae could be obtained from enhancement outputs of two schemes. The minutiae extraction employed in this experiment using basic thinning technique with crossing ridge number [5]. Assume that conventional 2-D Gabor enhancement gave the correct minutiae. The minutiae extraction from separable enhancement gave small different results with comparison shown in Table 2.

Finally, the impact of both enhancement to the verification performance was tested in the 3rd experiment. Fingerprint verification algorithm by Xudong Jiang and Wei-Yun Yau [6] was simulated and used in conjunction with both enhancement schemes. The equal error rate (EER) performance evaluation results [4] are shown in Table 3. The conventional Gabor filtering gains over the separable Gabor filtering about 0.593976% EER. Note that verification performance using separable Gabor filtering were only marginally inferior to conventional 2-D Gabor filtering.

In conclusion, the separable Gabor filter with 8 selected orientations for fingerprint enhancement was proposed. The advantage of the separable Gabor filter over the conventional Gabor filter is less computational complexity, approximately 38.2% of conventional enhancement execution time. The disadvantage is that separable Gabor filter creates more noise and error comparing with conventional Gabor filter. However, both schemes always create spurious minutiae, dropped minutiae, and exchanged minutiae resulting in slightly different in verification performance. In conclusion, the separable Gabor filter could be a good candidate for low computational complexity in low power digital signal processor applications in the near future.

Table 2. 2nd Experiment: extracted minutiae comparison, assuming that extracted minutiae from conventional 2-D Gabor is perfect, using 800 fingerprint images of DB2A, FVC2000

Dropped minutiae (%)	Spurious minutiae (%)	Exchanged minutiae (%)
22.942208	16.296314	6.272873

Table 3. 3rd Experiment: verification performance of two enhancement approaches

Threshold (%)	Separable Gabor Filtering Enhancement		Conventional 2-D Gabor Filtering Enhancement	
	FAR (%)	FRR (%)	FAR (%)	FRR (%)
9	47.517677	10.678572	50.807137	8.821428
10	36.579231	13.964287	39.296715	11.714286
11	26.840910	17.714287	29.068180	15.035714
12	18.821339	21.428572	20.490530	18.571428
13	12.608902	24.535713	13.806187	21.892857
14	8.144255	29.857143	8.879735	25.321430
15	4.989267	33.428570	5.429924	29.000000
16	2.898043	37.250000	3.143308	33.714287
	EER ≈ 20.124955 %		EER ≈ 19.530979 %	

Acknowledgement

References

[1] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, Handbook of Fingerprint Recognition, Springer-Verlag, 2003.
[2] 2. L. Hong, Y.Wan, and A.K. Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 777-789, August 1998.
[3] 3. R. C. Gonzalez and R. E. Woods, Digital Image Processing, 2nd Edition, Prentice Hall, 2002.
[4] 4. D. Maio, D. Maltoni, R. Cappelli, J.L.Wayman, and A.K. Jain, "FVC2000: Fingerprint Verification Competition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 402-412, March 2002.
[5] 5. C. Arcelli and G.S.D. Baja, "A Width Independent Fast Thinning Algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 4, no. 7, pp. 463-474, 1984.
[6] 6. X. Jiang and W.Y. Yau, "Fingerprint Minutiae Matching Based on the Local and Global Structures," Proceeding International Conference on Pattern Recognition (15th), vol.2, pp. 1042-1045, 2000.

A Real-Time Focusing Algorithm
for Iris Recognition Camera

Kang Ryoung Park[1] and Jaihie Kim[2]

[1] Division of Media Technology
Sangmyung University
Seoul, Republic of Korea
parkgr@smu.ac.kr
[2] Dept. of Electrical and Electronic Eng.
Yonsei Univ.
Seoul, Republic of Korea
jhkim@yonsei.ac.kr

Abstract. For fast iris recognition, it is very important to capture user's focused eye image at fast speed. In previous researches and systems, they use the focusing method which has been used for general landscape scene without considering the characteristics of iris image. So, they take much focusing time, especially in case of the user with glasses. To overcome such problems, we propose a new iris image acquisition method to capture user's focused eye image at very fast speed based on the corneal specular reflection. Experimental results show that the focusing time for both the users with glasses and without glasses is average 480 ms and we can conclude our method can be used for the real-time iris recognition camera.

1 Introduction

With the increasing needs for higher security level, the biometric systems have been widely used with many applications. Among them, the iris recognition system has been in the limelight for high-security biometric applications [1][5]. For fast iris recognition, it is essential to capture user's focused eye image at fast speed. If not, the total recognition time is increased and it causes the severe inconvenience of user. In previous researches and systems [2-4][8-15], they use the focusing method which has been used for general scene (landscape or photographic scene) without considering the characteristics of iris image. In detail, they use the focusing algorithm that calculates the gray level difference between the spacing pixels. That is, if the input image is defocused, then the difference value between the pixels is low. If the image is highly clear, the difference value is high. From that, they move the focusing lens to the direction of making the difference value be the highest. However, their method can provide the wrong focusing value in case of iris image. Especially, in case of users with glasses, if the lens is positioned for focusing the glass surface or the glass frame, the scratch on the glass surface or the glass frame may make that the difference value of

the spacing pixel is highest. In general, the distance between glasses and iris is more than 1 cm and the input iris image remains blurred consequently unless the focus lens does not move. Due to those problems, the research [16] uses the method of checking the pixel difference in the region of specular reflection on a cornea. However, they use only one illuminator for checking focus value and iris recognition. In such a case, the focus checking is impossible when the specular reflection which happens on the surface of glasses hides that on a cornea. In addition, in case that many specular reflections happen from the scratch on the glass surface, it is very difficult to detect the genuine specular reflection on a cornea by that method [16]. To overcome such problems, we propose a new focusing method to capture user's focused eye image at very fast speed using corneal specular reflections with On/Off scheme of dual illuminators. Base on such a focusing algorithm, we implemented a real-time iris recognition camera.

2   A Real-Time Focusing Algorithm for Iris Recognition Camera

2.1   Auto Zooming and Focusing for Iris Recognition Camera

Due to the limitation of increasing the DOF(Depth of Field) with the single(fixed) focal camera, we use the variable focal camera for iris recogntion and auto focusing algorithm is required consequently.

For focusing algorithm, we use the specular reflection generated by IR-LED illuminator. From now on, we abbreviate the specular reflection as SR (Specular Reflection). If the Z position of user's eye is not in the DOF of the focus lens, the input eye image is defocused and the SR size becomes bigger as shown in Fig. 1(a). In such case of region I, if we make the focus lens move backward, the DOF is also moved backward and we can get the focused eye image apt for iris recognition. In addition, in case of region II, we can get the focused eye image by moving forward of focus lens and the DOF. However, it is difficult to determine the lens direction because the blurred eye images in region I, II is hard to be discriminated as shown in Fig. 1(a). Here, to be notable, the shape of SRs in both regions (I and II) are different as shown in Fig. 1(a). When the eye is positioned in the region I, the dark gray pixels exist in the edge of SR. On the contrary, the dark gray pixels exist in the center of SR in case that the eye is positioned in the region II as shown in Fig. 1(a). Based on that information, we can determine the lens direction in case of defocusing. After determining the lens direction, the movement step of focus lens should be also determined. Our preliminary test shows that we can determine the amount of lens movement step from the diameter of SR in image. In our iris camera, we use a zoom lens in addition to a focus lens. Using the zoom lens, we can maintain the iris size of input image almost same according to the Z position of user and it can make the image processing algorithm (such as iris/pupil localization) much easier. In addition, because the iris diameter is maintained almost same size, we can increase the operating range of iris recognition (from 8 to 25 cm). So, the SR size
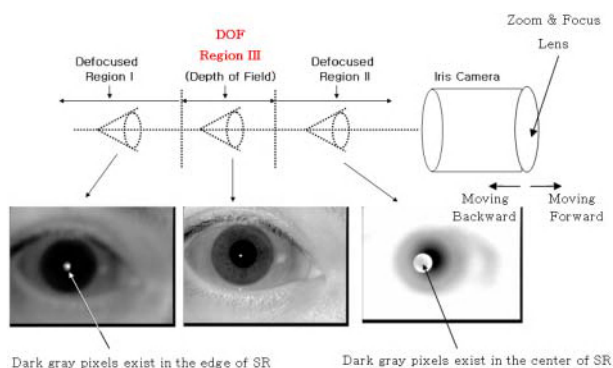
(a)
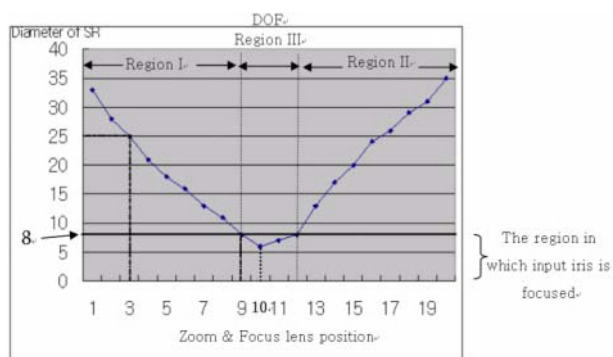
(b)

Fig. 1. The DOF region vs. the defocused region and focus lens step vs. SR diameter (a)The DOF region and the defocused region (b)Focus lens step vs. diameter of SR

is only affected by the optical defocusing and the relationship of Fig. 1(b) can be easily obtained. The zoom/focus lenses are moved according to the trace curve and the focusing range is determined based on each position of the zoom lens. The reason why we use the zoom and the focus tracing curve is to reduce the time of lens movement. Fig. 1(b) shows the experimental relationship between the zoom/focus lens position and the diameter of SR. Here, the diameter of SR (unit is pixel) is measured in the input image of 320*240 pixels. As shown in Fig. 1(b), if the diameter of detected SR is below 8 pixels, our system accepts the input iris image as the focused one.

2.2   Detecting SR in Input Image

The reason that we use the SR region for the lens focusing is that the SR can be easily detected by image processing algorithm compared to other region. However, when a user wears the glasses, some large SR on glasses surface may
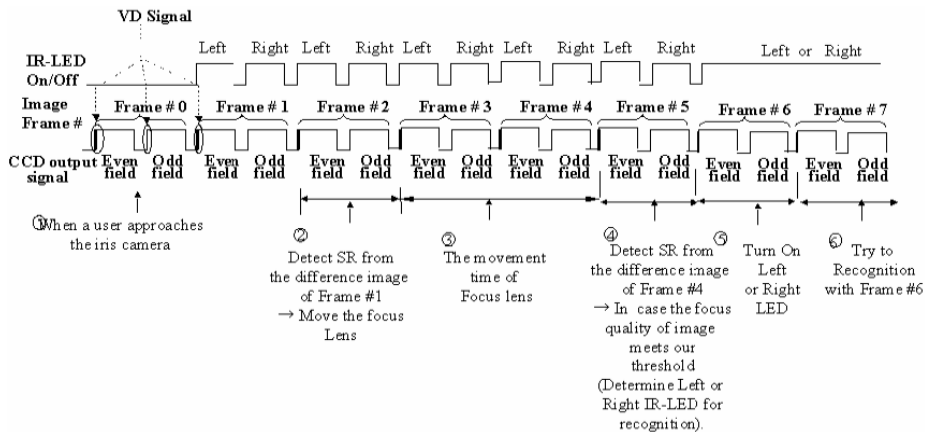
Fig. 2. The successive On/Off scheme for IR-LED illuminator

happen in an input image. In addition, if the surface of glasses is not smooth, many scratches may make a lot of small SRs. In order to detect the SR more easily in such cases, we use the method of changing the decoder value of frame grabber board. In general, the NTSC signal has high resolution $(0 - 2^{10}$-1), but the conventional decoder in frame grabber board has the ability of A/D converting with low resolution $(0 - 2^8$-1). So, the input NTSC signal cannot be fully represented and some signal range may be cut off. In this case, the NTSC signal in high saturated range is represented as $255(2^8$-1) gray level in the input image and both the genuine SR on eye (cornea) and the other reflection region on facial skin or glasses surface may be represented as same gray level(255) in the input image. However, the NTSC analog level of SR on eye is higher than that of other region such as the reflection on facial skin. That is because the reflectance rate on cornea is greater than that on facial skin. So, if we change the decoder's brightness setting (making the brightness value lower), then the A-D converting range with decoder can be shifted to the upper direction. In such case, there is no high saturated range and it is easy to discriminate the SR on eye and the other reflection. However, when a user wears the glasses, some large SR on glasses surface may still happen. In addition, if the surface of glasses is not smooth, many scratches may make a lot of small SRs. In these cases, it may be difficult to detect the genuine SR on the cornea. To overcome such problems, we use the successive On/Off scheme for IR-LED illuminators as shown in Fig. 2. When the user approaches in the operating range of the iris camera as shown in Fig. 2 (1), the distance measuring sensor can detect it and notify the user's approach to the micro-controller of camera. Then, the micro-controller controls (On/Off control) the IR-LED illuminator successively as shown in Fig. 2.

From the Fig. 2, one SR from left illuminator happen in even field and another one does in odd field. Because we know the curvature of general human cornea and the distance between left and right illuminator, we can estimate the
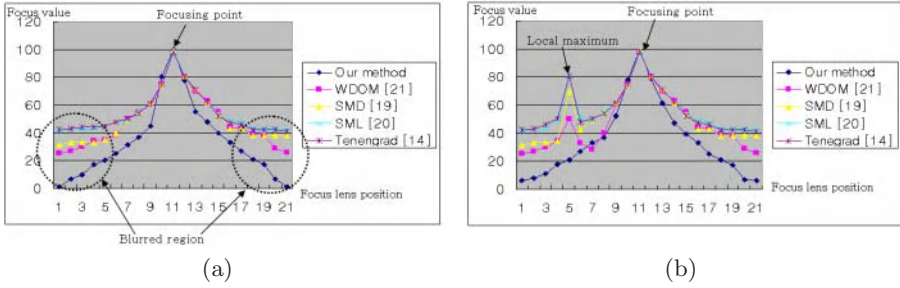
Fig. 3. Focus value vs. focus lens position (a)in case of users without glasses (b)in case of users with glasses

distance between the genuine SRs in even and odd field image. However, the other SRs (that happens on the glasses surface or the scratches of glasses) have the tendencies not to exist with the pair characteristics (or having different size in even and odd field) or the distance between each SR may be greater than that between the genuine SRs on the cornea. That is because the curvature of glasses is much smaller than that of human cornea. With the difference image of even and odd field image, we get an edge image by 3*3 sobel operator. As shown in Fig. 2, the time difference between even and odd field is only 16.7ms and the motion difference of user is very small during that time. So, almost only the edge for SR can be dominant in the edge image. In addition, due to small time difference, we do not need the time consuming procedure of motion compensation in order to reduce the motion difference and we can reduce the processing time consequently [6]. From that, we detect the center and radius of the corneal SR by 2D gradient-based circle Hough transform [7]. With this scheme, we can detect the exact SR regions on cornea and move the zoom/focus lens to the exact focusing position according to the SR size in image as shown in Fig. 2(2). From that, we can the clear and focused eye image for iris recognition at very fast speed. In addition, we can know which illuminator (left or right) makes less specular reflection on glasses surface from the detected SR as shown in Fig. 2(4) and select that illuminator for recognition in order to reduce the FRR caused by the SR on the glasses surface as shown in Fig. 2(4)(5).

3   Performance Evaluations and Discussions

The evaluation tests were performed on 350 persons (175 persons without glasses and 175 persons with glasses). Each person tried to recognize 10 times and total 3500 trial data were acquired to measure our focusing algorithm. The test data includes the persons from 23 to 60 ages, composed of 282 Korean and 68 Occidental. In addition, we collected(rearranged) the test data according to the approaching speed of user; 1000 data at normal speed (from 5cm/sec to 15cm/sec), 1000 data at fast speed (more than 15cm/sec), and 1000 data at

slow speed (below 5cm/sec). The remaining 500 data were collected in case that users approached to the camera not from the front but from the side. In the first experiment, we measured the processing time of detecting the SR in an input image and it takes a little processing time as 3 ms in Pentium-IV 1.8Ghz.

In the second experiment, we compared the performance of our focusing algorithm to those [8],[13],[14],[15] as shown in Fig. 3(a)(b). Fig. 3(a)(b) shows the focusing performance by the curve of focus value vs. focus lens position. In general, if the curve has the shape that the slope near a focusing point and that in the blurred region are maintained to be steep, it is reported that the focusing algorithm shows good performance [15]. According to Fig. 3(a), our method shows the best focusing performance. In addition, other methods show the local maximums of focus value curve which make the focusing be more difficult as shown in Fig. 3(b), but our method does not show any local maximum in focus value curve. In the third experiment, we compared the average focusing time. The focusing time of users without glasses is like these(unit is ms); 551 by Tenengrad[8], 434 by SMD[13], 535 by SML[14], 425 by WDOM[15], 328 by our method. The focusing time of users with glasses is like these(unit is ms); 1523 by Tenengrad, 928 by SMD, 1411 by SML, 890 by WDOM, 628 by our method. The focusing time of all users with/without glasses is like these(unit is ms); 1037 by Tenengrad, 681 by SMD, 973 by SML, 658 by WDOM, 474 by our method. From that, we can know our focusing method shows the best performance. In the fourth experiment, we measured the performances of our algorithm in terms of recognition speed. The average recognition time (including focusing and iris recognition time) is 722 ms in case of the users without glasses and that is 1221 ms in case of that with glasses. The reason that the recognition time is increased in the latter case is that large SR on glasses surface caused by illuminator hides the whole iris region sometimes. In such case, system turn on the other illuminator (from left to right or from right to left) and the total recognition time is increased consequently. The focusing time for both the users with glasses and without glasses is average 480 ms. In the fifth experiment, we measured the recognition rate such as FAR(0 percent) and FRR(0.8 percent (28/3500 trials)). The FRR is mainly caused by the large SR from glasses and most of them are recognized in second trial. In the sixth experiment, we tested the focusing time, recognition time and recognition rate according to the Z distance between user and the iris camera. The focusing time(ms) is like these; 482 at 8cm, 478 at 12cm, 478 at 16cm, 481 at 20cm, 480 at 25cm. The recognition time(ms) is like these; 936 at 8cm, 982 at 12cm, 976 at 16cm, 979 at 20cm, 980 at 25cm. The FAR is 0 percent at all Z distances. The FRR(percent) is like these; 0.7 at 8cm, 0.8 at 12cm, 0.8 at 16cm, 0.8 at 20cm, 0.8 at 25cm. From that, we can know the focusing time, recognition time and recognition rate are almost same according to the Z distance. In the last experiment, we tested the focusing time, recognition time and recognition rate by changing environmental lighting intensity(with fluorescent lamp). The focusing time(ms) is like these; 481 in 250 Lux., 479 in 500 Lux., 482 in 750 Lux., 481 in 1000 Lux., 475 in 1250 Lux. The recognition time(ms) is like these; 1320 in 250 Lux., 1311 in 500 Lux., 980 in 750 Lux., 981

in 1000 Lux., 978 in 1250 Lux. The FAR is 0 percent in all lighting intensity. The FRR(percent) is like these; 0.91 in 250 Lux., 0.84 in 500 Lux., 0.8 in 750 Lux., 0.8 in 1000 Lux., 0.8 in 1250 Lux. From that, we can know the focusing time, recognition time and recognition rate are almost same according to the change of lighting intensity. To be notable, in case that the lighting intensity is below 500 Lux., the FRR and the recognition speed is increased a little. That is because the pupil is dilated too much due to dark environmental light.

4   Conclusions

For fast iris recognition, it is very important to capture user's focused eye image at fast speed. In this paper, we propose a new iris image acquisition method to capture user's focused eye image at very fast speed based on the corneal specular reflection. Due to the difficulties of detecting genuine specular reflection in case of users with glasses, we use the method of changing the decoder value of frame grabber board and successive On/Off scheme for IR-LED illuminators. From the experimental results, we can conclude our method can be applicable for the real-time iris recognition camera.

Acknowledgement

References

[1] John G. Daugman, "High confidence visual recognition of personals by a test of statistical independence". IEEE Trans. PAMI., Vol. 15, No. 11, pp. 1148-1160, 1993   410
[2] http://www.lgiris.com
[3] http://www.iridiantech.com
[4] http://www.panasonic.com/cctv/products/biometrics.asp
[5] http://www.iris-recognition.org   410
[6] Ramesh Jain, "Machine Vision", McGraw-Hill International Edition, 1995   414
[7] D. Ioammou et al., "Circle Recognition through a 2D Hough transform and Radius Histogramming", Image and Vision Computing, vol. 17, pp. 15-26, 1999   414
[8] Je-Ho Lee et al., "Implementation of a passive automatic focusing algorithm for digital still camera", IEEE Trans. on CE, vol. 41, no. 3, pp. 449-454, Aug. 1995 415
[9] H. Toyoda et al., "New Automatic Focusing System for Video Camera", IEEE Transactions on Consumer Electronics, vol. CE-32, no. 3, pp. 312-319, Aug. 1986
[10] T. Haruki and K. Kikuchi, "Video Camera System Using Fuzzy Logic", IEEE Transactions on Consumer Electronics, vol. 38, no. 3, pp. 624-634, Aug. 1992
[11] K. Ooi et al., "An Advanced Auto-focusing System for Video Camera Using Quasi Condition Reasoning", IEEE Trans. on CE, vol. 36, no. 3, pp. 526-529, Aug. 1990

[12] K. Hanma et al., "Novel Technologies for Automatic Focusing & White Balancing of SolidState Color Video Camera", IEEE Trans. on CE, vol.CE-29, no. 3, pp.376-381, Aug. 1983

[13] R. A. Jarvis, "Focus Optimization Criteria for Computer Image Processing", Microscope, vol. 24(2), pp. 163-180   415

[14] S. K. Nayar and Y. Nakagawa, "Shape from Focus", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, no. 8, pp. 824-831, Aug. 1994   415

[15] Kang-Sun Choi et al., "New Auto-focusing Technique Using the Frequency Selective Weight Median Filter for Video Cameras", IEEE Trans. on CE, Vol.45, No.3, pp.820-827, Aug. 1999   415

[16] Y. Park et al., "A Fast Circular Edge Detector for the Iris Region Segmentation", LNCS, Springer Verlag, Vol. 1811, 2000, pp. 417-423   411

Improving Iris Recognition Accuracy
via Cascaded Classifiers

Zhenan Sun, Yunhong Wang, Tieniu Tan, and Jiali Cui

Center for Biometrics Authentication and Testing
National Laboratory of Pattern Recognition, Institute of Automation
Chinese Academy of Sciences
P.O. Box 2728, Beijing, 100080, P.R. China
{znsun,wangyh,tnt,jlcui}@nlpr.ia.ac.cn

Abstract. As a reliable approach to human identification, iris recognition has received increasing attention in recent years. In the literature of iris recognition, local feature of image details has been verified as an efficient iris signature. But measurements from minutiae are easily affected by noises, which greatly limits the system's accuracy. When the matching score between two intra-class iris images is near the local feature based classifier's (LFC) decision boundary, the poor quality iris images are usually involved in matching. Then a novel iris blob matching algorithm is resorted to make the recognition decision which is more robust than the LFC in the noisy environment. The extensive experimental results demonstrate that the cascading scheme significantly outperforms individual classifier in terms of accuracy and robustness.

1   Introduction

With the increasing demanding of security in our daily life, reliable personal identification through biometrics is currently an active topic in the literature of computer vision. Biometric solutions, such as identification systems using fingerprint, iris, face, palmprint, etc., have many advantages over the traditional authentication techniques based on what you know or what you possess [1, 2]. Among them, iris recognition is tested as the most accurate manner of personal identification [3]. Therefore nowadays many automatic security systems based on iris recognition have been deployed world wide for border control, restricted access, and so on [4].

The iris of human eye is the annular part between the black pupil and the white sclera (Fig.1). There are lots of irregular small blobs, such as freckles, coronas, stripes, furrows and crypts, etc., overlaying on the iris region. Furthermore, the spatial distribution of these blocks in the iris is also random. Such randomly distributed and irregular blocks constitute the most distinguishing characteristics of the iris [22].

Since last decade, a number of researchers have worked on iris recognition with the ambition to improve the application's performance specifications, such as accuracy, processing speed, storage cost and robustness [5-23]. According to the various iris features utilized, these algorithms can be grouped into four main categories: phase-based method [5-9], zero-crossings representation [10-12], texture analysis [13-21], local intensity variation [22,23]. Because the distinctive iris information is essentially embedded in the fine spatial changes in the iris image, local

feature based classifier (LFC) had achieved higher recognition accuracy compared with other methods. But on the other hand, the measurements from minutiae are easily affected by noises, such as occlusions by eyelids and eyelashes, localization error and nonlinear deformations, etc., which greatly limits the system's accuracy. In our experiments [22], about 90% false non-matches are incurred by all kinds of noises. Thus a blob matching algorithm, which attempts to establish the global correspondence between two iris images, is desirable to overcome the limitations of LFC, i.e. sensitive to photometric and geometric distortions.

For the purpose of improving iris recognition accuracy, a cascading strategy that combines the LFC and the global feature based classifier (GFC, because the global topological information is used in blob matcher) is proposed in this paper. The basic idea of this technique is to construct a two stage classification system with reject option. The LFC is implemented first and the GFC is seldom consulted unless the LFC is uncertain of its result. Because the large majority recognition tasks can be handled by LFC, the system's real-time performance is not essentially affected by the added GFC.

The remainder of this paper is organized as follows. Section 2 describes a novel iris blob matching algorithm. The multistage combination architecture will be introduced in Section 3. Section 4 provides the experimental results prior to conclusions in Section 5.

2   Alignment Based Iris Blob Matching Method

A typical iris recognition system includes localization, normalization, feature extraction and matching. Fig. 1 illustrates the preprocessing results [5,14,17].

Based on our observations, the block pattern in iris image is very informative for iris matching. The objective of blob matching algorithm is to find the spatial correspondences between the blocks in the input iris image and that in the stored model and quantitatively assess their similarity level based on the number of matched block pairs. The main steps of the blob matching algorithm are described as follows:

1) Blocks of interest (BOI) segmentation: Because the zero-crossings of wavelet transform often indicate the location of sharp variation points [24], the boundary of BOI can be efficiently detected after dyadic wavelet transform (DWT) performed on the input data. With local processing of edge linking, the closed-boundary regions are labeled. Because the pixels in the region of BOI always have lower intensity than others, the object is labeled as foreground if the result of DWT at its region is negative (Fig. 2b).

| (a) | (b) | (c) |
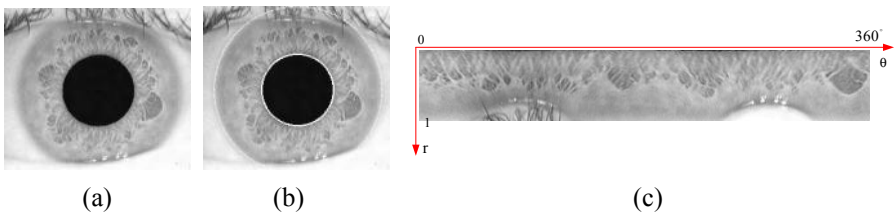
Fig. 1. Preprocessing of iris image; (a) Original image; (b) Result of iris localization; (c) Normalized iris image
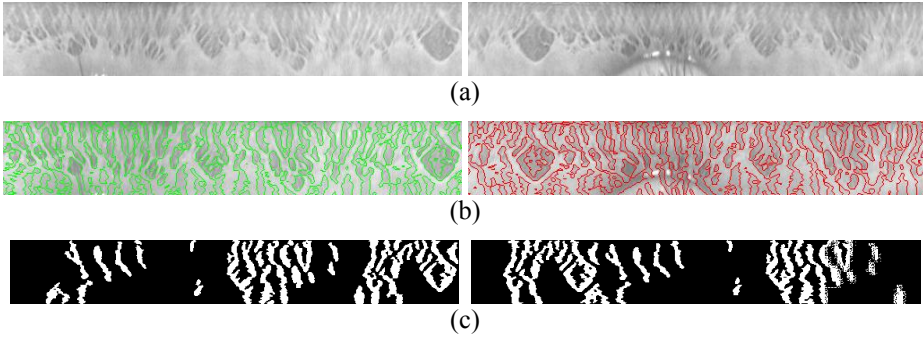
Fig. 2. Alignment based iris blob matching process; (a) Two normalized iris images from same eye; (b) Segmentation results, the contours denote zero-crossings of DWT; (c)The mated BOIs

2) Block pattern representation: In the coordinate system shown in Fig. 1.(c), each BOI's centroid coordinates (R, θ), area (A) and the second order central moments (*MomentR*, *Momentθ*) are recorded as its attributes. Generally, there are nearly one hundred BOIs in an iris image. So each iris image can be represented with a block set $\{(R_i, \theta_i, Area_i, MomentR_i, Moment\theta_i) | i = 1, 2, \cdots, N\}$, where N denotes the total number of BOIs in the image.

3) Alignment of two block patterns: At first all corresponding block pairs (with similar R, A, *MomentR* and *Momentθ*) of the two iris patterns are explored. Each pair of corresponding blocks are supposed as the origins in their own images respectively, so blocks of other pairs should have relative angles ranging from $0°$ to $360°$ with respect to their reference blocks. In each temporary coordinate system pair, number of block pairs which have similar relative θ location is counted. After all iterations, the rotation parameter can be computed from the optimal coordinate system pair with maximum matching count N_m.

4) Similarity assessment: At last, a quantitative matching score of the two iris block patterns is defined as

$$MS = \min(\frac{M_1}{N_1}, \frac{M_2}{N_2}) \tag{1}$$

where M_i and N_i (i=1,2) denote the number of mated blocks (with similar location, area and moments) and all blocks in the i th iris block pattern respectively.

3   Cascading Architecture for Accurate Iris Recognition

In this paper, we aim to reduce the positive identification system's false reject rate (FRR) when the system is operated in the status of low false accept rate (FAR). The

cascading scheme is illustrated in Fig. 3. This is a typical two stage classification system [25]. TL, TH and TB are the predefined thresholds, and TL is defined via

$$P(SL < TL|\text{ Imposter})=0.9999 \tag{2}$$

and TH is defined by

$$P(SL < TH|\text{ Imposter})=0.999999 \tag{3}$$

Thus TL and TH correspond to FAR 0.01% and 0.0001% respectively, which can be learned from the distribution of inter-class comparisons. We found that the majority of false rejections' matching scores fall into the interval between TL and TH. So the second session is introduced to give these false rejected subjects once more chance to provide another set of necessary evidences to verify their identities.

4   Experimental Results

4.1 Database

With the self made iris capture device, an iris database named CASIA has been constructed. This database has been worldwide shared for research purposes [26]. To test the proposed method, we make totally 1,135,566 comparisons between iris images captured in different time, including 3,711 intra-class comparisons and 1,131,855 inter-class comparisons.
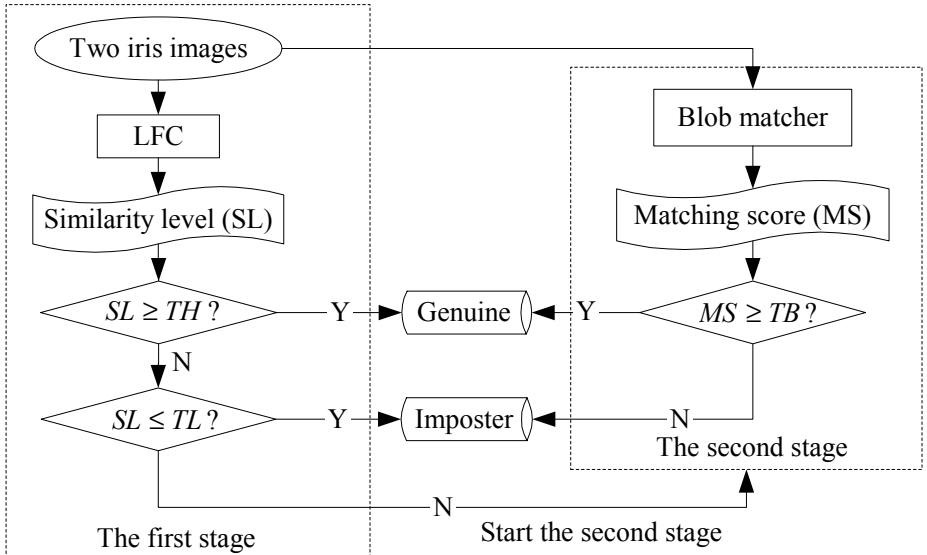
Fig. 3. Cascading scheme for iris recognition
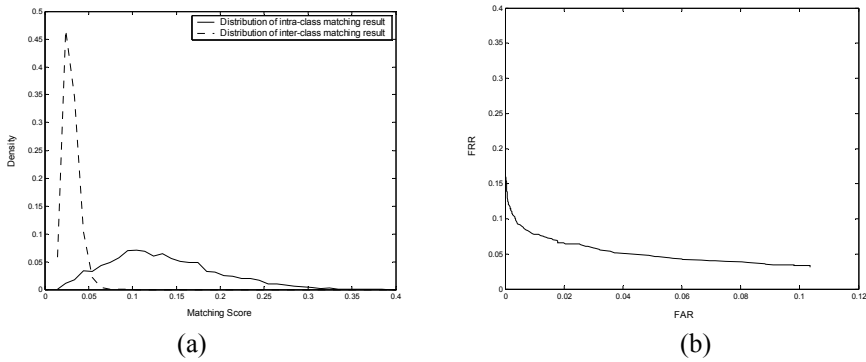
Fig. 4. Performance of the blob matching algorithm; (a) Distribution of intra-class and inter-class matching scores; (b) ROC curve

4.2 Discriminating Power of the Blob Matching Algorithm

The distribution of matching scores of the proposed blob matcher is shown in Fig. 4.(a) and the ROC curve is drawn in Fig. 4.(b). From this experiment, we can see that it is difficult for two random block patterns to achieve high matching score. In this case, more than 98% inter-class matching scores are lower than 0.05, i.e. less than about 5 mated pairs. But accurate segmentation of blobs can not be guaranteed in all cases. So this method is generally worse than most of the existing iris recognition methods in terms of accuracy. However, it will be proved later that the blob matching method is much better than LFC in case of poor quality images because the elastic matching strategy make it accommodate localization error, distortion and occlusions of eyelids or eyelashes. What we need just is the complementary property of this method to the LFC.
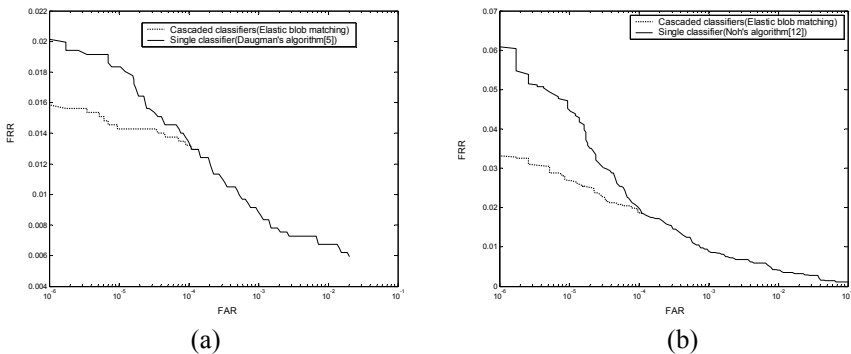
Fig. 5.   Performance of the cascaded classifiers; (a) Results of Daugman's algorithm; (b) Results of Noh's algorithm (only use the local feature)

(a)

(b)
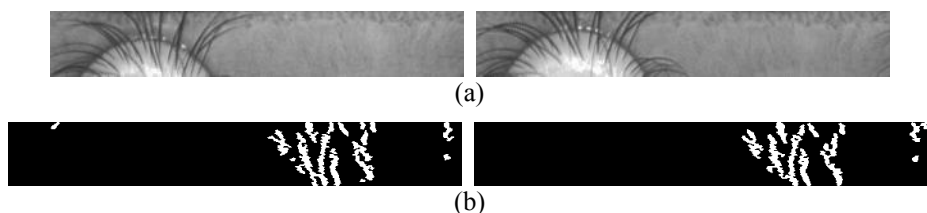
Fig. 6.   An example when GFC is more reasonable than LFC; (a) Two normalized iris images from same eye, the similarity level of Daugman's method is 0.58; (b) The mated BOIs by the blob matching algorithm, the blob matching score is 0.18

4.3 Performance Evaluation of Cascaded Classifiers

Both Daugman's phase demodulation algorithm [5,6] and Noh's wavelet decomposition method [12] are typical LFC. In this subsection they are both cascaded with the blob matching algorithm respectively as shown in Fig.3. The ROC curves of both LFC and the cascaded classifiers are demonstrated in Fig. 5 for comparison. We can see that when $10^{-6}<=FAR<=10^{-4}$ the cascaded system achieves the lower FRR, which demonstrate the GFC is better than LFC in recognizing poor quality iris images. The two curves overlay when $FAR>10^{-4}$. It should be noted then the system's operation state (FAR/FRR) is controlled by the criterion of the GFC.

One example is shown in Fig. 6. which is a false rejection of LFC but correctly accepted by GFC. Because of the occlusions of eyelids and eyelashes, a test of statistical independence [5] of the two images is passed. But the blob matching score is high enough to determine they are from same eye.

From the experimental results, we can see that LFC and GFC are iris image quality-dependent. Although GFC is a weak classifier compared with LFC when the iris images are high quality, which does not affect its expertise in recognize noisy iris images (Fig. 6.). In our case, the poor quality iris images are roughly detected by measuring whether their matching score of LFC is near the classifier's decision boundary. Therefore the system's ROC can be improved if the final decision is then provided by the GFC.

The computational cost of GFC is about two times of Daugman's, but its probability to be used is only about 1% empirically. So for Daugman's system, the average time cost only increases 2%. For Noh's algorithm, this number is about 3%. Compared with the cheap price paid for computation, the benefit in terms of the system's accuracy and robustness is deserved.

5   Conclusions

In this paper, a cascading scheme of multiple representations and matching algorithms for iris recognition is proposed. This method is also applicable in other biometric systems. Another contribution of this paper is a novel iris blob matching algorithm is presented. This method can register two intra-class iris images into sub-pixel precision. Although the experimental results show that blob matching algorithm alone is not promising for iris recognition directly, it has its own specialties for

discriminating noisy iris images, which is complementary with LFC. Our further work is to exploit the mated blocks' area information to give a more accurate result. And another potential application of this blob shape based iris representation is to facilitate iris identification in large scale database by shape indexing technique, which is also an interesting research topic.

Acknowledgements

References

[1]   A.K. Jain, R.M. Bolle and S. Pankanti, (eds.): Biometrics: Personal Identification in Networked Society, Norwell, MA: Kluwer (1999)

[2]   D. Zhang (ed.): Automated Biometrics: Technologies and Systems, Norwell, MA: Kluwer (2000)

[3]   T. Mansfield, G. Kelly, D. Chandler and J. Kane: Biometric Product Testing Final Report, issue 1.0, National Physical Laboratory of UK (2001)

[4]   http://www.iridiantech.com/solutions.php

[5]   J. Daugman: High Confidence Visual Recognition of Persons by a Test of Statistical Independence, IEEE Trans. PAMI, Vol.15, No.11 (1993) 1148-1161

[6]   J. Daugman: Statistical Richness of Visual Phase Information: Update on Recognizing Persons by Iris Patterns, IJCV, Vol. 45(1) (2001) 25-38

[7]   R. Sanchez-Reillo and C. Sanchez-Avila: Iris Recognition With Low Template Size, AVBPA (2001) 324-329

[8]   C. Tisse, L. Martin, L. Torres and M. Robert: Person Identification Technique Using Human Iris Recognition, Proc. of Vision Interface (2002) 294-299

[9]   J. Huang, L. Ma, Y. Wang, T. Tan: Iris Recognition Based On Local Orientation Description, ACCV (2004) 954-959

[10]   W.W. Boles and B. Boashash: A Human Identification Technique Using Images of the Iris and Wavelet Transform, IEEE Trans. on Signal Processing, Vol.46, No.4 (1998) 1185-1188

[11]   C. Sanchez-Avila, R. Sanchez-Reillo, et al.: Iris-based Biometric Recognition using Dyadic Wavelet Transform, IEEE Aerospace and Electronic Systems Magazine (2002.8) 3-6

[12]   S. Noh, K. Bae, J. Kim: A Novel Method to Extract Features for Iris Recognition System, AVBPA (2003) 838-844

[13]   Y. Zhu, T. Tan, Y. Wang: Biometric Personal Identification Based on Iris Patterns, ICPR (2000) Vol.II 805-808

[14]   L. Ma, T. Tan, Y. Wang and D. Zhang: Personal identification based on iris texture analysis, IEEE Trans. PAMI, Vol. 25, No.12 (2003) 1519-1533

[15]   L. Ma, Y. Wang, T. Tan: Iris Recognition Based on Multichannel Gabor Filtering, ACCV (2002) Vol.I 279-283

[16]   L. Ma, Y. Wang, T. Tan: Iris Recognition Using Circular Symmetric Filters, ICPR (2002) Vol.II 414-417

[17]   R.P. Wildes, J.C. Asmuth, et al.: A Machine-vision System for Iris Recognition, Machine Vision and Applications (1996) Vol.9 pp.1-8

[18]   B. Kumar, C. Xie and J. Thornton: Iris Verification Using Correlation Filters, AVBPA, (2003) 697-705

[19]   C. Park, J. Lee, M. Smith and K. Park: Iris-Based Personal Authentication Using a Normalized Directional Energy Feature, AVBPA (2003) 224-232

[20]   S. Lim, K. Lee, et al.: Efficient Iris Recognition through Improvement of Feature Vector and Classifier, ETRI Journal, Vol. 23, No. 2 (2001) 61-70

[21]   K. Bae, S. Noh and J. Kim: Iris Feature Extraction Using Independent Component Analysis, AVBPA (2003) 838-844

[22]   L. Ma, T. Tan, Y. Wang and D. Zhang: Efficient Iris Recognition by Characterizing Key Local Variations, IEEE Trans. Image Processing (Accepted)

[23]   L. Ma, T. Tan, D. Zhang, Y. Wang: Local Intensity Variation Analysis for Iris Recognition, Pattern Recognition (Accepted)

[24]   S. Mallat: Zero-Crossings of a Wavelet Transform, IEEE Trans. Information Theory, Vol.37, No.4 (1992) 1019-1033

[25]   P. Pudil, J. Novovicova, S.Blaha, J. Kittler: Multistage Pattern Recognition with Rejection Option, ICPR (1992) Vol.B 92-95

[26]   CASIA Iris Image Database, http://www.sinobiometrics.com/casiairis.htm

Iris Recognition for Palm-Top Application

Chun-Nam Chun and Ronald Chung

Department of Automation and Computer-Aided Engineering
The Chinese University of Hong Kong, Shatin, Hong Kong, China
`{cnchun,rchung}@acae.cuhk.edu.hk`

Abstract. With palm-top devices increasingly popular, security for both the data contained in the devices and e-commerce transactions through them becomes an issue. Iris texture is a nature-given password that has advantages over alphanumeric password and other biometric features. Existing iris recognition systems however are all designed for either mainframe or desktop applications. This paper describes an iris recognition algorithm that is compact and simple enough to be implemented on palm-top devices. The algorithm exploits a simple transformation -- Radon Transform -- to encode iris images. Experimental results on real image data are shown to illustrate how the proposed system performs in comparison with an existing Gabor wavelet coefficient based algorithm.

1   Introduction

Iris texture is a distinct and stable biometric feature that could be used for authenticity check. Ophthalmologists noted from clinical experience that every iris has a highly detailed and unique texture [4]. Clinical evidence [4] has shown that even twins have different iris patterns. It is also relatively stable across one's life and is rarely subject to the sort of damage that other body parts like fingers could be.

Palm-top computing devices (e.g., PPC, Palm, mobile phones etc.) are increasingly popular for data storage, communication, and e-commerce transaction. Security clearance for the access of data inside the devices or the financial transactions through them is thus an issue. These days palm-top devices often come with not only certain computing power and memory capacity but also built-in digital camera, which are exactly the hardware resources needed for iris recognition. Iris recognition is thus a natural biometric cue to use for authenticity check on palm-top devices.

Iris recognition has been investigated quite extensively; examples of previous work include [2] [4] [5] [7]. However, all such systems are designed for desktop applications. They require computing capacity beyond that of palm-top devices to operate.

In this work, we aim at developing an iris recognition algorithm for palm-top applications, that: (1) requires *simple computations*, so as to let it be amenable to the capacity of palm-top devices; and (2) allows *scalable authenticity level*, so as to allow the required computations of the system to be adjustable according to the needed security level of the application and the available resources of the computing platform.

Radon Transform is a general technique that allows partial information of a signal be captured as projections of the signal to various angles. By varying the number of

angles used, the total amount of partial information collected about the original signal could be increased or decreased. Due to the simplicity of its operation, Radon Transform requires little computations to operate.

In this paper, we propose an iris recognition algorithm that uses Radon Transform to encode iris texture. More specifically, we encode an iris as the projections of its image along a number of directions, with the total number of directions adjustable in accordance with the security level needed. The system also consists of an image pre-processing component that makes the system more user-friendly to lay-users, who could picture their irises not exactly in the head-on fashion.

2   Previous Work

There have been a number of previous works on iris recognition; examples are [2], [4], [5], [7]. Iris recognition system could be combined by four modules: iris image acquisition, iris-ring localization, iris texture encoding, and matching with database.

The general mechanism of image acquisition has been well addressed in [4], [7], although image acquisition for Asians could be tricky as Asian irises tend to have darker background which makes the iris texture not as visible. Our experience and strategies in capturing Asian irises in better contrast are detailed in [3].

Iris localization is about the isolation of the iris-ring. On this, circular edge detector [4] is typically used to detect the boundaries of the iris-ring and isolate it. However, in real world, users are often lay-users and iris might not be always imaged orthogonally. Oblique imaging of iris is thus possible, and as shown in Fig. 1, it could cause eccentricity of the iris-ring. Blind application of the circular edge detector to such image data could make the coding of the same iris be varying across images. In [3], we describe a system that makes use of an elliptical edge detector to detect the boundaries of the iris-ring, and of a coordinate transformation that unfolds the eccentric iris-ring and turns it into a rectangular strip with standardized coordinate axes for matching. Experimental results show the matching performance could be much boosted.

The image encoding and data matching modules are often related, since matching is related to how the input image and the reference data are encoded. A number of coding schemes have been proposed in the literature, ranging from raw intensity data [7], to the coefficients of wavelet transformation [4]. For the raw intensity data, direct comparison schemes like cross-correlation are used for matching. For coding like the coefficients of wavelet transformation, Hamming distance is typically used to describe how similar two codes are. However, both coding scheme require computations which generally exceed the computing capacity of palm-top devices.
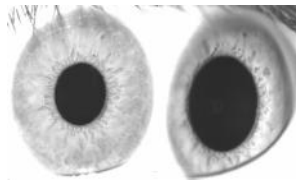
Fig. 1. Eccentric deformation of iris in an image upon lateral viewing. **Left:** Iris imaged from a near-orthogonal angle. **Right:** The same iris imaged from an oblique angle

3   Image Pre-processing

3.1 Isolation of the Region of Interest

Iris appears in image as a ring between the pupil and sclera of the eye. The region of interest is thus the part between the pupil boundary and the sclera boundary. Oblique imaging of iris could result in substantial eccentricity in the ring (Fig. 1). In our work, we allow that to happen by using an ellipse detector to extract the two boundaries. More details about the process are available in [3].

3.2 Iris-ring Unfolding and Normalization

Once the pupil and sclera boundaries are extracted, the iris-ring, which is the part of the image between the two boundaries, has been located. The size and shape of the iris-ring however vary with the imaging distance, imaging angle, and even the illumination. To ease subsequent matching with the reference data, a process that warps each input iris-ring to a form that uses a standard coordinate frame for reference is needed.

We use a coordinate system we call the eccentric-polar coordinate system to unfold the iris ring and normalize it to a rectangular strip with standardized coordinate axes. [3] has the details. An example result is shown in Fig. 2.

3.3 Data Binarization

Unfolded iris-ring can maintain reference to the same region of iris tissue regardless of the imaging condition, but for the Radon encoding, the image has to have its contrast enhanced to make the iris texture standout, and be converted to a binary image.

Fig. 2: Image of an iris-ring under the eccentric-polar coordinates defined by its boundaries. **Left:** the intensity; **Right:** the binarized result

The image contrast is enhanced by stretch their intensities using a linear mapping from $[\alpha, \beta]$ to $[0, 255]$.

In the contrast-enhanced image, the darker pixels correspond to the iris texture, while the brighter ones correspond to the background. We observe from all the image data we have worked with that 100 is a good intensity threshold to binarize the contrast-enhanced image, turning the iris texture pixels to black pixels (regarded as having value 1), and the background pixels to white pixels (regarded as having value 0). The resultant image for the iris ring shown in Fig. 2 is displayed in the same Figure.

4   Radon Transform based Encoding

Radon Transform is a transform that, when applied to an image, computes projections of the image intensities along radial lines oriented at specified angle. Fig. 3 illustrates the operation of the transform. Radon Transform $R_\phi(\theta')$ of $f(\theta, d)$ with respect to the angle ϕ could be described as the line integral of f along each of the θ'-lines:

$$R_\phi(\theta') = \int_{-\infty}^{\infty} f(\theta'\cos\phi - d'\sin\phi, \theta'\sin\phi + d'\cos\phi)\,\mathrm{d}\,d'. \tag{1}$$

where

$$\begin{bmatrix} \theta' \\ d' \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \theta \\ d \end{bmatrix}.$$

In case if f is a binary function and $\theta - d$ a discrete space, $R_\phi(\theta')$ is simply the number of (θ, d) positions on the θ'-line that has $f(\theta, d)$ being 1.

Fig. 4(b) shows the results of Radon Transform with respect to the same angle of 2 irises. In general, while different images of the same iris would have similar Radon Transform profiles, different irises have different iris textures and would generally have different Radon transforms. Radon transform could thus be used to encode the images of iris for verification or recognition purpose.

Yet, Radon transform is only the projection of iris image along a particular angle, and represents only partial information of the iris. The case is analogous to the image projection process: while an image of a 3D scene represents distinct information of the scene, it is only partial information along a particular imaging angle, and it is possible that different scenes have the same image projection along that particular imaging angle. In using visual data to represent a scene, one could use more images of the scene taken from different angles to raise the richness of the image data: the more images are taken, the more distinct would be the collection of image data. We could do the same in our Radon transform based iris encoding scheme.

To extract richer information of iris so as to reduce matching ambiguity, we use Radon transforms with respect to multiple angles of the iris image. Fig. 4 shows Radon transforms of two different irises with respect to two different angles. While their Radon transforms with respect to 0° are similar, their Radon transforms with respect to 135° are substantially different.

An iris could thus be encoded as Radon transforms of its image with respect to a number of angles. The number of angles used could be increased or decreased in accordance with the required authenticity level and the available computing capacity. As to the distribution of the angles, they should be evenly spaced over 180°so as to extract as orthogonal pieces of information about the iris as possible. In all our experiments, we used the angles: 0°, 45°, 90°, and 135° to encode each iris.
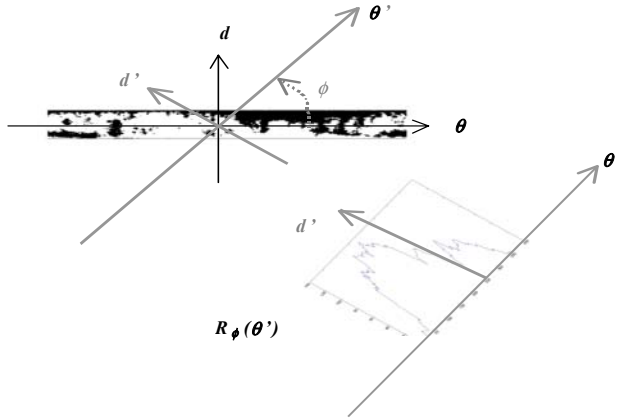
Fig. 3. Radon Transform along angle ϕ, transforming a function $f(\theta, d)$ to a function $R_\phi(\theta')$. For every θ' which represents a line oriented at angle ϕ on the (θ, d) space, the line integral of f along the θ'-line is the value of R_ϕ at θ'

5   Iris Code Matching

On matching the iris codes described above, we use a measure similar to the regional correlation used in [8] but with the dc component of the signals excluded. Given a reference iris code $A_\phi(\theta')$ (of a particular iris) and the iris code $B_\phi(\theta')$ extracted from an input image, both of which are Radon Transforms with respect to the same angle ϕ, we define their similarity measure as:

$$S_\phi = \underset{\tau}{\text{Max}} \left[\frac{\sum_{\theta'} [A_\phi(\theta') - \overline{A}_\phi] \bullet [B_\phi(\theta' - \tau) - \overline{B}_\phi]}{\left[\sum_{\theta'} [A_\phi(\theta') - \overline{A}_\phi]^2 \right]^{1/2} \left[\sum_{\theta} [B_\phi(\theta') - \overline{B}_\phi]^2 \right]^{1/2}} \right] \tag{2}$$

where $\overline{A}_\phi, \overline{B}_\phi$ are respectively the average values of $A_\phi(\theta'), B_\phi(\theta')$ over all θ' values, and τ corresponds to the possible offset between the iris codes along the θ'-axis (due to possibly different rotations of the iris about the camera optical axis). In our implementation, $\tau \in [-5°, 5°]$.

As stated earlier, in our system we use four of ϕ = 0°, 45°, 90°, and 135°, in encoding iris. Thus for each pair of reference iris code and input iris code, we have altogether four similarity measures as defined by Equation (2). We determine the overall similarity measure S of the pair as simply the mean value of the four similarity measures. S has a value between 0 and 1. While a bad match should have S close to 0, a good match should have it close to 1.

(a) Binarized images of two different irises

(b) Radon Transforms with respect to angle 0°

(c) Radon Transforms with respect to angle 135°
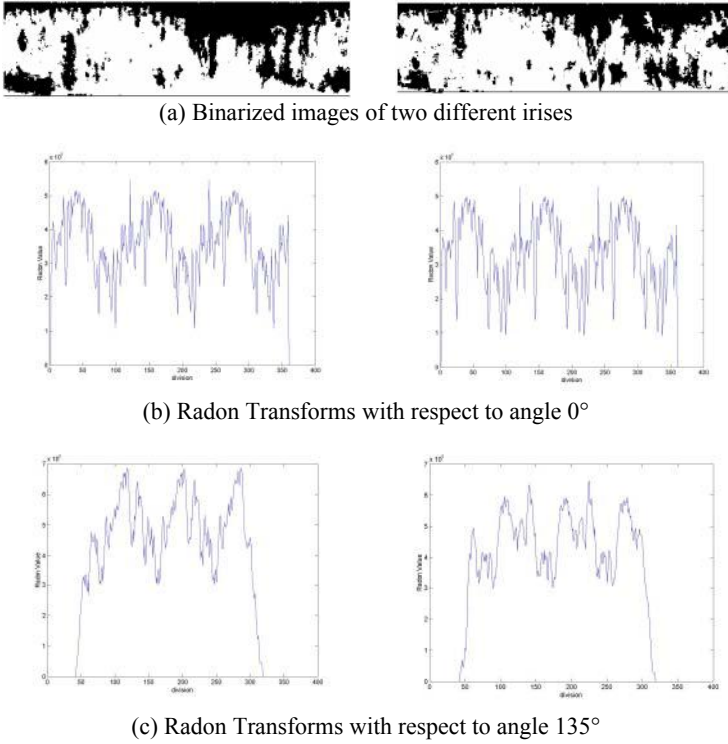
Fig. 4. Radon Transforms of two irises with respect to multiple angles

6   Experimental Results

We have implemented the proposed system and compared its performance on real image data with that of an existing wavelet coefficient-based system [1]. We paid particular attention to the processing speed and storage requirement of the systems.

6.1 The Test Data

We searched for publicly available dataset of iris images for benchmarking purpose but they are not released in the public. We were forced to take image data ourselves for the experiments. Our in-house taken data consist of 36 groups of images, each group comprised of 5 images of the same iris. The data have been made publicly available at the web site: http://www.acae.cuhk.edu.hk/~cvl/.

6.2 Intra-group Similarity vs. Inter-group Similarity

In each image group, we pick one of the images as the reference image of the associated iris, and the rest as the test images. Thus in our dataset there could be four

image-to-reference-image pairings within each of the 36 image groups. We call such pairings the *intra-group pairings*, and the mean of their similarity measures the group's *average intra-group similarity measure*. Each image group *i* also has 35 pairings between its reference image and one test image of every other group. We call such pairings the *inter-group pairings*, and the mean of their similarity measures group *i*'s *average inter-group similarity measure*. A successful iris verification or recognition system should have this property: for all image groups, the intra-group average similarity measure is higher than the inter-group average similarity measure.

Fig. 5 shows the intra-group and inter-group average similarity measures of the 36 image groups. The majority of the image groups show an intra-group average similarity measure of at least 0.98. On the inter-group measures, it could be observed that the similarity measure is generally lower if an image of an iris is matched to the reference image of a different iris, with the majority of the image groups showing an inter-group average similarity measure of at most 0.9.

Based upon the data displayed in Fig. 5a and 5b, the Radon transform-based system could readily identify any input image as either one of the 36 known irises, or an unknown iris. More specifically, an input image could be recognized as iris *i* among the known irises if (1) the matching between the image and the reference image of iris *i* has the highest similarity measure among all known irises in the database, and (2) such a highest similarity measure exceeds the threshold 0.9. If none of the known irises could have, with the input image, a similarity measure that exceeds 0.9, the input image could be regarded as belonging to an unknown iris.

6.3 Performance Comparison with an Existing System

We have also compared the computational performance of the proposed system with that of an existing system [1], whose source codes were publicly available at the web site. The system is a 2-D polar Gabor filter based. The system is a simplified and speeded-up version of the system described in [4].
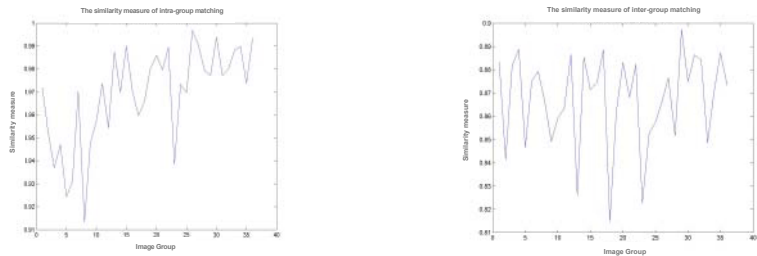
Fig. 5. Intra-group (left) and inter-group (right) similarity measures of 36 image groups

Table 1.   Computational requirement comparison between the Radon Transform-based system and a wavelet coefficient-based system [1]

	Wavelet-coefficient-based System [1]	Radon Transform-based System
CPU Time	31.3860 sec.	4.946 sec.

   Table 1 shows the total CPU times of the two systems in a typical trial of recognizing the iris identity of an input image from the above-described dataset The CPU times were counted all the way from the availability of an input image to the completion of the recognition with respect to the 36 known irises. It could be observed that the computational requirement in the Radon Transform-based system could be as low as 1/6 of that in [1]. We conjecture that the gain in computational speed is largely due to the simplicity of the Radon transform-based approach in comparison with the wavelet coefficient-based approach.

7   Conclusion

The Radon-transform based iris recognition system requires simple computations to operate and is scalable in its recognition performance in accordance with the needed authenticity level and available computing capacity. It is thus more amenable than the existing iris recognition systems to the limited resource-capacity of palm-top devices. Experimental results indicate the proposed algorithm could indeed consume far less computing resources than one of the latest algorithms.

Acknowledgement

References

[1]   W. Berglund and S. Youngblood. Iris Scan: Human Identification Based on a Simple Biometric.   Study Report, Department of Electrical and Computer Engineering, University of Texas at Austin, December 2000.

[2]   W.W. Boles.   A Security System Based on Human Iris Identification Using Wavelet Transform. In Proceedings of First IEEE International Conference on Knowledge-Based Intelligent Electronic Systems, Vol. 2, pp. 533-541, 1997.

[3]   C.N. Chun and R. Chung. Iris Recognition for Iris tiled in Depth. In Proceedings of the 10th International Conference on Computer Analysis of Images and Patterns, Groningen, The Netherlands, August 2003.

[4]   J.G. Daugman. High Confidence Visual Recognition of Persons by a Test of Statistical Independence. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15, No. 11, pp.1148-1161, Nov. 1993.

[5]   P.W. Hallinan. Recognizing human eyes. Geometric Methods Comput. Vision, Vol. 1570, pp. 214-226, 1991.

[6]   C.F. Olson. Constrained Hough Transforms for Curve Detection. Computer Vision and Image Understanding, Vol. 73, No. 3, pp. 329-345, March, 1999.

[7]   R.P. Wildes. Iris Recognition: An Emerging Biometric Technology. Proceedings of the IEEE, Vol. 85, No. 9, pp. 1348-1363, September, 1997.

[8]   N.M. Herbst and H. Morrissey. Signature Verification Method and Apparatus. U.S. Patent 3983535, 1976.

[9]   F. Bouchier, J.S. Ahrens, and G. Wells. Laboratory evaluation of the IriScan prototype biometric identifier.   Technical Report SAND'96-1033, Sandia National Laboratories, Albuquerque, NM, 1996.

Using Multi-matching System
Based on a Simplified Deformable Model
of the Human Iris for Iris Recognition

Xing Ming[1], Tao Xu[2], and Zhengxuan Wang[3]

[1] Department of Computer Science and Technology
Jilin University
5988 Renmin Street, Changchun 130022, China
x.ming@mail.edu.cn
[2] Department of Mechanic Science and Engineering
Jilin University
5988 Renmin Street, Changchun 130022, China
[3] Department of Computer Science and Technology
Jilin University,
10 Qianwei Street, Changchun 130012, China

Abstract. This paper proposes a new method for iris recognition using a multi-matching system based on a simplified deformable model of the human iris. The method defines iris feature points and forms the feature space based on a wavelet transform. In a coarse-fine manner the existence of a simplified deformable iris model is judged and its parameters are determined. By means of such multi-matching system, the task of iris recognition is accomplished. This process can preserve the deformation between two compared iris images and decrease the computation time with improving the recognition precision. The experimental results indicate the validity of this method.

1 Introduction

Iris recognition technology is based on the stable and distinctive iris patterns, and has become a most important biometric solution for personal identification nowadays. Since the 1990s, much work on iris recognition has been done and great progress has been made. Some predominant algorithms for iris recognition have been proposed [1]–[4]. However, there is much room for improvements of iris recognition technology.

In this paper a method is proposed for iris recognition by using multi-matching system based on a simplified deformable model of the human iris, which is simply noted as SDMI. The method uses a locally supported wavelet transform with arbitrarily high regularity, which is simply noted as LSARWT, to process an iris image. According to the definition of iris feature points given by us, the feature space is formed first for the following matching stage. The feature spaces of two within-class irises satisfy a simplified deformable model. For two unknown irises, the coarse matching judges the existence of SDMI and draw

a conclusion. For the existed SDMI, its parameters are estimated by the least square method and the matching is refined. Such matching process decreases the computation time and improve the recognition precision. This strategy is proved to be valid and deserves to be further researched on from a practical viewpoint.

2   The Core Problem

Let I_t be an existing iris template and I_u be an iris image for matching. If I_u and I_t are originated from the same eye, I_u would follow some deformation to a small extent due to the condition variances in an iris acquisition. The key solution is to establish a deformable model between them for a better similarity matching under the relation of homogeneous intrinsic construction. If the mapping function can not exist between I_u and I_t, it suggests that they are different (i.e., the existence of the mapping function between I_u and I_t is the necessary condition of their identification).

3   Iris Feature Points Extraction

3.1   Definition of Iris Feature Points

An iris veins surface presents a kind of radial structure, including various irregular appearances like silk, coronal and crypt forms, etc. Such irregular and stochastic information constitutes the significant features of an iris and distributes along the contours of its trabecular meshwork such as furrows, rings, freckles and so on. From the inherent correlations among coefficients at the same resolution level of the wavelet transform, we know that the interested wavelet coefficients congregate around such zones as edges, etc. Therefore, based on the following Definition 1, we can select those important feature points from the contour points generated by a wavelet transform.

Definition 1. *Iris Feature Points and its Mathematical Representation. The points that are distributed along contours of iris minutiae texture and manifest the property of relatively and uniquely strong edges among its neighbors are called iris feature points.*

For a wavelet transform modulus image noted as I, set a threshold $T = \alpha(\sigma_{2^j} + \mu_{2^j})$, where α is a constant whose initial value is defined by the user. It controls the number of feature points selected for matching; μ_{2^j} and σ_{2^j} are the mean and the standard deviation of I at level 2^j. If the modulus $M_{2^j}p(x,y)$ of a point satisfies $M_{2^j}p(x,y) > T$ and $M_{2^j}p(x,y) = \max_{(x',y') \in N_p}\{M_{2^j}p(x',y')\}$ (N_p is the neighborhood of $p(x,y)$), $p(x,y)$ is taken as an iris feature point.

3.2   The WT-based Feature Space Construction

In image processing, the local maxima of wavelet transform can be used to detect singular points, which provide reliable features for recognition purposes [5]. In
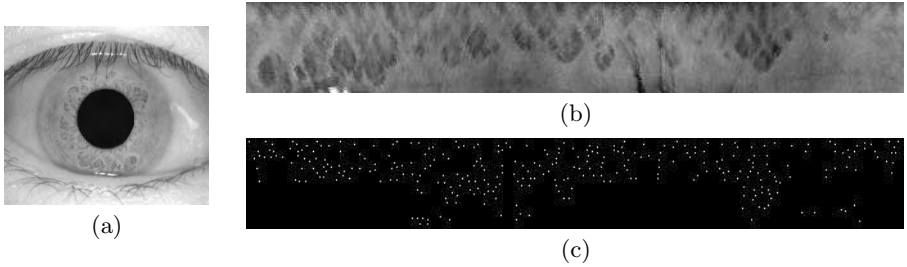
Fig. 1.   Iris preprocessing and feature points extraction: (a) original image; (b) unwrapped image after preprocessing and (c) spatial distribution of WT-based iris feature space (iris feature points are in white)

this paper the wavelet transform LSARWT [6] is used generate the modulus image and the feature space of an iris image is formed based on the following definition.

Definition 2. *Iris Feature Space. For a wavelet transform modulus image noted as I, all the feature points denoted by $p_1, p_2, \cdots, p_M$ form the feature space denoted by a column M-vector P. That is $P = \{p_i | i = 1, 2, \cdots, M\}$.*

Experiments have demonstrated that the number of the elements in the feature space is less than 1/5 number of the contour points. It obviously enhances the efficiency of our method and is favor of iris recognition. Fig. 1(c) illustrates the spatial distribution of WT-based iris feature space.

4   Iris Pattern Multi-matching

4.1   A Simplified Deformable Iris Model

Due to the variance of the iris acquirement, the pupil expansion and contraction will eventually change the distribution of the feature points on the surface of the iris. Generally, from the statistical viewpoint the radial and angular variation of an acquired iris image changes in a small range at different time. Ivins [7] describes a five-parameter deformable model of the iris which we develop into a new simplified deformable iris model. It has the following form,

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \tag{1}$$

It is also denoted by $Z' = RZ + T$. This model is controlled by six parameters: translation in the x and y co-ordinates given by t_x and t_y respectively; a, b, c and d jointly control the small deformation of the normalized iris texture.

4.2   Coarse Matching

Let P_t and P_u denote the feature space of I_t and the one of I_u respectively. To save the computation time, a feature subspace of P_u denoted by Q_u with K elements is formed first based on Definition 3. Then a simple algorithm as shown in 4.2.2 will be used to find the point correspondences between Q_u and P_t. Finally the result of the coarse matching can be easily brought out.

Constructing the Feature Subspace

Definition 3. *Iris Feature Subspace. For a wavelet transform modulus image noted as I and its column M-vector feature space P, I is divided into several sub-regions under a certain rule. Within every designated sub-region, the same number of points are picked out, whose feature values are relatively larger. All the feature points selected will form the subspace of P, which is noted as Q =* $\{q_j | j = 1, 2, \cdots, K\}$.

The valid feature subspace Q should satisfy the following two conditions,

Condition 1: At least three points which can not be aligned on one line should be included in Q.

Condition 2: Three points by which a triangle can be constructed to overlay a larger region should be included in Q.

Coarse Matching Algorithm and Results

Algorithm 1. *Finding the Point Correspondences.*

(1) Let $Q_t = \Phi$ (null set), Q_t is a feature subspace of I_t;

(2) For $i = 1, 2, \cdots, K$, $q_{u,i}$ is an element in Q_u. Δ is a neighborhood of the corresponding position of $q_{u,i}$ in I_t. Compute all the 2-D cross correlations between $q_{u,i}$ and all the feature points in Δ [8]. If the correlation D_c of one feature point p_t satisfies $D_c = \min(\Delta)$ and $D_c < T_c$, p_t is marked as a point correspondence of $q_{u,i}$ and is added to Q_t. Here, T_c is a preset threshold. Otherwise, no matching point to $q_{u,i}$ is supposed to be found in P_t;

(3) Complete constructing Q_t and end the algorithm.

For the preset threshold T_p and if the number of the elements in Q_t is equal to or more than T_p, we accept I_t as a coarse match for I_u and a further confirmation should be carried out. Inversely, we take them as different due to lack of similarity.

4.3   Refined Coarse Matching

We can use the least square method to estimate the parameters of the deformable model. Then the feature space P_t can be mapped into a new one specified by P'_t. Finally the refined matching is performed using a similarity measure based on the distances between best-matched feature pairs (Definition 4).

Least-Square Based Solution of SDMI. Let $q_{u,i}$ be a feature point in Q_u and let $q_{t,i}$ be the corresponding point in Q_t. The representation of N point correspondences between Q_t and Q_u follows as $q_{t,i} \Leftrightarrow q_{u,i}, i = 1, 2, \cdots, N$. By using Equ. 1 an equation in matrix vector format written as

$$q_{u,i} = Rq_{t,i} + T \tag{2}$$

The corresponding least squares solution is as follows

$$\|Y_i\|^2 = \|Rq_{t,i} + T - q_{u,i}\|^2 = min \tag{3}$$

Once we have R and T, P_t can be mapped into P'_t.

Similarity Measure

Definition 4. *Similarity Measure Based on the Distances between Best-matched Feature Pairs is computed by*

$$D = \lambda_0 M_0/M + \lambda_1 M_1/M + \cdots + \lambda_n M_n/M \tag{4}$$

Where M represents the total amount of elements in P'_t and P_u. $M_0, M_1, \cdots, M_n$ stand for the number of point correspondences whose distance value is localized in $[0, 2), [2, 4), \cdots, [2n, 2(n + 1))$ respectively. $\lambda_0, \lambda_1, \cdots, \lambda_n$ are corresponding weights under condition of $0 \leq \lambda_n \leq \cdots \leq \lambda_1 \leq \lambda_0 \leq 1.0$. Thus we can easily get $0 \leq M_0 + M_1 + \cdots + M_n \leq M$ and $0 \leq D \leq 1$.

5   Experiments

To evaluate the performance of our proposed method, we applied it to the CASIA Iris Image Database [9] to do some experiments. Here two application instances are given below.
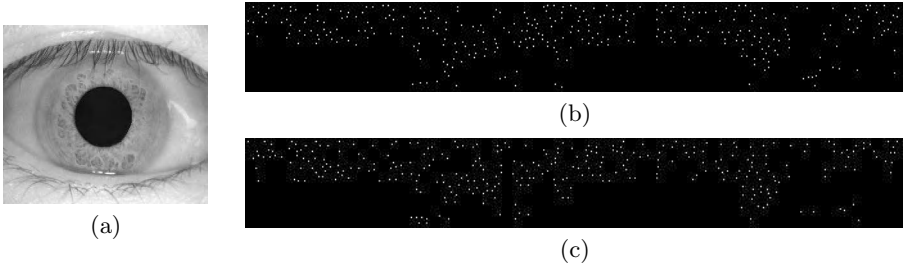
Fig. 2. The input iris image and maps of two feature spaces (white points): (a) the input iris image; (b) map of feature space in I_u and (c) map of feature space in I_t

Table 1. Results and Analysis of the Coarse Matching

	Description	Analysis
Matching among within-class irises	35 in 458 comparisons are rejected to be refined	(1)Too much excessive eyelid occlusion; (2)Inaccurate iris localization.
Matching among between-class irises	254 in 22,333 comparisons need to be refined	Relative with the selection of some thresholds.

5.1   Instance I

An iris image selected from CASIA is taken as I_u, which is shown in Fig. 2(a). Another selected iris image(Fig. 1(a)) is taken as I_t. The maps of their feature spaces are shown in Fig. 2(b) and Fig. 2(c) respectively.

In Algorithm 1, let K be 16 and T_p be 6. In this instance 8 point correspondences were computed out. Using Equ. 3, we obtained the parameters of SDMI, $R = \begin{pmatrix} 1.0047 & -0.0327 \\ 0.0011 & 1.0148 \end{pmatrix}, T = \begin{pmatrix} -0.9016 \\ 0.0415 \end{pmatrix}$. The weights in Definition 4 were initialized by $\lambda_0 = 1.0, \lambda_1 = 0.9, \lambda_2 = 0.6, \lambda_n = 0, (n \geq 3)$. Because the computed similarity between I_t and I_u was 0.58, we accepted that they originated from the same eye.

5.2   Instance II

We utilized 214 iris images from CASIA to do the overall test of our method. Table 1 shows the results of the coarse matching and its analysis. Results of the refined coarse matching are shown in Fig. 3.

5.3   Data Analysis

(1)Instance I suggested that the shift parameters in Equ. 1 can be pre-determined because of the iris normalization; (2)Table 1. suggested that almost 98.9% between-class irises were rejected correctly by the coarse matching and eventually the speed of recognition was sharply increased; (3)Instance II suggested that the degree of match for the within-class irises gathered in the range $[0.4, 0.7]$. For between-class irises, the degree of match gathered in the range $[0.1, 0.3]$. Accordingly 0.35 can be used as the decision criterion for the final matching. The performance evaluation of our method is shown in Table 2.

Table 2. Performance evaluation of our method

Decision criterion	FRR(%)	FAR(%)
0.35	1.24	0.00
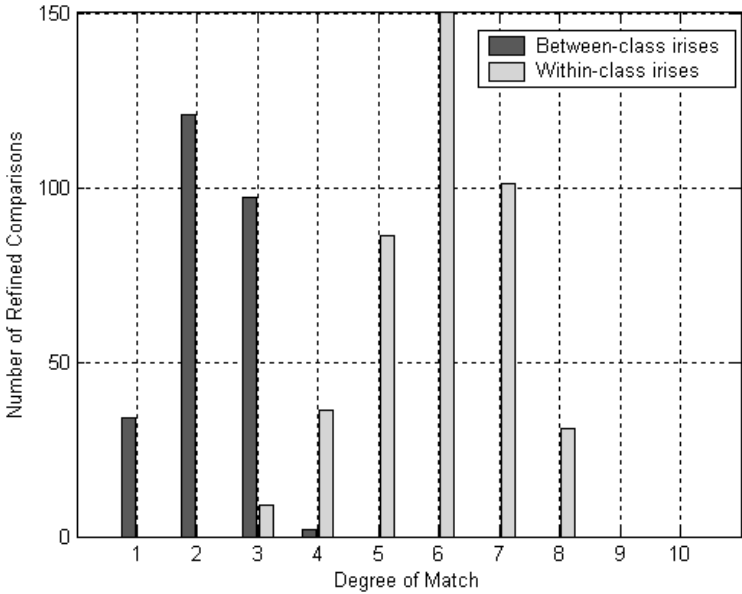
Fig. 3. Results of the refined coarse matching: $1, 2, \cdots, 10$ along x-axis represent distance measure value located in $[0, 0.1], (0.1, 0.2], \cdots, (0.9, 1.0]$ respectively; y-axis indicates the number of refined comparisons

6   Conclusions

A new strategy for iris recognition is proposed in this paper. The wavelet transform of the iris image is the foundation of this strategy. The given definition of iris feature points is reasonable and available. Specially, the judgement and determination of SDMI serve as a multi-matching system. The refined coarse matching achieves the rotation and radial scaling invariance. Instances conducted in this paper indicate the validity of our proposed method.

Acknowledgements

References

[1] Daugman, J.: High Confidence Visual Recognition of Persons by a Test of Statistical Independence. IEEE Trans. on PAMI. **vol.15, no.11** (1993) 1148–1161
[2] Wildes, R. P., Asmuth, J. C.: A Machine Vision System for Iris Recognition. Machine Vision Applicat. **vol.9** (1996) 1–8

[3] Boles, W. W.: A Security System Based on Human Iris Identification Using Wavelet Transform. First International Conference on Knowledge-based Intelligent Electronic System. **vol.2** (1997) 533–541

[4] Zhu, Y., Tan, T., Wang, Y.: Biometric Personal Identification Based on Iris Patterns. Proc. of IAPR, Int. Conf. Pattern Recognition (ICPR'2000). **vol.II** (2000) 805–808

[5] Mallat, S., Hwang, W. L.: Singularity Detection and Processing with Wavelets. IEEE Trans. Information Theory. **vol.38, no.2** (1992) 617–643

[6] Li Cui-Hua, Zheng Nan-Ning: A Locally Supported Wavelet With Infinitely High Regularity and Fast Algorithm for Edge Detection. Chinese J.Computers. **vol.22, no.3** (1999) 269–274

[7] Ivins, J. P., Porrill, J., Frisby, J. P.: A Deformable Model of the Human Iris Driven by Non-linear Least-squares Minimisation. Sixth International Conference on Image Processing and Its Applications (IPA'97; Dublin, Ireland). **vol.1** (1997) 234–238

[8] Zheng, Q., Chellappa, R.: A Computational Vision Approach to Image Registration. IEEE Trans. Image Process. **vol.2, no.3** (1993) 311–326

[9] CASIA Iris Image Database. http://www.sinobiometrics.com

An Iris Recognition Algorithm
Using Local Extreme Points

Jiali Cui, Yunhong Wang, Tieniu Tan, Li Ma, and Zhenan Sun

Center for Biometric Authentication and Testing
National Laboratory of Pattern Recognition, Institute of Automation
Chinese Academy of Sciences
P.O. Box 2728, Beijing, P.R.China, 100080
{jlcui,wangyh,tnt,lma,znsun}@nlpr.ia.ac.cn

Abstract. The performance of an iris recognition algorithm depends greatly on its classification ability as well as speed. In this paper, an iris recognition algorithm using local extreme points is proposed. It first detects the local extreme points along the angular direction as key points. Then, the sample vector along the angular direction is encoded into a binary feature vector according to the surface trend (gradient) characterized by the local extreme points. Finally, the Hamming distance between two iris patterns is calculated to make a decision. Extensive experimental results show the high performance of the proposed method in terms of accuracy and speed.

1 Introduction

To meet the increasing security requirement of the current networked society, personal identification is becoming more and more important. Traditional methods are usually not reliable because they are token-based or knowledge-based. Therefore, a new method for personal identification named biometrics has been attracting more and more attention. As a promising way of authentication, biometrics aims to recognize a person using the physiological and (or) behavioral characteristics such as fingerprints, face, gait, voice, and so on [1]. As a new branch of biometrics, iris recognition shows satisfactory performance due to the rich texture.

Iris recognition has a long history [2]. Great progress has been achieved in automatic iris recognition. for instance, Daugman [3] realized an iris recognition system in 1993 and the identification accuracy is up to 100%. Wildes [4] developed a device to capture iris images from a distance, and a super-resolution method was used to obtain clear images. In our previous work [6-9], we used Gabor filters, multi-channel filters or wavelet for feature extraction and obtained satisfactory performance. Although the existing methods have promising performance, the proposed method in this paper focuses on the improvement of accuracy and speed.

The remainder of the paper is organized as follows. Section 2 introduces the related work. The proposed algorithm is described in detail in Section 3. Section 4 gives the experimental results on the CASIA Iris Database. Section 5 concludes the paper.

(a)   (b)
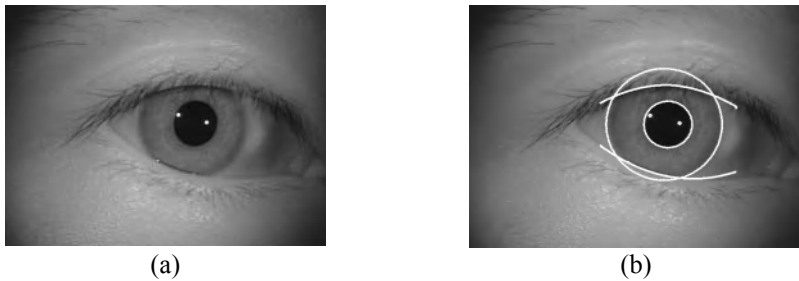
Fig. 1. An example of iris localization results (a) iris image before preprocessing, and (b) iris image after localization

2   Related Work

An iris recognition system includes two important parts: image acquisition (hardware) and recognition algorithm (software). Here, we just introduce the recognition algorithms that include preprocessing, feature extraction and matching.

Driven by the superiority of iris recognition, many researchers have worked on iris recognition.

The principle of Daugman's iris recognition algorithm is the failure of a test of statistical independence on iris phase structure encoded by quadrature wavelets [3]. Wildes *et al*. [4] used Hough transform to localize iris boundaries. Laplacian pyramid was used to capture the discrimination at four different resolutions, and the normalized correlation was used to determine whether the input image and the model image were from the same class. Our earlier attempts to iris recognition are based on texture analysis or shape description [6-9]. In [8], comparison is done between existing methods and the conclusions showed that our method based on texture analysis was just worse than Daugman's method. In [9], we extracted the local sharp variation points of each row of normalized image as the key points of the iris patterns. Although the method has promising performance better than [8], the intra-class distribution can be further improved. Other progress is not introduced due to the space limitation.

3   Local Extreme Point (LEP) Algorithm

3.1 Preprocessing

The preprocessing is very important in an iris recognition system. It includes iris localization, image normalization and, if necessary, image enhancement.

Iris localization (shown in Fig.1) fulfills two tasks: (inner and outer) boundary detection and eyelid detection. It means finding the effective area of the annular iris. The outer and inner boundaries of iris are two non-concentric circles that are localized with Hough Transform following the edge detection. Eyelid detection is done by combing gray values and texture because the eyelid is a weak signal and it is often contaminated by great noises such as eyelashes. Here, we adopt a method based

on texture segmentation to localize eyelids [13]. The method takes a two-step procedure: texture segmentation followed by curve fitting and local searching. The adopted localization method is not detailed here, because it is not the focus of this paper. However, it is very important for the subsequent processing. An example of the localization results is shown in Fig.1.

3.2 Feature Extraction

Here, we propose an easily realized but effective method based on local extreme points.

The trend (gradient) of a surface is almost fully characterized by its local extreme points except for the saddle points and ridges. Given function $z = f(x, y)$, a point (x_0, y_0) is called a local extreme point if it satisfies $f(x_0, y_0) = \max(or\ \min)\{f(x, y)\ |\ (x, y) \in D\}$, where D is a small neighborhood of (x_0, y_0).

However, there are some saddle points in the annular iris from our extensive experiments. The problem does not exist in 1D space. The reasons are that a signal in 1D space varies along only one direction, whereas a signal in 2D space varies in many different directions. Moreover, the significant iris feature varies in the angular direction [8], so we take many slices (sample vectors) with different radiuses along the angular direction to describe the iris patterns with the local extreme points of the serial slices.

In this paper, we employ the local extreme points as key points of the iris patterns. The local extreme points can be detected by Eq.(1)

$$T = I * G',$$
(1)

where I is the normalized iris image and G' serves for a smooth operator and local extreme point detector. It is useful for reducing computational cost to combine the smoothing and differential operators. Also, the local extreme point detection in 1D space along the angular direction can be fulfilled through the selection of G. For example, if we select $G(x) = e^{-\frac{x^2}{2}}$, the differential is only along the angular direction. Therefore, the computational time is much less than that in 2D space. The detection results are illustrated in Fig.2.

After the local extreme points are detected, the iris pattern with the specified radius is encoded in the following way. The intervals between the connected peaks and valleys are set to zero or one. If the signal travels from a valley to a peak, the interval is set to 1; and if the signal goes the opposite way, the interval is set to 0. Finally we obtain a binary vector, denoted by F^0_i, corresponding to the specified slice. Then, the whole feature vector of the iris pattern is represented as follows.

$$F^0 = [F^0_1;\quad F^0_2;\quad \cdots;\quad F^0_k]$$
(2)

where F^0_i denotes the i th feature vector corresponding to the i th slice; and k is the number of sample vectors along the radial direction and can be obtained from experiments. It is obvious that the length of F^0 of different iris patterns is the same.

3.3 Matching

In the matching stage, we calculate the Hamming distance between two feature vectors with Boolean Exclusive-OR operator (XOR) according to Eq.(3)

$$Dist(F^1, \quad F^2) = \frac{1}{Lk} \sum_{i=1}^{k} \sum_{j=1}^{L} XOR(F^1_{i,j} \quad F^2_{i,j})$$ (3)

where L is the total length of the feature vector F^1_i and $F^m_{i,j}$ denotes the j th component of the i th vector of the m th class. Moreover, the area occluded by eyelids is excluded in matching.

The nearest neighborhood (NN) classifier is adopted here.

4   Experimental Results and Analysis

4.1 Image Database

As much as we know, there are currently no common iris databases for algorithm evaluation, so the CASIA Iris Database (version 2) is adopted here. The first version of the CASIA Iris Database is publicly available [12]. The second version of CASIA iris database is more challenging than the first version. It includes more than 1500 iris images from 122 classes, about 95% of the volunteers are from Asia and other 5% are from America, Russia, and so on. The images are captured using the sensor designed by Pattek Corp. and each image has 640*480 pixels. The capturing course is during two sessions and the time interval is 3 months. Because the images are captured from a distance of about 4.5 centimeters, the frequent eyelid occlusion is very serious, even the pupil is occluded partially. Moreover, some iris images are captured under different illuminations and some are captured from users wearing glasses. These factors boost up the challenges in our processing.

4.2 Local Extreme Point Detection

Extensive experiments are done on a PC with P4 2.4GHz processor and 256M DRAM. First, iris is localized using the method in Section 3.1, and then the annular iris is transformed into a polar system to obtain the normalized image.

Fig. 2. The detection results, where "+" denotes some of the valleys

Table 1. The classification power of ten slices (%)

	Slice 1	Slice 2	Slice 3	Slice 4	Slice 5	Slice 6
FAR	38.55	26.83	29.93	34.02	32.90	33.41
FRR	5.74	3.77	3.17	2.08	1.98	1.38
	Slice 7	Slice 8	Slice 9	Slice 10	All 10 slices	
FAR	33.90	33.79	33.72	34.64	<0.0001	
FRR	0.98	2.87	1.59	2.07	1.35	

Each row of the normalized image can be regarded as a slice of the iris surface. We first smooth it and detect the local extreme points, then encode it into a binary vector. The local minimum points of the detection results are shown in Fig.2.

From the above results, we can find that:

(1) There are about 45 key points in each slice and the number is sufficient for describing iris patterns.
(2) Basically, there is most inherent similarity between any two successive slices, so the key points can be down sampled along the radial direction to reduce the length of the feature vector.

Because the iris texture near the pupil is much richer than that near the sclera, the classification ability is different from slice to slice. To evaluate the classification power of each slice, we use each slice of an iris pattern for verification and the results are show in Table 1.

In Table 1, Slice 1 to Slice 10 is sampled from the pupillary boundary to the limbic boundary equally. From Table 1, we can find that:

(1) The low FRR shows that the intra-class clustering is good because the local extreme points are relatively stable;
(2) The high FAR indicates the variability between different classes because of the randomness of the iris texture.
(3) The center of the iris (in radial direction) is of the highest distinguishing power.

From the above discussions, we can find that there is still much redundance in the feature vector, so the code can be further down sampled. To evaluate the performance of the algorithm with different number of slices, further experiments are done and the results are shown in Fig.3.

We can find in Fig.3 that the selection of more than 10 slices does little to the performance of the algorithm. Therefore, ten slices are formed by averaging the successive slices and the results are shown in Table 1.

4.3 Comparison with Other Methods

To evaluate the performance of a system, two indexes are employed to make comparison between different algorithms: decidability index and equal error rate (EER).
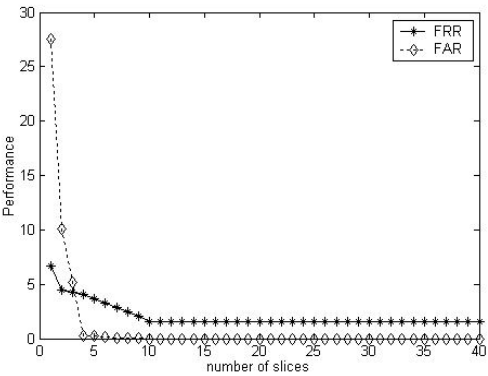
Fig. 3. The performance vs different number of slices

Suppose that u_w and u_b are the mean of the intra-class distance and inter-class distance respectively, and σ_w and σ_b are the standard deviation of the intra-class distance and inter-class distance respectively, the decidability index is estimated by Eq.(4)

$$D = \frac{|u_w - u_b|}{\sqrt{\dfrac{\left(\sigma_w^2 + \sigma_b^2\right)}{2}}} \tag{4}$$

We compare the distribution of distance of our method with some existing iris methods on CASIA database (version 2). As mentioned in Section 2, comparison has been done in our previous work [8], so here we only compare with Daugman's method [3] and the Local Sharp Variation (LSV) method [9].

We re-implement the iris recognition algorithm developed by Daugman and test it on the CASIA Iris Database. The times of intra-class and inter-class verification are 9502 and 1042472, respectively. The distance distribution of intra-class and inter-class verification is plotted in Fig.4 and the curve FRR against FAR is plotted in Fig.5.
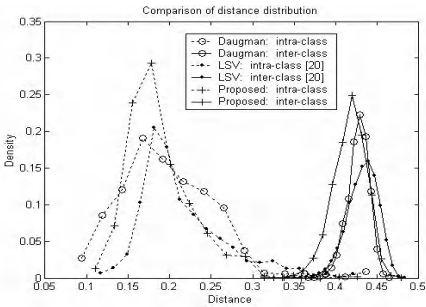
Fig. 4. Distance distribution   **Fig. 5**. ROC curve
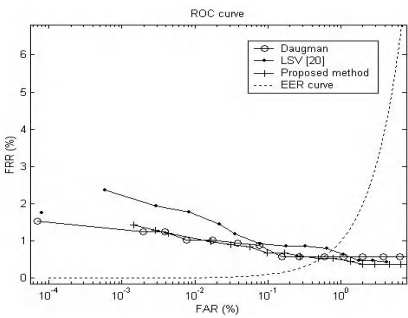
Table 2. The comparison of the performance of the algorithms

Method\Item	u_w	σ_w	u_b	σ_b	D	EER
Daugman	0.1978	0.0580	0.4270	0.0128	5.4593	0.58%
LSV [9]	0.2159	0.0521	0.4323	0.0186	5.5326	0.73%
Proposed method	0.1880	0.0402	0.4162	0.0199	7.1889	0.53%

Table 3. Comparison of the time consumed in feature extraction stage

Method	Daugman	LSV [20]	Proposed Method
Time (second)	0.3426	0.1069	0.0652

The quantified indexes are shown in Table 2, which show that the local extreme point method has satisfactory discrimination power.

In Table 2, "Daugman" denotes Daugman's method we re-implemented. The discrimination index of the proposed method is bigger than that of the others because the mean and standard deviation of the intra-class distance are both smaller, i.e., the clustering of the proposed method is very good. However, The results of Daugman's method are somewhat different from [10] (where u_w =0.0840, σ_w =0.0435, u_b =0.4500, σ_b =0.0380, and D =8.9612) and [11] (where decidability indexes under different conditions are listed). The reasons may come from the difference of the iris database and (or) optimized parameters we selected. We select a challenging iris image database and some tricks of his algorithm may not be realized perfectly.

The proposed algorithm has a large boosting space, because we only employ the "trend" of the iris patterns along the angular direction. The "speed" (magnitude) and the independence between the connected slices, which can characterize the iris patterns more exactly, have been abandoned. Also, we find that the large intra-class distance of our method is incurred by the errors of preprocessing and this is the future work we should focus on.

The time cost of feature extraction is listed in Table 3.

From the data in Table 2 and 3, we can conclude that the proposed algorithm has larger discrimination index and lower EER, so it should have better performance than other algorithms on a challenging database. And it is fast because it combines smooth operator and difference operator in 1D space. As much as we know, it is currently the fastest method.

5   Conclusions

In this paper, an iris recognition algorithm using local extreme points is proposed. The basic idea of the method is that the local extreme points can be used as key points to describe the trend (gradient) of iris patterns. After the iris patterns are encoded into binary feature vectors, Hamming distance is computed to measure the similarity of two iris patterns. Finally, the NN classifier is adopted to make a decision. The superiority of the algorithm is that the local extreme points are relatively stable and the number of the local extreme points is very large for characterizing the uniqueness of iris patterns. The proposed method shows good discrimination in our experiments.

In our future work, we will continue to focus on the global information of iris pattern to compensate for the deformation of iris image incurred by localization errors and different poses. Also, we will continue improving the preprocessing because we find that most false reject occurs when the iris images are localized with errors. Moreover, gradient and relativity of connected slices will be employed to improve the performance of the proposed method.

Acknowledgements

References

[1]   A.K.Jain, R.M.Bolle and S.Pankanti, Eds., Biometrics: Personal Identification in a Networked Society. Norwell, MA: Kluwer,1999.
[2]   L. Flom and A. Safir, "Iris Recognition System", U.S. Patent, No. 4641394, 1987.
[3]   J.Daugman, "High Confidence Visual Recognition of Persons by a Test of Statistical Independence", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 15, No.11, pp.1148-1161, 1993.
[4]   R.Wildes, J.Asmuth, et al., "A Machine-vision System for Iris Recognition", Machine Vision and Applications, Vol.9, pp.1-8, 1996.
[5]   W.W. Boles and B. Boashah, "A Human Identification Technique Using Images of the Iris and Wavelet Transform", IEEE Trans. on Signal Processing, Vol.46, pp.1185-1188, 1998.
[6]   Li Ma, Y.Wang, T.Tan, "Iris Recognition Based on Multichannel Gabor Filters", Proc. of the Fifth Asian Conference on Computer Vision,Vol.I, pp.279-283, 2002.
[7]   Li Ma, Y.Wang, T.Tan, "Iris Recognition Using Circular Symmetric Filters", the Sixteenth International Conference on Pattern Recognition, Vol.II, pp.414-417, 2002.
[8]   Li Ma, Tieniu Tan, Yunhong Wang, Dexin Zhang, "Personal Identification Based on Iris Texture Analysis", IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL. 25, NO. 12, pp.1519-1533, 2003.
[9]   Li Ma, Tieniu Tan, Yunhong Wang, Dexin Zhang, "Efficient Iris Recognition by Characterizing Key Local Variations", accepted by IEEE Trans. on Image Processing.
[10]   John Daugman, "Neural Image Processing Strategies Applied in Real-Time pattern Recognition", Real-Time Imaging 3, pp.157-171, 1997.
[11]   J.Daugman, "Statistical Richness of Visual Phase Information: Update on Recognizing Persons by Iris Patterns", International Journal of Computer Vision, Vol.45(1),pp.25-38, 2001.
[12]   http://www.sinobiometrics.com.
[13]   Jiali Cui, Li Ma, Yunhong Wang, Tieniu Tan, Zhenan Sun, A Fast and Robust Iris Localization Method based on Texture Segmentation, SPIE, Defense and Security Symposium 2004.

Multi-unit Iris Recognition System
by Image Check Algorithm

Jain Jang[1], Kang Ryoung Park[2], Jinho Son[1], and Yillbyung Lee[1]

[1] Division of Computer and Information Engineering, and of BERC
Yonsei University
134, Sinchon-dong Seodaemun-gu, SEOUL 120-749, Korea
{jijang,darkotto,yblee}@cs.yonsei.ac.kr
[2] Division of Media Technology, SangMyung University
7, Hongji-dong, Chongro-gu, SEOUL 110-743, Korea
{parkgr}@smu.ac.kr

Abstract. In this paper, we propose the iris recognition system, which can select the good quality data between left and right eye images of same person. Although iris recognition system has achieved good performance, but it is affected by the quality of input images. So, eye image check algorithm, which can select the good quality image is very important. The proposed system is composed of four steps. At the first step, both eye images are captured at the same time. At the second step, the eye image check algorithm picks out noisy and counterfeit data between both eye images and offer a good qualified image to the next step. At the third step, Daubechies' Wavelet is used as a feature extraction method. Finally, Support Vector Machines(SVM) and Euclidian distance are used as classification methods. Experiment results involve 1694 eye images of 111 different people and the best accuracy rate of 99.1%.

1 Introduction

Iris recognition system is the most promising method among many biometrics because every one has the unique iris pattern and the iris pattern is unchanged as long as one lives. However, at the practical environment, performance of iris recognition system is declined by several reasons. First, some of target population can not use the iris recognition system. Because people who are blind or have severe eye disease does not easy to use the iris recognition system. Second, system performance can be declined in the real environmental conditions due to sensor noise, occlusion, specular reflection or other reasons.

So we need to multi-modal or multi-unit biometrics for improving the system performance and assuring the high confidence. In this paper we propose the iris recognition system based on multiple iris units(left and right iris image). Researches about multiple iris unit are rarely performed except Panasonic(BM-ET300) [1]. It captures both(left and right) eye images at the same time and system accept a user when at least one iris can be identified.

The proposed new method selects better quality iris data between left and right iris images. In this paper, we examine the performance of new method and

also compare the performance to the iris recognition system using the single unit, conjunction and disjunction rules. The paper is organized as follows: In section 2, we prove the theory of proposed method in the view of FAR(False Accept Rate) and FRR(False Reject Rate). In section 3, we explain the iris recognition system which is composed of eye image check algorithm and Daubechies' Wavelet for feature extraction. In addition, the Support Vector Machines(SVM) and Euclidian distance for classification algorithms are described. In section 4, experiment results are reported. Conclusion and future work are drawn in section 5.

2 Performance Improvement of Combined Biometrics

Daugman proved that stronger biometric is better used alone than in combination with a weaker one in the paper [2]. However, many experiments [3] about combined biometrics have showed opposite results of Daugman's analysis. In addition, he did not consider the effect of input data quality. In this section, we explain the theory of proposed method in the view of FAR and FRR.

2.1 Proposed Method Performance Improvement
of Iris Recognition System

In the practical application, system can not always have good quality data due to sensor noise, light condition or other problems. Therefore, Daugman's analysis [2] has to be modified by considering of input data quality. Supposing there are two biometrics, I1 and I2. I1 is iris recognition system only using good quality left iris data and I2 is also iris recognition system only using right iris data of which quality is not uniform. In right iris data, there are good and bad images. The bad quality data means that input iris image has specular reflection severe blurring, counterfeit data and etc. So, we suppose that FA and FR of I1 and I2 are given as below.

$$I1 : P_{I1}(\text{FA}) = P_{I1}(\text{FR}) = P$$

$$I2 : \begin{cases} P_{I2}(FA) = P_{I2}(FR) = P \text{ (in the case of good quality iris data)} \\ P_{I2}(FR) = Q, P_{I2}(FA) = P \text{ (in the case of bad quality iris data)} \\ (P < Q, P < 1, Q =< 1, Q =< tP(t > 1)) \end{cases}$$

When we perform the number of M "Authentic" tests, if X is the number of good data, M-X is the result of bad data. In addition, when we also perform the number of M "Imposter" tests, if Y is the number of good data, M-Y is the number of bad data. In the case of using only single unit in the iris recognition system, total error cases (TEC) of each system can be evaluated by equation (1) and (2).

1) Case of only using I1 (Using only left iris image)

$$TEC = FR \ cases \ + \ FA \ cases = MP + MP = 2MP \qquad (1)$$

2) Case of only using I2 (Using only right iris image)

$$TEC = FR\ cases\ +\ FA\ cases = XP + (M - X)Q + MP$$
$$= (M + X)P + (M - X)Q \qquad (2)$$

When Rule A (disjunctive rule) and Rule B (conjunctive rule) are applied as combination method of I1 and I2 , the total error case of combined biometrics are evaluated by equation (3) and (4).

3) Combination using the Rule A with I1 and I2

$$P_{OR}(FR) = P_{I1}(FR) * P_{I2}(FR) = P * P = P^2 (\text{in case of good data})$$
$$= P_{I1}(FR) * P_{I2}(FR) = PQ (\text{In case of bad data})$$

$$\mathsf{P}_{OR}(FA) = P_{I1}(FA) + P_{I2}(FA) - P_{I1}(FA) * P_{I2}(FA)$$
$$= P + P - P^2 = 2P - P^2$$

$$\therefore TEC = FR\ cases\ +\ FA\ cases = XP^2 + (M - X)PQ + M(2P - P^2) \quad (3)$$

4) Combination using the Rule B with I1 and I2

$$P_{AND}(FA) = P_{I1}(FA) * P_{I2}(FA) = P * P = P^2$$

$$P_{AND}(FR) = P_{I1}(FR) + P_{I2}(FR) - P_{I1}(FR) * P_{I2}(FR)$$
$$= 2P - P^2 (\text{In case of good data})$$
$$= P_{I1}(FR) + P_{I2}(FR) - P_{I1}(FR) * P_{I2}(FR)$$
$$= P + Q - PQ (\text{In case of bad data})$$

$$\therefore TEC\text{=}FR\ cases\ +\ FA\ cases = X(2P - P^2) + (M - X)(P + Q - PQ) + MP^2 \quad (4)$$

When the proposed method is used for the selecting the better iris data between two iris images at the preprocessing stage, error rate is evaluated by equation (5).

5) Proposed Method

- In case of good data (selecting left and right iris)

$$TEC = FR\ cases\ +\ FA\ cases = XP + YP = P(X + Y)$$

- In case of bad data (selecting left iris)

$$TEC = FR\ cases\ +\ FA\ cases = (M - X)P + (M - Y)P = P(2M - X - Y)$$

$$\therefore Overall\ TEC\ of\ 5) = P(X + Y) + P(2M - X - Y) = 2MP \qquad (5)$$

Here, the TEC of case 1) and proposed method are same and TEC of case 1) is smaller than from case 2) to 4). That means if there is no bad quality data in iris images, the performance of our method is same to that of case 1), but such case is not feasible in actual recognition system. Consequently, if we can select only good quality data in the identification or verification process, proposed method has best performance.

3   Iris Recognition System

In the iris recognition system, we capture left and right eye images at the same time at the first step. At the second step, for selecting the good quality iris image, the eye image check algorithm picks out the bad quality image from both iris images. After selecting the good iris image, we use the Daubechies' wavelet to decompose and to analysis the iris image. Finally, as a matching algorithm, we use the SVM and Euclidian distance. After comparing the performance between two measures, we select the algorithm that has good result.

3.1   Selection of Good Iris Data

In the image acquisition step, left and right eye images of each user are rapidly captured by the CCD camera, 30 frames per second and stored by $320 * 240$ size. After that, to detect the counterfeit iris and to evaluate the quality of eye image, we use an efficient algorithm of eye image check, which evaluates the coefficient of variation of pupil radius with 2D bisection-based Hough transform and after that, tests the eyelid movement with a region-based template deformation and masking method [4]. If the eye images pass these two tests, to detect the counterfeit iris data, 2D-FFT spectrum of eye image is analyzed. At the second step, the algorithm which checks eye image quality to find out bad quality images such as occlusion, eyelash interference and the truncation of iris region. After finishing the algorithm of eye image check, we can select the qualified image between both iris images.

3.2   Feature Extraction

After finishing the algorithm of eye image check, we perform the interpolation and normalization of iris image for feature extraction. In the feature extraction step, all the iris images are decomposed into sub bands using Daubechis' wavelet transform [5]. Daubechis' wavelet has good localization property and power of high texture classification. Hence we use four features i.e. mean, variance, standard deviation, and energy from the grey-level histogram of sub bands[5]. In this paper, each iris image was decomposed into three level using Daubechies' tap-4 filter which resulted in 12 sub bands from which to extract iris features. Feature vectors are represented by statistical features, thus four statistical features were computed from each sub band. In addition, we divide the sub images into local windows in order to get robust feature sets against shift, translation and noisy environment. Statistical features are extracted from local windows on the corresponding sub bands, the sub bands of the intermediate levels, to represent feature vectors.

3.3   Pattern Matching

We used two pattern matching methods, Support Vector Machine(SVM) [6] and the Euclidian distance. SVM is a similarity measure and performs pattern recog-

Table 1. Data Set

	Naked eye	Glasses	Contact Lens	Total
Real Data	1,363	160	111	1,634
Fake Data	40	.	20	60
Total	1,403	160	131	1,694

nition for two-class problem by determining the optimal linear decision hyper-plane. We use a RBF kernel because it shows the best discriminant performance.

We also use Euclidian distance measure for evaluating the distance between input vector and enrolled feature vectors. To evaluate the distance, we compute the representation vectors of each class. Normalized feature vectors which are extracted in the feature extraction step, are grouped by each class and we calculate the average feature vectors in the same class. The criterion of choosing the class is the shortest distance.

4   Experiment Result

As the experiment data, we captured 1,694 eye images of 111 peoples with Senex Camera [7]. Table 1 is the data set. As counterfeit data, forty data of naked eye images were printed on the paper with the resolution of 1200 dpi and twenty contact lens data with fake eye pattern printed were used. Experiments were performed on the PC with Pentium IV 2.4 GHz CPU. The average time of identification process spent 0.134 seconds and average time identification process spent 0.096 seconds.

We performed four types of experiments which are SVM based and Euclidian based verification and identification. In each SVM based and Euclidian distance based experiment, another three experiments were performed by using the single(left and right) iris and proposed method. About the SVM based identification system, results of each experiment are shown in Table 2 and ROC curve of experiment is shown in Figure 1(a). We also measured the verification system performance on the 1,694 pattern set (1,694 authentic tests and 155,771 imposter tests). Results of each experiment are shown in Table 2 and ROC curve of experiment is shown Figure 1(b). The threshold of RBF kernel is -0.08437 and at this point we observed the 11 False reject and 682 False accept.

Experiment result of Euclidian distance based identification is shown in Table 3, Figure 2(a) and result of verification is shown in Table 3, Figure 2(b). Results of SVM and Euclidian distance based system are shown the improved evaluation performance when it use proposed method.

Through experiments, we can conclude that proposed system is very effective and improve the accuracy of the system performance. In addition, performance of SVM based system is better than Euclidian distance based system. It is due to the different characteristic between SVM and Euclidian distance. SVM can reflect the similarity of features in the same class through the training process

Table 2. False Rejection rate on the test with different values of false acceptance rates in the SVM based system

FAR	FRR(%)									
	Identification					Verification				
	Left	Right	AND	OR	Proposed	Left	Right	AND	OR	Proposed
0.1	3.2	3.4	3.50	3.09	2.3	1.27	1.29	1.30	1.25	0.79
0.4	2.1	2.54	2.59	2.04	1.6	1.02	1.12	1.13	1.00	0.64
0.6	1.8	1.96	1.99	1.76	1.2	0.95	0.99	0.99	0.94	0.53
0.8	1.32	1.54	1.50	1.29	0.9	0.74	0.82	0.82	0.73	0.48
1	1.03	1.21	1.22	1.017	0.55	0.59	0.61	0.61	0.58	0.43

but Euclidian distance can not reflect the similarity. We select the result of SVM and accuracy rate of identification is 99.1%.

5   Conclusion and Future Work

As experiments of the section 5, the iris recognition system using multi unit(left and right iris) is achieved considerable improvement of system performance by selecting the better quality iris image between left and right images. In addition, we theoretically and empirically prove that proposed method is better than the method using AND, OR rule and single iris. In future experiment, we will perform the experiment at feature extraction level or matching score level. In addition, we will develop the more efficient method of preprocessing step. For the selection of good quality iris image, we will also have to check the specula reflection in the image. In the feature extraction step, we will compare the performance of Gabor filter vs. Daubechies' Wavelet vs. ICA. About the
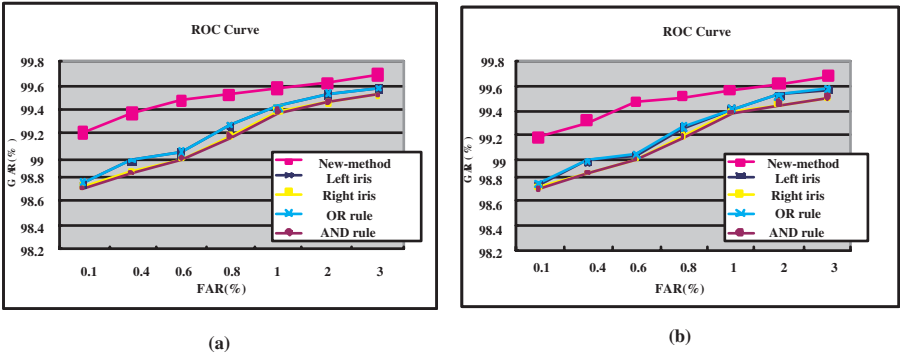
Fig. 1. SVM based (a) Identification system (b) Verification system using proposed method vs. single iris vs. AND , OR rule

Table 3. False Rejection rate on the test with different values of false acceptance rates in the Euclidian distance based system

FAR	FRR(%)									
	Identification					Verification				
	Left	Right	AND	OR	Proposed	Left	Right	AND	OR	Proposed
0.1	4.12	4.33	4.50	3.94	3.4	1.84	1.93	1.96	1.80	1.67
0.4	3.83	3.92	4.07	3.67	2.89	1.56	1.66	1.68	1.53	1.29
0.6	3.26	3.54	3.65	3.14	2.54	1.23	1.29	1.30	1.21	0.96
0.8	2.87	2.96	3.04	2.78	1.98	0.89	0.97	0.97	0.88	0.73
1	2.43	2.56	2.62	2.36	1.76	0.69	0.82	0.82	0.66	0.54

(a)   (b)
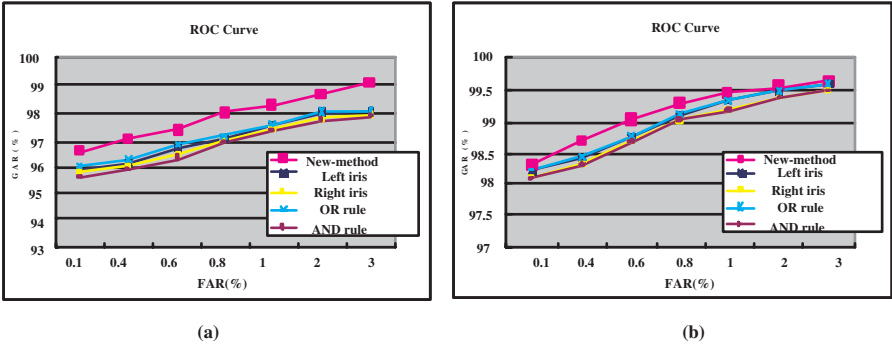
Fig. 2. Euclidian based (a)Identification system (b)Verification system using proposed method vs. single iris vs. AND , OR rule

pattern matching method, we will compare performances among several kernel functions(e.g. liner, polynomial) of SVM.

Acknowledgement

References

[1] ftp://ftp.panasonic.com/pub/Panasonic/CCTV/SpecSheets/BM-ET300.pdf. 450
[2] J. Daugman., Biometric decision landscapes, Technical Report No. TR482, University of Cambridge Computer Laboratory, (2000). 451

[3] A.K. Jain and A. Ross., Multibiometric Systems, To appear in the Communi-
    cations of the ACM, Special Issue on Multimodal Interfaces , January (2004).
    451

[4] J. Deng and F. Lai., Region-Based Template Deformation and Masking for Eye-
    Feature Extraction and Description. Pattern Recognition, 30(3): 403-419, Mar.
    (1997). 453

[5] G. Kee., Iris Recognition System Using Wavelet Packet and Support Vector Ma-
    chines. Ph. D thesis, Yoinsei University, (2003). 453

[6] V. Vapnik, Statistical learning theory, John Wiley & Sons, NewYork, (1998). 453

[7] http://www.senextech.com/. 454

Iris Based Human Verification Algorithms

Bhola Ram Meena, Mayank Vatsa, Richa Singh, and Phalguni Gupta

Department of Computer Science and Engineering
Indian Institute of Technology Kanpur
Kanpur – 208016, INDIA

`{brmeena,mayankv,richas,pg}@cse.iitk.ac.in`

Abstract. In this paper three algorithms for iris verification have been presented. Iris detection algorithms include the normalization and iris extraction steps. Three algorithms for verification process are (a) Algorithm using radial and circular features, (b) Algorithm using Fourier transforms and (c) Algorithm using Circular–Mellin transforms. Proposed algorithms have been tested on CASIA database and some non–infrared Iris images. The experimental results show that the algorithm based on Circular – Mellin Transform gives the best result with an accuracy of 95.45%. Some initial experiments on non–infrared iris images shows that this algorithm can work on such images but it still requires some more attention and this is our future work.

1   Introduction

Most of the biometrics authentication systems are based on signal and image processing based algorithms. Among the various traits, iris recognition has attracted a lot of attention because it has various advantageous factors like greater speed, simplicity and accuracy compared to other biometric traits. Iris recognition relies on the unique patterns of the human iris to identify or verify the identity of an individual.

For iris recognition, an input video stream is taken using Infra-red sensitive CCD camera and the frame grabber. From this video stream eye is localized using various image processing algorithms. Area of interest i.e. iris is then detected from the eye and the features are extracted. These features are encoded into pattern which is stored in the database for enrollment and are matched with the database for authentication.

This paper presents three new algorithms for iris recognition. The first algorithm is based on extracting circular and radial features using edge detection while in the second algorithm features are extracted using Fourier transforms along radial direction. The third algorithm is based on feature extraction with Circular-Mellin operators. These operators are found to be scale invariant and rotational invariant and convolution with them results in shift invariant features.

In Section 2 various steps of the proposed iris verification algorithms are presented. Experimental results of the proposed algorithms are given in Section 3. The experiments have been carried out on the CASIA image database [4] provided by Prof. T. Tan and some iris images taken in non – infrared lighting environment by a CCD Camera. Last section is the conclusion and future work.

2   Iris Verification Algorithm

This section describes various steps of our iris verification algorithm which consists of two sub-algorithms: Iris Detection and Matching Iris Patterns. Iris Detection involves detection of pupil boundary and outer boundary of iris. The other algorithm extracts the feature vector and performs matching.

2.1 Iris Detection

The first step in iris detection (also known as preprocessing) is to detect the pupil. The center of pupil can be used to detect the outer radius of iris patterns. Usually the image captured for iris has many undesired parts like eyelids, pupil etc. This step also involves minimizing noise in iris and making iris scale invariant, i.e. iris patterns independent on size of input image. This can be achieved by converting iris in polar coordinates and normalizing it radially.

Detection of Inner Boundary. Inner boundary of iris is detected by detecting pupil. First the gray scale input image is changed to binary format by using a suitable tight threshold. Assuming that circular area of pupil is the largest black circular part, pupil is detected by searching for largest black circular part in binary image. Points are evenly initialized on image for searching largest black circular area around it and after several iterations we can get the largest circular black area and the boundary of this circular area is the inner boundary (Fig. 1)

Detection of Outer Boundary. The outer boundaries of iris are detected with the help of center of pupil. The binary mage is taken and concentric circles of different radii are drawn with respect to center of pupil. For a particular circle the change in intensity between normal pointing toward center and away from center is measured. The radius having highest change in intensity is considered as outer boundary. Fig. 1 shows the change in intensity along normal for a circle.

Iris in Polar Coordinates. As iris patterns lies between two circle concentric at center of pupil. By changing image coordinates to polar coordinates we can remove useless pupillary part. In polar coordinates iris can be seen as rectangular plate. Let *(x, y)* be any point on input image with respect to center of pupil, which lies between inner and outer boundaries of iris. Let $f(x, y)$ be pixel value of point *(x, y)* then in polar

Input image   Inner boundary   Change in intensity   Inner and outer
                                                                       boundaries

Fig. 1. Iris Detection

Iris with inner and | Iris in polar | Mask to avoid eyelids | Normalized polar iris
outer boundary | coordinate | and hairs
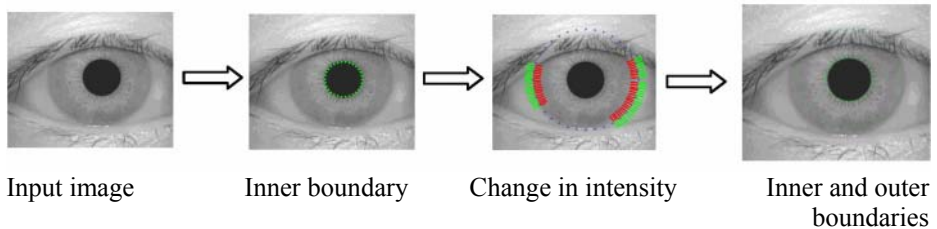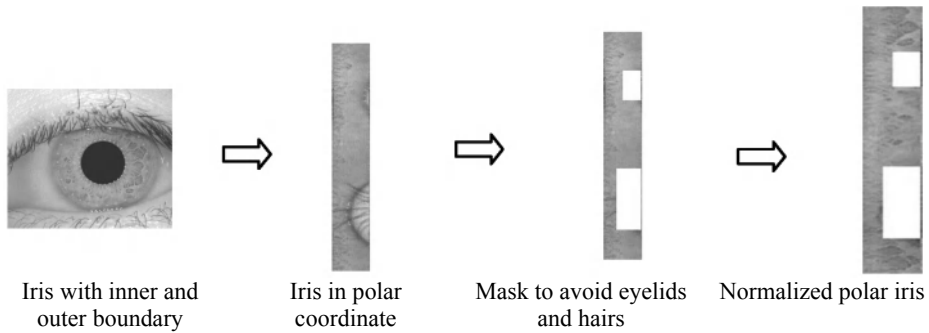
Fig. 2. Getting iris in polar coordinates and reconstruction of iris image

coordinates it will map to the points *(r , θ)* where

$$r = \sqrt{x^2 + y^2} \quad \text{and} \quad \theta = \tan^{-1}(y/x) \quad \text{for} \quad \theta \in (-\pi, \pi] \tag{1}$$

Figure 2 shows an example of the iris in polar coordinates. There are some errors in the iris image due to presence of eyelids. We can get rid of these eyelids and hairs by using mask on the polar image. We used rectangular masks to remove noise due to eye lids. Assuming iris boundary as circle we found that upper eye lids come in 60° - 120° part of upper iris region and lower eye lids come in 255° - 285° of lower iris region. We have consider the 75% gap between pupil and iris boundary as noise in upper iris region and 50% gap in lower region as noise.

Iris Normalization. Iris normalization is done to make iris image in polar coordinates independent of size of the input image. Polar iris image is normalized by fixing number of points along radius. The constant number of points along radius of iris is known as normalization constant (N). For normalization all *r* calculated in Eq. (1) are mapped between radius 0 to N.

$$r_{norm} = (r - r_{min})\frac{N}{r_{max} - r_{min}} \quad \text{where} \quad r_{max} = Maximum(r) \quad \text{and} \quad r_{min} = Minimum(r) \tag{2}$$

It may happen that more than one point on polar image map to same value of r_{norm} and θ then pixel value of the point $[r_{norm}, \theta]$ is assigned by the minimum pixel value of these points. On the other hand it may happen that for a particular r_{norm} and θ no point in polar image maps to it then pixel value of this point is taken as minimum pixel value of its neighbor normalized polar points. Fig. 2 shows the example of iris normalization.

2.2 Algorithms for Verification

Each of the proposed algorithms for verification consists of two steps: (a) Iris code generation and (b) Matching Iris Codes. First step is used to extract features from the normalized iris image given in polar coordinates and to generate an Iris code. Second step deals with matching of the Iris code generated for query image with the other available in database.

Three algorithms have been presented and analyzed. The first algorithm is based on extracting circular and radial features using edge detection while in the second one, features are extracted using Fourier transforms along radial direction. Feature extraction in the third algorithm is done using Circular-Mellin transform.

Verification Based on Circular and Radial Features. This algorithm is based on edge detection. Edges are detected in input image using canny edge detector [1]. After edge detection image is changed to binary format in which white pixels are present on edges and black pixels elsewhere. The number of white pixels in radial direction and on circle of different radius gives important information about iris patterns. Normalized polar iris image will contain only white and black pixels as it is obtained from above edge detected input image.

Extracting Radial Features. In iris image value of feature at particular angle is the number of white pixels along the radial direction. If we consider

$$S_{r,\theta} = \begin{cases} 1 & iris\_polar\_image[r][\theta] == WHITE \\ 0 & iris\_polar\_image[r][\theta] == BLACK \end{cases} \tag{3}$$

then feature at angle θ will be

$$F_\theta = \sum_{r=1}^{N} S_{r,\theta} \tag{4}$$

Extracting circular features. In iris image, value of feature for particular radius is considered as sum of white pixels along the circle of that radius and keeping the meaning of $S_{r,\theta}$ same as in Eq. (3). The feature of particular radius r will be given as $F_r = \sum_{\theta=0}^{360} S_{r,\theta}$. Figure 3 shows the graphical representation of radial and circular features of the iris. Iris code is generated as vector of radial and circular features.

Matching. Each Iris code can be considered as signal and then the matching of two Iris codes become the problem of comparing two signals. Absolute difference between two Iris codes is used to compare two Iris codes. Let us define

$$\Delta F_\theta = Minimum \quad \{\sum_{\theta=0}^{360} \left|F_{\theta+\phi}^1 - F_\theta^2\right|\} \quad \text{for} \quad -10 \le \phi \le 10 \tag{5}$$

$$\Delta F_r = \sum_{r=1}^{N} \left|F_r^1 - F_r^2\right| \tag{6}$$

Equation (5) differs from Eq. (6) because radial features can be rotated if input image is rotated. Assuming that there is no rotational error of more than $10°$, then the final difference is taken as minimum of all differences considering the rotational error.

If $\alpha\Delta F_\theta + \beta\Delta F_r \le Threshold$ then both iris codes are of same person where α and β are weighted parameters. The reason for using different weight for radial and circular features is that circular features are scale and rotational invariant while radial features are not rotational invariant.

(a)

(b)
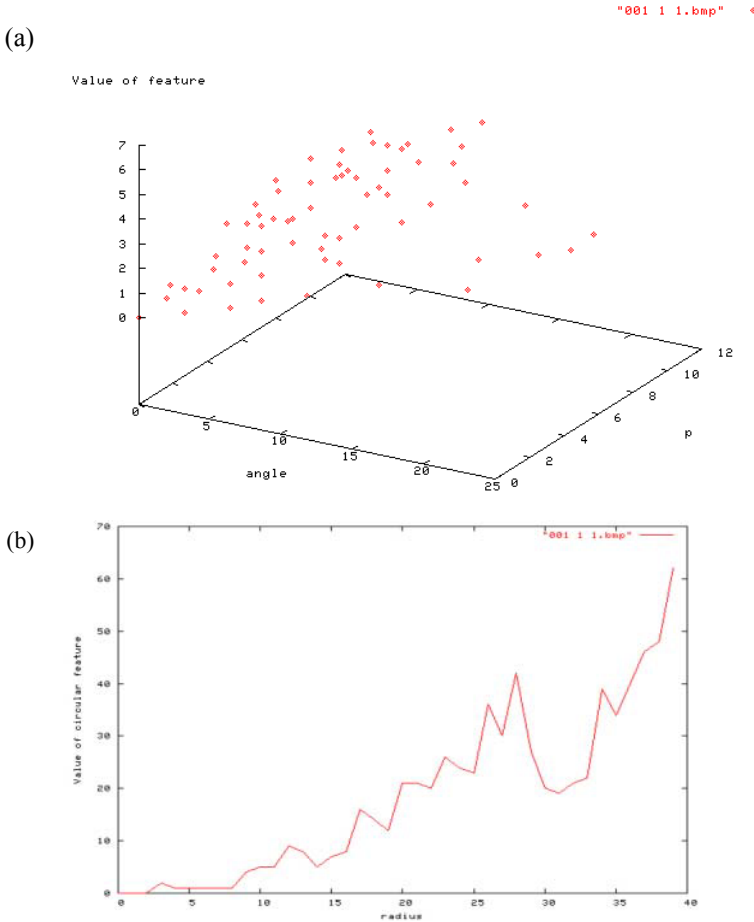
Fig. 3. (a) radial features of iris, (b) circular features of iris

Verification Based on Fourier Transforms Along Radial Directions. This algorithm is also based on edge detection. At first, edges are detected in input image which changes it in binary format. One minor change is done in normalized polar iris image that white pixel has the value 1 instead of 255. The reason for using Fourier transforms on radial direction instead on angular direction is that rotational error can be coped at matching time but not the radial error introduce due to extraction of iris from eye image. Fourier coefficients extract the features along radial direction. Radial features are more important because most of the iris images are noise free.

Generating Two Dimensional Iris Code. Let $I(r, \theta)$ be modified normalized polar iris image. Fourier transforms are applied in radial directions for getting features using Eq. (7)

$$Z = \int_r I(r,\theta)e^{-jpr}dr \tag{7}$$
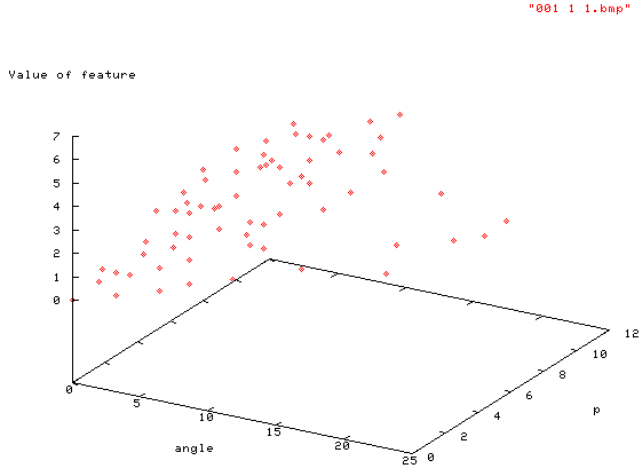
Fig. 4. 2 D Feature Vector
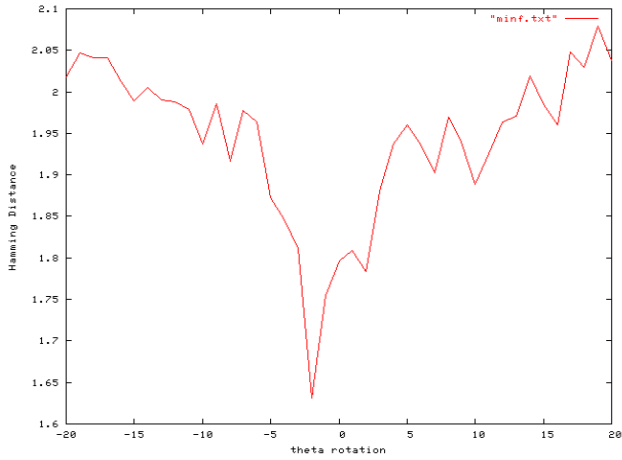
Fig. 5. Hamming distance for different rotational error

The value of Iris code for particular θ_0 and p will be integral part of $|Z|$ .For $0 \le \theta_0 \le 360$ and $1 \le p \le P$ we can create a two dimensional feature vector as shown in Fig. 4.

Matching. Hamming distance between two iris codes is used to determine whether they belong to same person or not. Let A and B be two iris codes to be matched. Hamming distance [2] is calculated as follows:

$$HD = Min\{\sum_{i=0}^{M} \sum_{j=0}^{N} (A(i+\phi, j) - B(i, j))\} \quad \text{for} \quad -10 \le \phi \le 10 \quad (8)$$

where M is size of angular part generally 360 and N is value of P as in Eq. (7) taken at time of creating 2-dimensional iris code. If $HD \leq threshold$ then the given two iris belong to the same person. Fig. 5. shows the graph for selecting the threshold value for hamming distance for different rotational error.

Verification Using Circular-Mellin Operators. The Circular Mellin operator [3] is given by

$$h(\lambda,\theta) = e^{-\lambda}e^{j(2\pi P\lambda + Q\theta)} \quad j = \sqrt{-1} \tag{9}$$

where constants P and Q controls the weights given to radial and circular features of image. Circular - Mellin Operators are scale and rotational invariant. Convolution of iris image with it makes it shift-invariant. There is no use of scale invariance of Mellin operator as polar iris images are normalized to be scale invariant. Features in iris image are extracted based on phase of its convolution with Mellin operators. The polar image of iris given by $P(r, \theta)$ is convolved with Circular - Mellin operator.

$$Z = \int_r \int_\theta P(r,\theta) \frac{r_0}{r^2} e^{j(2\pi p \log(r/r_0) + q(\theta - \theta_0))} d\theta dr \tag{10}$$

where $Z = Re(Z) + j\, Im(Z)$ and p, q are parameters controlling behavior of filter. Iris code is generated based on phase of above Z. Iris code at point (r_0, θ_0) is taken as

$$iriscode[r_0,\theta_0] = \begin{cases} 1 & Re(Z)*Im(Z) \geq 0 \\ 0 & Re(Z)*Im(Z) < 0 \end{cases} \tag{11}$$

Fig. 6. shows the iris code generation from polar image.

Matching. The noise in iris also introduces the noise in iris code. A new method for comparing two iris patterns has been presented. As iris code generated by our algorithm is two dimensional array of bits. Instead of comparing it bit by bit, window-wise comparison is done. In every corresponding window in both the Iris codes the number of 1's present is counted. If the difference in number of 1's in corresponding windows is greater than a threshold then difference between them is increased by one. Let us consider

$$D_{i,j} = \begin{cases} 1 & (IC^1(i,j)\otimes K) - (IC^2(i,j)\otimes K) \geq thresh \\ 0 & (IC^1(i,j)\otimes K) - (IC^2(i,j)\otimes K) < thresh \end{cases} \tag{12}$$

where K is kernel given by $K = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ and $\otimes$ stands for convolution. IC stands for iris code. Now if M and N are dimensions of iris code then the total difference is calculated by

$$Difference = \sum_{i=1}^{M}\sum_{j=1}^{N} D_{i,j} \tag{13}$$
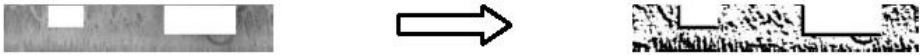
Fig. 6. Generation of iris code.

Dynamically the value of the threshold has been selected. It varies person to person. For calculating local threshold, distance between Iris codes of same person has been calculated using above method and finally local threshold is computed as

Local Threshold = Max. Difference + 0.5*(Max. Difference – Avg. Difference) (14)

3 Results

We have tested these above mentioned algorithms on the CASIA image database [4] and some iris images taken in non – infrared lighting environment by a CCD Camera.

The accuracy of iris detection algorithm is 100%. Table 1 shows the experimental results for verification using the three algorithms. Among the three algorithms, Circular Mellin Transform based algorithm gives the best results with an overall accuracy of 95.45%. Fig. 7.(a) shows results for the Circular – Mellin transform on CASIA Database and Fig. 7 (b) on the non-infrared images.

Input Image	Matched Image	Input Image	Matched Image
(a) Results on CASIA Database		(b) Results on Non–Infrared Iris Images	
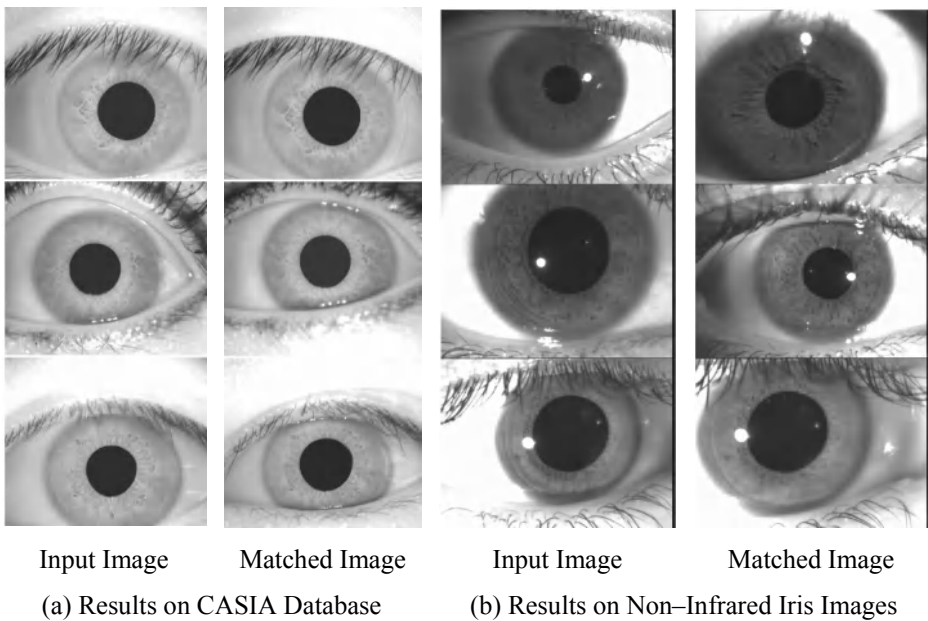
Fig. 7. Experimental Results of Circular – Mellin Transform

Table 1. Results of the Three Algorithms

Algorithm	FRR	FAR
Radial and circular features	1.89 %	23.26 %
Fourier Transforms	6.77 %	21.05 %
Circular – Mellin Transforms	3.52 %	5.58 %

4   Conclusion

In this paper new algorithms for iris verification have been presented. We have shown three algorithms for verification process namely 1 - Radial and circular features, 2 - Fourier Transforms and 3 Circular – Mellin Transforms. The experiments carried out our experiments on CASIA database and some non – infrared Iris images show that the algorithm based on Circular – Mellin Transform gives the best results in terms of false reject accuracy and false accepts accuracy and hence an overall accuracy of 95.45%. Some initial experiments on non–infrared iris images shows that this algorithm can work on such images but it still requires some more attention.

Acknowledgement

Reference

[1]   Canny J.: A computational approach to edge detection, IEEE transactions on Pattern Analysis and Machine Intelligence, Vol. 8, No. 6, (1986) 679 - 698.
[2]   Daugman, J.: High confidence visual recognition of persons by a test of statistical independence, IEEE transactions on Pattern Analysis and Machine Intelligence, Vol. 15, No. 11, (1993) 1148–1161.
[3]   Gopalan Ravichandran, Mohan M. Trivedi: Texture Segmentation using Circular-Mellin Operators. Proceedings of ICIP Vol. 2 (1994) 635-639.
[4]   http://nlpr-web.ia.ac.cn/english/irds/irisdatabase.htm.

Individual Recognition Based on Human Iris Using Fractal Dimension Approach

Peik Shyan Lee[1] and Hong Tat Ewe[2]

[1] Faculty of Engineering and [2]Faculty of Information Technology
Multimedia University, Selangor, Malaysia
`peikshyan@myjaring.net`

Abstract. In this paper, we present an approach for individual recognition system based on human iris using the estimation of fractal dimension in feature extraction. In this research, 500 iris images have been collected from different races for system validation. The attempt of capturing iris images in 320×240 resolution is intended to enable iris recognition in small-embedded system or portable devices with tight memory constraint and limited storage space. Hough transform and maximum vote find method are employed to localise the iris portion from the iris image. For feature extraction, a new approach based on fractal dimension is used to measure the important biometric information carried by human iris. A modified exclusive OR operator is designed to determine the failure of a match of two iris patterns. The experimental results show that the proposed method could be used to recognise an individual effectively.

1 Introduction

Iris recognition emerges to be one of the important areas in biometrics. In last ten years, there were numbers of iris recognition algorithms developed by researchers. Different approaches such as integro-differential operator and two-dimensional Gabor Transform [1], histogram-based model-fitting method and Laplacian Pyramid technique [2], zero-crossings of wavelet transform [3], multi-channel Gabor filtering [4], circular symmetry filter [5], two-dimensional multiresolution wavelet transform [6] and two-dimensional Hilbert Transform [7] were proposed. This paper examines a new approach for feature extraction in iris recognition system. In the new proposed method, fractal dimension approach is used to extract the biometric information carried by human iris. A fractal is an irregular geometric object with an infinite nesting of structure at all scales. Human iris can be characterised using fractal analysis because it exhibits a rich self-similarity and random patterns.

2 Biological Data Acquisition

Biological traits acquisition is the very initial step of any biometrics recognition system. In our research, iris images are collected from different ethnic groups, age and gender are taken for validation of the proposed system. Each volunteer

contributes 10 images, 5 from left eye and 5 from right eye. The acquired iris images are used for system validation. Images are taken with resolution in 320x240 which the system supports whereas many existing research works [1]-[7] in iris recognition reported system validation with iris images in size of 640x480. The taking of smaller resolution images may be useful for the consideration of iris recognition in small-embedded system or portable devices with tight memory constraint and limited storage space.

3   Image Processing

Information of iris features is needed from the acquired image. As such iris region of an eye image is required to be isolated correctly from irrelevant information in the stage of image processing. The process of boundary localisation is done to designate iris in an image. Assumption is made to take both inner and outer iris boundaries as circles and the circles are not co-centric. Outer boundary is easily determined using Canny edge detection method [8] and Hough Transform technique [9].

To reduce computing time of peak point detection in Hough Transform, a new method called maximum vote finding method is proposed and applied to locate inner boundary. The inner boundary is extracted based on the geometrical symmetry of a circle. For every row within the outer circle where two feature points are obtained through Canny edge detection, the center x coordinate is calculated. The final value of x coordinate with the maximum vote is chosen as the final center x coordinate of the inner circle. This process is repeated for the column direction to get the final center y coordinate. The radius of the inner circle can be calculated through the same method with the known final center coordinates.

Figure 1 show some sample iris images with localised boundaries. Two iris images from right are taken from volunteers wearing their spectacles and contact lenses. Maximum Vote Finding Method also performed a faster computation process than Generalised Hough Transform with the ratio of 0.60: 1.00.

After the iris portion is located, the iris image in polar representation is then mapped to a rectangular shape image with axes of θ and r, respectively as shown in Fig. 2. Due to irregularity of the iris radius for different individual at different time, images in polar representation are varied in dimension. As such, spatial normalisation based on interpolation is also done to standardise the size.
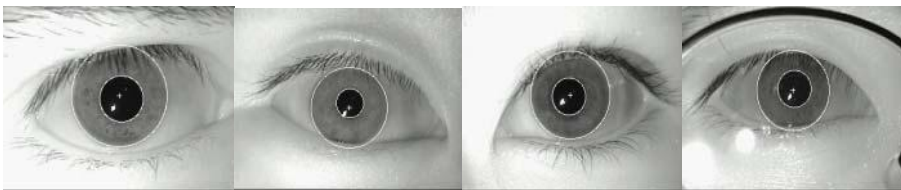
Fig. 1. Iris image with localized boundaries using Hough Transform and maximum vote finding method
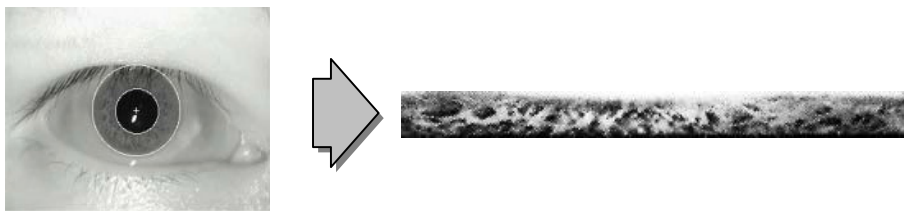
Fig. 2. Example of iris image transformed from circular shape to rectangular shape

4   Iris Signature Generation Using Fractal Dimension Approach

Fractal analysis provides a general framework to study representation of irregular sets found in natural phenomena [10]. Generally, natural objects exhibit scale invariant properties which are known as self-affinity. Koch, Hausdorff, Cantor, Sierpinski and Mandelbrot [10] have done the pioneer work to characterize the natural objects using fractal analysis. The work of analyzing texture in digital image using fractal geometry were also carried out [11]-[12]. It is found that fractal dimension is one of the important parameters that can be used to delineate characteristics of texture.

4.1 Feature Extraction with Conventional Fractal Dimension Approach

From the concept introduced by Mandelbrot [10], fractal dimensions of a self-similar object can be estimated. The fractal dimension, D, of an object is derived as:

$$D = \frac{\log(N)}{\log(\frac{1}{r})}$$

(1)

where D is fractal dimension, N_r is the number of scaled down copies (with linear dimension r) of the original object which is needed to fill up the original object.

The fractal information from a 2D image can be calculated by constructing it into three-dimensional representation [13]. By preserving the x-y mapping of an image, a third axis (h-axis) is constructed from its gray level value. Fractal dimension are calculated for this generated 3D surface within a defined area by the square window in x and y directions. An odd number of window size is chosen so that the window can be centred at a particular point and not between points. Appropriate window size, L, also ensures good extraction of textural information from an image. In a selected window, h for all points in the selected area is normalised by:

$$h_n = h \times \frac{L}{H}$$

(2)

where h_n is the normalised height and H (= 255) is the maximum gray level. This normalisation is required so that the calculated fractal surface dimension is within the limit of 3.0 (its topological dimension).

From equation (1), the calculation of fractal dimension is carried out using surface coverage method [14]. In this method, the total number of small squares (size 1 unit x 1 unit) needed to fill up the selected surface within the window, N_r, is calculated, and

D is obtained from equation (1) where r = 1/L in this case. The value of D is assigned to the center pixel (u_c, v_c) of the windows. The fractal dimensions of other points are obtained by using sliding window technique that moves the current window in x and y direction, and each time the surface bounded by the window is used to calculate the fractal dimension.

4.2 Feature Extraction with Modified Fractal Dimension Approach

A modified fractal dimension calculation based on dimension normalisation technique is also carried out to extract the textural information of iris. Previously, pixel height within a defined window is normalised before surface counting. In the modified method, the process of pixel height normalisation is not performed in order to maintain the original data set. But N_r is normalised by the surface area covered by window size, L. The normalised, N_r' is given by equation at (3).

$$N_{r}'n = \frac{N_{r}'}{(LxL)} = H^f \tag{3}$$

The normalised N_r' can be related to maximum gray level H, thus

$$f = \frac{\log(N_r'n)}{\log(H)} \tag{4}$$

where f is partial fractal surface dimension. The modified fractal surface dimension, D_m, can be obtained by adding physical dimension of 2.0 to it.

4.3 Signature Generation

Iris signature is constructed from the calculated fractal dimension. Fig.s 3 and 4 show the 3D plots of the D and D_m. The graphs show iris signature of a volunteer generated from the conventional and modified fractal dimension respectively.

5 Matching

A modified exclusive-OR operator is designed to measure the agreement of two iris signatures. The ratio of agreement between two iris signatures is then calculated. In previous feature extraction stage, the data for each dimension in iris signature is given in the range of 2.0-3.0 for both analysis methods of Fractal Dimension. As such, the XOR Boolean operator would produce an 'Agree' between two comparisons of pixel dimension if the observed value satisfy in the range of value stored in database.

Implementation of the operator can be formulated as below: Let FD_O denotes fractal dimension in observed iris image and FD_A denotes fractal dimension in iris image previously stored in database. Thus the XOR operation of the pair of dimensions would be given as:

$$FD_O \otimes FD_A = \begin{cases} 'Agree' & , \text{if } FD_A - C \leq FD_o \leq FD_A + C \\ 'Disagree' & , \text{else} \end{cases} \tag{5}$$
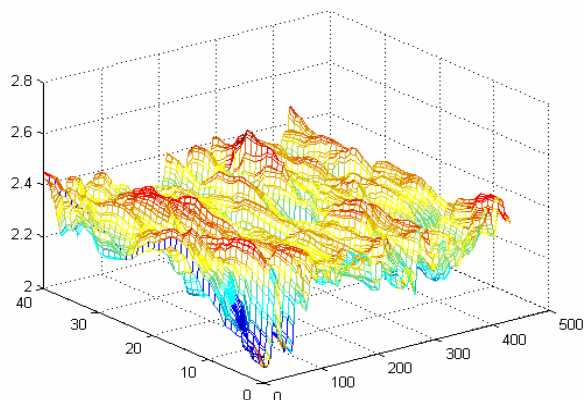
Fig. 3. 3D plot of the conventional FM
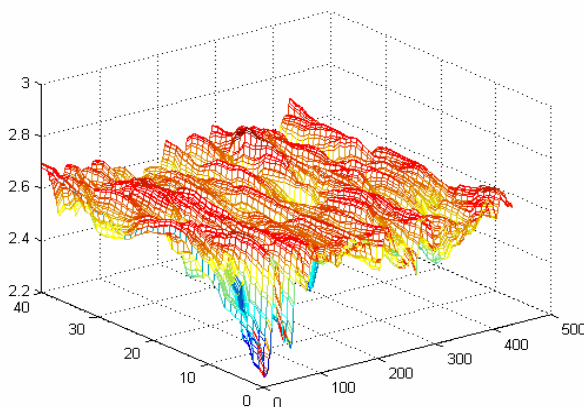
Fig. 4. 3D plot of the modified FM

where *C* is a constant to be determined. Ratio of agreement between comparisons of two iris signatures, *Agreement Ratio (AR)*, is defined as:

$$\text{Agreement Ratio} = \frac{\text{Total number of comparison with 'Agree'}}{\text{Total number of comparisons}} \quad (6)$$

Calculated *Agreement Ratio* exceeding threshold will be accepted as successful authentication and matched to the enrolled user in the system. Threshold is determined to pass or fail identification. If the measured *Agreement Ratio* is lower than threshold, an imposter is rejected whereas if the measured *Agreement Ratio* is higher than the threshold, an enrolled user is identified.

6   Experimental Results and Discussions

Statistical test is carried out to measure the accuracy and identification strength of the system. False Accept Rate (FAR), False Reject Rate (FRR) and Equal Error Rate (EER) of the system with Fractal Dimension approach are examined. The validation is done using 500 real iris images. The images are taken from 50 volunteers where each volunteer provided 5 of their left eye images and 5 right eye images.

Table 1 shows the recognition produced by different window sizes and values of constant. From the table, it is found that both conventional and modified fractal dimensions with 11x11 window size of constant values $C = 0.065$ and 0.04 respectively hive the best recognition rate among the different combinations.

Fig.s 5 and 6 show the change of two error rates according to the threshold value from feature extraction method using conventional fractal dimension approach. From the graph we can observe that the Equal Error Rate happens when the threshold is set at 56.5 and the value of EER is 2%. For modified fractal dimension, the EER is 3.9% when the threshold is set at 54.5.

We have also tested our algorithm with iris images where the volunteers wear their contact lenses and spectacles. 50 iris images were tested using the feature extractor of conventional fractal dimension approach with window size 11x11. There are 10 groups of images in 50 iris images with contact lenses or spectacles. 7 groups of them are wearing spectacles whereas the rest are wearing contact lenses.

Table 1. Recognition rate of iris signature in spatial domain resulted from the proposed feature extraction method

Feature Extractor / Constant (C)		FAR (%)	FRR (%)
Conventional Fractal Dimension/ 0.02	11x11 window	0.62	8.00
	15x15 window	0.70	9.00
Conventional Fractal Dimension/ 0.065	11x11 window	1.03	4.00
	15x15 window	1.18	5.00
Conventional Fractal Dimension/ 0.09	11x11 window	1.86	3.00
	15x15 window	2.00	4.00
Modified Fractal Dimension/ 0.01	11x11 window	0.74	8.00
	15x15 window	0.80	9.00
Modified Fractal Dimension/ 0.04	11x11 window	1.09	5.00
	15x15 window	1.22	6.00
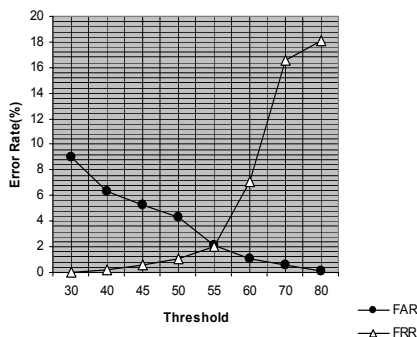
Fig. 5. Change of FAR and FRR according threshto threshold value for conventional FM
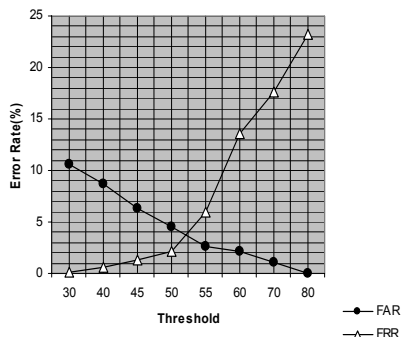
Fig. 6. Change of FAR and FRR accord ing to threshold value for modified FM

5 groups of the images achieved 100% successful matching rate and 5 other groups with above 80% successful matching rate. The high recognition rate among these iris images show that the algorithm can be applied on iris image with contact lenses or spectacles.

7   Conclusion

In this paper, a new method of iris recognition system is presented. The system uses Hough Transform and a new maximum vote finding method to localize iris portion in an iris image. Fractal Dimension Approach is used to extract textural information of human iris. The implementation shows that the iris recognition based on fractal approach has given satisfactory matching results and is also applicable for cases with contact lenses and spectacles.

References

[1]   Daugman, J.G.: High Confidence Visual Recognition of Persons by a Test of Statistical Independence. IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 15, no. 11, Nov 1993, pp 1145-1161.
[2]   Wildes, R. P., Asmuth, J.C., Green, G.L., Hsu, S.C., Kolczynski, R.J., Matey, J.R. and McBride, S.E.: A System for Automated Iris Recognition. Proceedings of the Second IEEE Workshop on Applications of Computer Vision, Florida, U.S.A., Dec 1994, pp. 121-128.
[3]   Boles, W.W.: A Security System Based on Human Iris Identification Using Wavelet Transform. First Conference on Knowledge-based Intelligent Electronic Systems, Adelaide Australia, May 1997, pp. 533-540.
[4]   Zhu, Y., Tan, T.N. and Wang, Y.H.: Biometrics Personal Identification Based on Iris Patterns. 15th International Conference on Pattern Recognition, Barcelona Spain, Sept 2000, pp. 801-804.

[5]   Ma, L., Wang, Y.H. and Tan, T.N.: Iris Recognition Using Circular Symmetrics Filters. 16th International Conference on Pattern Recognition, Quebec Canada, Aug 2002, pp. 414-417.

[6]   Lim S., Lee K., Byeon O. and Kim T.: Efficient Iris Recognition Through Improvement of Feature Vector and Classifier. Electronics and Telecommunications Research Institute (ETRI) Journal, Vol. 23, No. 2, June 2001, pp. 61-70.

[7]   Tisse C., Martin L., Torres L. and Robert M.: Person identification technique using human iris recognition, 15th International Conference on Vision Interface, Calgary Canada, May 2002, pp. 294-299.

[8]   Canny, J.: A Computational Approach to Edge Detection. IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol 8, 1986, pp. 679-700.

[9]   Gonzalez R.C. and Woods R.E.: Digital Image Processing. Addison Wesley, 1933.

[10]   Falconer K.: Fractal Geometry Mathematical Foundation and Applications. John Wiley and Son, 1990.

[11]   Pentland A.P.: Fractal based description of natural scenes. IEEE Transactions on Pattern and Machine Intelligence, vol: 6(6), 1984, pp. 661- 674.

[12]   Dubuisson M. P.and Dubes R.C.: Efficacy of Fractal Features in Segmenting Images of Natural Textures. Pattern Recognition Letters, vol: 15(4), April 1994, pp. 419-431.

[13]   Low H.K., Chuah H.T. and Ewe H.T.: A Neural Network Landuse Classifier for SAR Images using Textural and Fractal Information. Geocarto International, Vol. 14, No. 1, 1999, pp. 67-74.

[14]   Ewe H.T., Au W.C., Shin R.T. and Kong J.A.: Classification of SAR Images Using a Fractal Approach. Proceedings of Progress in Electromagnetics Research Symposium, Los Angeles, U.S.A., 1993, pp. 493.

Iris Recognition in Wearable Computer

Jeong Jun Lee[1], Seungin Noh[1], Kang Ryoung Park[2], and Jaihie Kim[1]

[1] Computer Vision Lab.
Dept. of Electrical and Electronic Eng.
Biometrics Engineering Research Center,
Yonsei University
Seoul, Republic of Korea
{jjlee,sinoh,jhkim}@yonsei.ac.kr
[2] Division of Media Technology
Sangmyung University
Seoul, Republic of Korea
parkgr@smu.ac.kr

Abstract. Recently, there has been much progress in wearable computing. Input/output interface and security are important research areas in wearable computing. In this paper, we describe considerations which should be taken into account in designing a wearable interface and iris recognition system. One of the most important issues in designing a wearable iris recognition system is the radial lens distortion. We suggest a radial distortion elimination method which is simple but accurate. Experimental results show that the quality of the image captured by our system is good enough for iris recognition and we can achieve 0.18% of the EER by applying our proposed radial distortion elimination method.

1   Introduction

Recently, computers have become smaller and more powerful, and even wearable computers have appeared. Wearable computers are already being used in many fields like medicine, the military, industrial training, education and entertainment, and the number of fields using wearable computers is likely to increase [1]. In addition, network access in a mobile computing environment for internet, e-mail and electronic commerce etc. has increased.

One of the most important issues in developing a wearable computer is the interface. For an output device, a monitor is widely used and today there are many wearable monitors which are commercially available [2][3][4]. On the contrary, for an input device of the computer, there are many kinds of different method; keyboard, mouse, etc. However, in a wearable computer system, a keyboard or mouse is cumbersome. Since the monitor is inevitable in many cases, the input interface using a gaze point on a monitor screen can be a good alternative for a keyboard and mouse.

Another important issue in a wearable computer is security. When a user accesses a network for LAN, e-mail or electronic commerce, etc. in a mobile

environment with a wearable computer, user authentication is essential. In addition, the login of the wearable computer itself needs user authentication. Iris recognition can be used for such authentication. The camera which is used for gaze detection can also be used for iris recognition. By using one small camera attached on a wearable monitor, both the interface problem and the security problem can be solved. We developed such a gaze detection system [5] and an iris recognition system using one small camera, IR-LEDs and a head mounted display (HMD). In this paper, we will mainly discuss the points which should be considered in iris recognition in a wearable computer environment.

The following should be considered when designing an iris recognition system under a wearable computer environment:

1. *Weight and size:* The weight/size of the equipment including camera, lightings (LEDs) etc. should be as light/small as possible because we wear them.
2. *Power consumption:* Because of the limited power supply in mobile computing, it is desirable for the camera to consume as little power as possible.
3. *Wavelength of LED:* Since in our case the LEDs are placed very close to the eye, they can dazzle the user's eye unless it is invisible light. For this reason, infrared light at around an 850nm wavelength band is suitable.
4. *Radial distortion:* Because a very wide angle lens is used to capture the eye image at a close distance from the eye, it is likely to show lens distortion like radial distortion [6]. It should be eliminated.

There are also advantages due to wearable monitor and camera environment.

1. *Exclusion of outer light:* The head mounted display itself plays a role of shutter of the outer light which may degrade the quality of the iris image.
2. *Small power of lighting (IR-LED) is enough:* Because the head mounted display cuts off much of the outer light, there is no need of strong infrared lighting and electric shutter to eliminate the outer visible light. In addition, since the distance between the LED and the eye is very close, strong lighting is not needed.
3. *A CMOS-type image sensor can be used:* In a wearable computer environment a CMOS-type camera is preferred over a CCD-type camera because it can be made smaller and lighter with less power consumption than a CCD-type camera. The only disadvantage is that a CMOS is less sensitive than a CCD in infrared wavelength band. However, since the camera is located very close to the eye, that nolonger matters. Experimental results show that the quality of the image taken by the CMOS sensor is as good as that taken by the CCD sensor.
4. *An auto zoom/focus lens is not needed:* When a user wears a head mounted display, he/she adjusts it so that the monitor screen can be seen most clearly in front of his/her eye. By this adjustment process, the user's eye is always located at an almost fixed X/Y/Z position from the HMD and camera. As a result, there is no need of an auto zoom or focus lens, because the user himself/herself gets his/her eye to the position where the image is in focus.
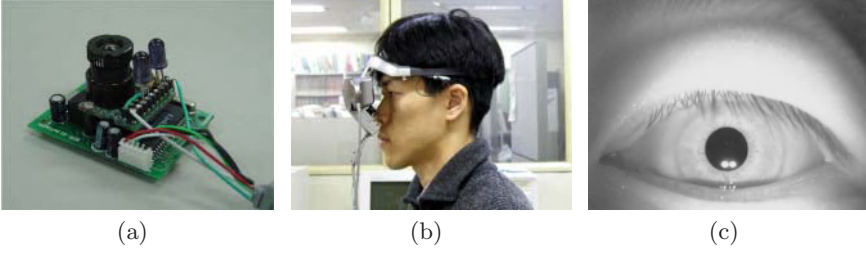
Fig. 1. (a) Proposed small camera system (b) Prototype of our gaze detection and iris recognition system (c) Captured eye image

Most former research and products are based on the assumption that the user's eye is located relatively far away from the camera. Many of them adopt auto zoom/focus lenses because the distance between the user and the camera is not constant. The LG IrisAccess3000 and the Panasonic BM-ET500 were developed for the use of door access control, and it adopts a CCD imaging sensor and auto zoom/focus lens [12][13]. The Panasonic BM-ET100 was designed to work at 48-53cm away from the user, and it does not show the lens distortion which we considered in this paper. Most former products are developed for the use of the door or PC access control. The development or optimization of iris recognition in a wearable computer has not yet been reported.

In this paper, hardware implementation of our system(sec.2), a lens distortion elimination method(sec.3) and the experimental results(sec.4) are described.

2 Hardware Implementation

We made a prototype of gaze detection and iris recognition system which consists of a wearable monitor, a small camera and infrared LEDs. For the wearable monitor, the Daeyang Cy-visor DH4400 [2] was adopted. The screen resolution is 800×600 pixels and the viewing angle of it is 31.2°. For the small camera, we employed a CMOS-type camera considering a simple circuit, lightweight and low power consumption. The effective pixels of the camera are 640(H)×480(V). For the infrared illumination, we used two IR-LEDs whose wavelengths are 850nm, respectively, which is invisible to eye. The viewing angles of the two LED are 44°, respectively, which covers the whole eye region when they are 3cm away from the eye. The positions of the LEDs are selected so that the corneal reflections (glints) may not exist in an iris area but in the pupil area in the image. For that reason, the LEDs are placed as close to the lens as possible. The viewing angle of the lens is about 70°, which is relatively large because the camera is positioned very close to the eye. Figure 1(a) shows our proposed camera, lens and LED lighting system. The dimension is 32(W)×36(H)×25(D) mm³ and the weight is 20g without cable. Figure 1(b) shows the prototype of our system.

(a)   (b)
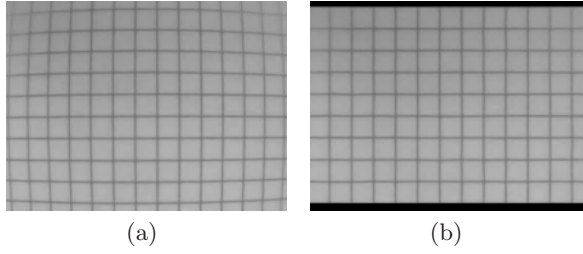
Fig. 2. (a) Radial lens distortion of our iris recognition camera (b) Undistorted image

3   Elimination of Lens Distortion

In the case when a user authentication is needed, the user should glance toward the camera. Figure 1(c) is the image captured at this time. The image quality is good enough for iris recognition. However, there is a radial lens distortion. Generally, wide-angle lenses tend to distort the image especially near its boundaries [6]. In a wearable monitor and camera system, since the camera should image the whole eye region at a very close distance, the viewing angle should be very large and consequently lens distortion happens. In our case, the distance between camera and eye is about 3cm and the eye size is about 4.2cm, which results in 70° of the viewing angle. Figure 2(a) shows the lens distortion. The image was taken with a grid board being perpendicular to the optical axis. The grid on the board is an exact square grid. To eliminate such distortions, specially designed multiple optical lenses are needed, but they are expensive. Above all, they cause the system to be heavier and more voluminous, which is very undesirable in a wearable computer. In our research we eliminated the lens distortion only by a software algorithm without using any special multiple optical lenses, and we prevented the performance drop of iris recognition due to the distortion.

Lens distortion is composed of two types: radial distortion and tangential distortion. It is known that tangential distortion is very small and can be ignored [6]. As a result, lens distortion can be modeled as a radial distortion as follows:

$$x_u = x_d(1 + kr_d^2) \qquad y_u = y_d(1 + kr_d^2), \tag{1}$$

where (x_u, y_u) is an "undistorted" image point corresponding to the "distorted" image point (x_d, y_d), and $r_d = \sqrt{x_d^2 + y_d^2}$ is the distance between (x_d, y_d) and the image center. Note that the origin of all coordinates is the image center. k is the distortion parameter.

The classic method for correction of the lens distortion is to use a calibration object with known 3D coordinates [6][7]. The calibration object is generally made as big as possible to measure the 3D position precisely, because in such a method a small error in 3D position may cause a large estimation error. However, in our case where the working distance of a camera is very short, we can only use a small-sized calibration object and as a result it is difficult to measure the

(a)   (b)
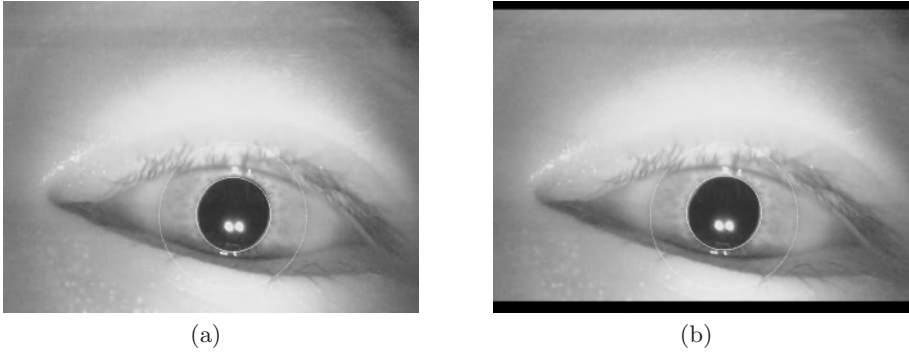
Fig. 3. Boundary detection results of (a) distorted image and (b) undistorted image

dimension precisely. Methods utilizing a special camera motion (pure rotation) are reported [8][9]. However, it is not easy to implement such a motion.

In our research we used a perspective constraint that, under a perspective projection, all the squares of the same size in a plane perpendicular to the optical axis project to squares of all the same size in the image plane. This method is simple but accurate. The following are the steps to compute the parameter k:

1. Under the constraint of *same sized squares*, compute the new positions of the undistorted squares' vertexes by using the square in the image center. We assumed that the square in the image center is not distorted.
2. Extract the crossing points of the lines (or curves) in the captured grid image.
3. Apply a least-squares method to the Eq. 1 to find the parameter k .

Once the parameter k is obtained, it can be applied to every eye image to remove the radial distortion. Experimental results indicate that the distortion parameter(k) of our camera is 3.195×10^{-7}. Figure 2(b) is the image obtained by removing the distortion of the original image using k computed above.

4   Experimental Results

4.1   Boundary Detection of Pupil and Iris

We employed the well-known circular edge detection algorithm to extract the boundaries of the pupil and iris [10]. Figure 3(a) and (b) show the examples of the results. We can see a minute difference between the two pupil boundary detection results. The difference is 2 or 3 pixels, but it may cause significant degradation of the recognition performance. It is because the pupil's shape is deformed when it is located away from the image center. Our goal is *to make the iris and pupil region have a regular shape wherever the pupil's position is in the image.* Experimental results show that, in the case of distorted image, the pupil's (or iris') shape is deformed especially when it is located away from the
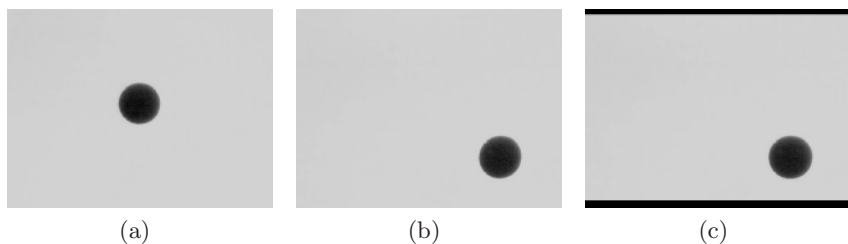
Fig. 4. Deformation of a circle (pupil or iris) (a) deformation does not happen when it is around the image center (b) deformation happens when it is away from the image center (c) undistorted image

image center. On the other hand, in the case of an undistorted image, the pupil's (or iris') shape is regular wherever its position may be in the image.

We tested our lens distortion elimination algorithm using a synthesized circle image. The circle can be thought of as a pupil or iris. The synthesized circle image was made by a laser printer (600dpi). When the pupil (or iris) is located away from the image center, the shape is deformed to almost an ellipse (Fig. 4(b)), which causes an irregular pupil (or iris) boundary extraction and a deformation of iris pattern. By applying our algorithm we can restore the pupil's original shape, a circle (Fig. 4(c)). Figure 5(c) shows an example of the difference in the 1-D iris signal between a distorted image and undistorted image. The two 1-D iris signals were obtained at the same position in the stretched 2-D iris pattern (Fig. 5(a) and (b)). The difference is due to the deformation of the pupil and iris's shape.

4.2   Iris Recognition Performance

The test data were acquired from 90 Asians from the ages of 20-50, and 20 images were acquired from each person (total 1800 test images). Most iris colors are brown. Thirty persons wore glasses, 10 persons wore contact lenses and 50 persons had no glasses or lenses. The image size is 640×480 pixels and the diameter of the iris is about 230-250 pixels.

We made two kinds of experiments; First, we compared the performance of the CMOS sensor with the CCD sensor. All other conditions like IR-LED and HMD etc. are equal except that the Samsung CCD camera MPC-C30 (resolution: 640×480 pixels) was used. The number of authentic tests is 1710 and the number of imposter tests is 160,200. Experimental results (Table 1) show that the CMOS camera produces as high quality image for iris recognition as the CCD camera in our environment. The Gabor filter method [10] was used for iris recognition.

Second, we examined the effect of radial lens distortion on iris recognition by the Gabor filter method [10] and the ICA method [11] using the CMOS sensor. From the results (Table 2), we found that the distorted image degrades the recognition performance significantly. However, by applying our proposed radial

(a)

(b)

(c)
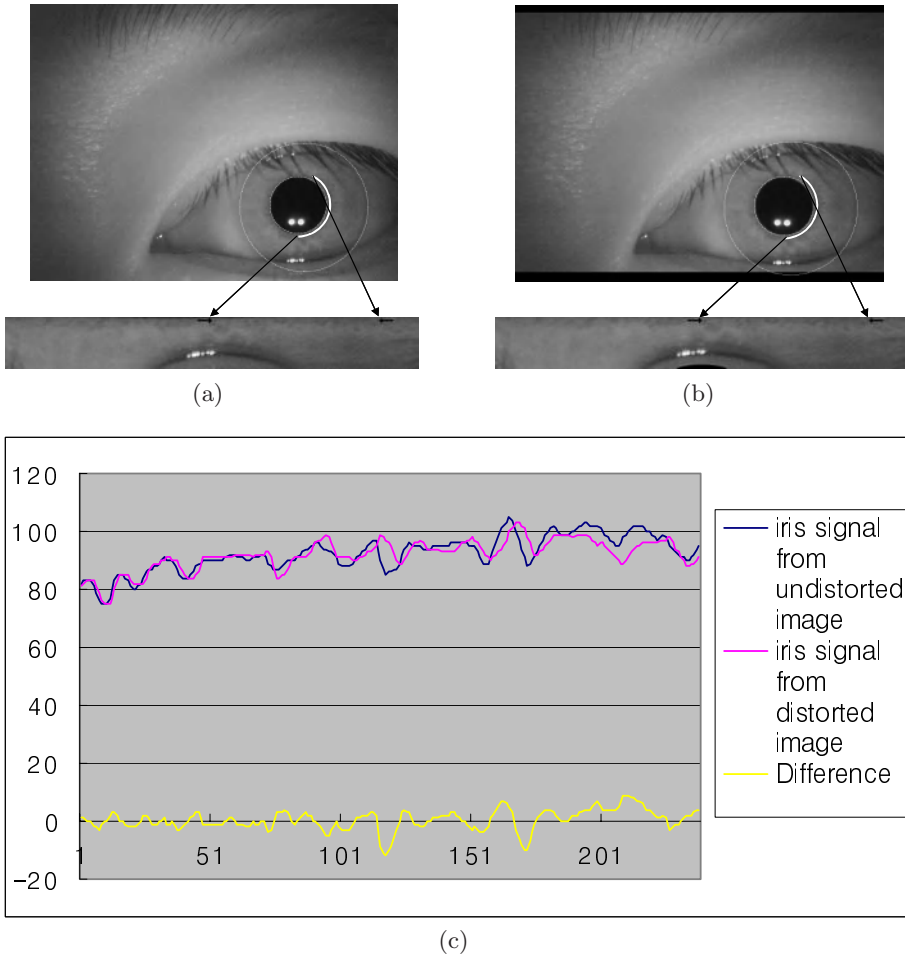
Fig. 5. (a)distorted image (b)undistorted image (c)two 1-D iris signals and the difference

distortion elimination process, we could get about 0.18% of the Equal Error Rate (EER). Because the CMOS image sensor is known to be noisy (in other words, have a high frequency), we performed an experiment on several Gabor filter frequencies. However, experimental results show that it does not make any significant difference in recognition accuracy as long as the frequency is between $\pi/16$ and $\pi/8$. Experimental results also indicate that the two methods (Gabor filter and ICA) have similar performances.

Table 1. Equal Error Rate (%)

Sensor / Image	CCD	CMOS
Undistorted image	0.170	0.173

Table 2. Equal Error Rate (%)

Method / Image	Gabor	ICA
Distorted image	0.422	0.431
Undistorted image	0.173	0.180

5   Conclusion

We developed a prototype of the gaze detection and iris recognition system using one small camera, two IR LEDs and a head mounted display (HMD) which can be used in a wearable computing environment. In the case where a wide-angle lens is used like in our system, radial lens distortion should be eliminated because it causes irregular iris boundary detection and deformation of the iris pattern. Experimental results show that the CMOS camera produces as high quality image for iris recognition as the CCD camera in our environment. In addition, by applying our proposed radial distortion elimination method we achieved about 0.18% of the Equal Error Rate.

Acknowledgements

References

[1] P. Huang, "Promoting wearable computing: A survey and future agenda," Technical Report TIK-Nr.95, Computer Engineering and Networks Laboratory, Swiss Federal Institute of Technology, Sep. 2000. 475
[2] http://www.personaldisplay.com/english/default.html. 475, 477
[3] http://www.microopticalcorp.com/HomePage.html. 475
[4] http://www.xybernaut.com/home.asp. 475
[5] J. J. Lee, K. R. Park and J. Kim, "Gaze detection system under HMD environment for user interface," In Supplementary Proc. of Int'l Conf. on Artificial Neural Network, pp. 512-515, Istanbul, Turkey, Jun. 2003. 476
[6] R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," IEEE Journal of Robotics and Automation vol. 3, no. 4, pp. 323-344, Aug. 1987. 476, 478

[7] J. Weng, P. Cohen, and M. Herniou, "Camera Calibration with Distortion Models and Accuracy Evaluation," IEEE Trans. on Patt. Anal. and Mach. Intell. vol. 14, no. 10, pp. 965-980, Oct. 1992. 478

[8] G. P. Stein, "Accurate Internal Camera Calibration Using Rotation, with Analysis of Sources of Error," In Proc. Fifth Int'l Conf. Computer Vision, pp. 230-236, 1995. 479

[9] F. Du and M. Brady, "Self Calibration of the Intrinsic Parameters of Cameras for Active Vision Systems," In Proc. of CVPR 93, pp. 477-482, New York, Jun. 1993. 479

[10] J. Daugman, "High confidence visual recognition of persons by a test of statistical independence," IEEE Trans. on Patt. Anal. and Mach. Intell. vol. 15 no. 11 pp. 1148-1161, 1993. 479, 480

[11] K. Bae, S. Noh and J. Kim, "Iris Feature Extraction Using Independent Component Analysis," LNCS, 4th Int'l Conf. AVBPA, pp. 838-844, Guildford, UK, Jun. 2003 480

[12] http://www.lgiris.com. 477

[13] http://www.panasonic.com/cctv/products/biometrics.asp. 477

Iris Recognition Based on Location of Key Points

Wen Yang[1], Li Yu[2], Guangming Lu[2], and Kuanquan Wang[2]

[1]Shenzhen Graduate School of Harbin Institute of Technology,
Shenzhen, China
yangwen@hit.edu.cn
[2]School of Computer Science and Technology
Harbin Institute of Technology,
Harbin, China
lyu@hit.edu.cn
csglu@comp.polyu.edu.hk
wangkq@hope.hit.edu.cn

Abstract. This paper proposes a new iris recognition method based on the location of key points. When preprocessed, the annular iris is normalized into a rectangular image. Multi-channel 2-D Gabor filters are used to capture the feature of iris texture. In each filtered sub-image, points that can represent the local texture feature most effectively in each channel are extracted. The barycenter of these points is the so-called key point. For a given iris, the location of key points is recorded as its feature vector. The iris feature matching is based on the Euclidean distance between the corresponding key points. Experimental results illustrate the effectiveness of this method.

1   Introduction

With the rapid development of information technology, people are getting increasingly reliant on various information systems. How to control the access to the important information is a challenging problem. Traditional ways such as passwords, ID cards are widely used. However, passwords are easy to be forgotten or confused, and guessed by imposters and ID cards are easy to be lost or stolen. Therefore, people want to find new tools to solve these problems. The unique physical character of body can not be borrowed, stolen, or forgotten, so biometric-based recognition is one of the most secure and convenient authentication technologies. Common characteristics include fingerprint, hand or palm geometry, retina, iris, face, signature, voice, keystroke pattern, gait and so on. So far, all above biometric technologies have been developed. Among all of these methods, iris recognition is the most promising one for high security environments and noninvasive for users.

There is abundant texture information in iris. The iris texture pattern is encoded into a 256-byte iris code by means of using some 2-D Gabor filters in Daugman's system[2]. The Wildes *et al.* system makes use of an isotropic band-pass decomposition derived from application of Laplacian of Gaussian filters[7]. Boles uses 1-D wavelet transform for iris identification[8], but he only uses one-dimensional information of the iris texture. Ma *et al*. propose some methods for iris recognition based on some spatial filters[3], [4], [5]. Most of the mentioned methods

or systems make use of the texture analysis. However, according to our iris samples (shown in Fig. 1), there are various minutiae in iris image.

Minutiae-based recognition method is widely used in fingerprint recognition systems. The two most prominent structures are ridge endings and ridge bifurcations[1]. The fingerprint pattern identification is accomplished by matching the corresponding minutiae. The feature point-based approach is used in palmprint recognition system[6]. As there are rich structures in iris texture, it is difficult to define and detect the minutial structure of iris texture. Structure-based iris recognition systems have seldom been introduced until now.

This paper proposes a novel method for iris recognition based on location of key points (not feature points). It dose not attempt to define and detect the structure of local iris texture as in fingerprint or palmprint recognition systems, however, it uses a group of 2-D Gabor filters to detect a set of points that can represent the local texture information most effectively. Subsequently, an effective approach to measure the distance of two feature vectors is used in this paper.

The rest of this paper is organized as following: Section 2 introduces the preprocessing of iris image and Section 3 presents an introduction to the algorithm for key points extraction. The experimental results are addressed in Section 4. The paper ends with conclusions.

2   Image Preprocessing

In a captured iris image, the gray level of point (x, y) is described as $I(x, y)$. After the operation for locating an iris, the centers of the inner boundary and the outer boundary are located. To exclude the eyelids and eyelashes, only the inner 3/4 of the lower half of an iris is used. As the captured iris image always varies in size, the detected iris is normalized into a rectangular block. The preprocessing ends with histogram equalization. In our experiment, the preprocessed images are transformed into images with size of 256×64. The procedure of preprocessing can be illustrated in Fig. 2.
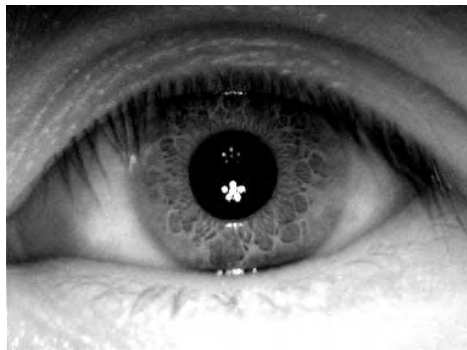
Fig. 1. An iris image with minutiae
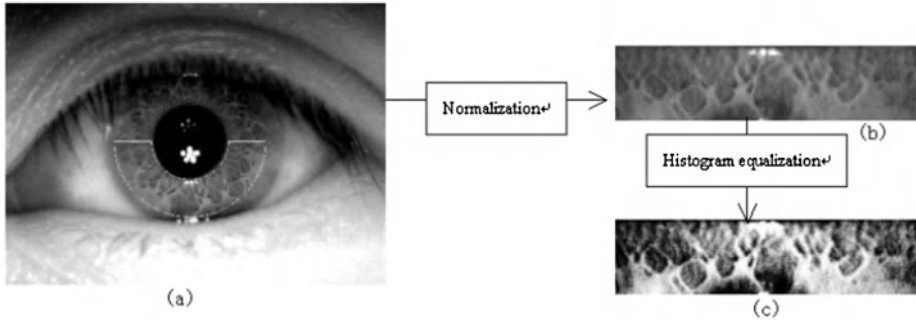
Fig. 2. Preprocessing of iris image, (a) the original image, (b) the normalized iris texture and (c) the normalized iris texture after histogram equalization

3   Key Points: Feature Extraction

The iris texture is much more complicated than fingerprints and palmprints. Because of the variety, randomicity and scrambling in iris texture, it is challenging to classify the minutial iris texture structure into some classes. It is difficult to capture the feature points of iris through the approaches that are widely used in fingerprint recognition systems and palmprint recognition systems. Therefore, some 2-D Gabor filters instead of a few types of minutial structures are used to describe the local texture information.

When we apply a filter to an image, the coefficient describes the similarity between the filter and the local regions. Though we can not define a few types of structures for iris texture and extract these minutial structures from iris texture, we can take advantage of a few types of filters to extract feature points that are most similar to the some filters. 2-D Gabor filter is the ideal filter. In our method, 2-D Gabor filters are used to extract feature points.

In (x, y) coordinate system, a 2-D Gabor filter can be given by:

$$G(x, y, f, \theta) = e^{-\frac{x'^2}{2\alpha^2} - \frac{y'^2}{2\beta^2}} \cdot e^{-\frac{2\pi i}{T} x'} \tag{1}$$

where α and β are the width and height (standard deviation) of 2-D Gaussian function respectively, $i = \sqrt{-1}$ is the imaginary unit, T is the periods(or wavelength) in spatial domain, and x' and y' are given by

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} \tag{2}$$

where θ is the angle between (x, y) coordinate system and (x', y') coordinate system. Each 2-D Gabor filter has an even-symmetric real part and an odd-symmetric imaginary part.

To get texture information in various directions and different scales, multi-channel 2-D Gabor filters are used. Each channel is corresponding to a direction in different scales. As the symmetry of 2-D Gabor filters, we set 4 different values for θ: $0°$, $45°$, $90°$, $135°$. To extract texture information in different scales, we set the wavelength to 4 discrete values, so there are $4 \times 4 = 16$ channels, 32 filters. Then we get 32 filtered images.

In each filtered image, those points with the largest absolute value are most similar to the corresponding filter. These points can be looked as feature points. Due to the intensity of illumination, the noise and other factors, there are many points with nearly the same coefficients. It is not reliable to choose only one point that has the largest coefficient as the feature point, so a fixed number of feature points are gotten. Then the barycenter of the set of points is the key point, whose location is recorded in the feature vector.

Let $F(x, y)$ be the absolute value of a filtered image, and

$$LFP = \{(x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m)\} \tag{3}$$

be the location of feature points and m be the fixed number, then the location of key point (x_{KP}, y_{KP}) can be given by:

$$\begin{cases} x_{KP} = \dfrac{\sum\limits_{1}^{m} x_i F(x_i, y_i)}{\sum\limits_{1}^{m} F(x_i, y_i)} \\ y_{KP} = \dfrac{\sum\limits_{1}^{m} y_i F(x_i, y_i)}{\sum\limits_{1}^{m} F(x_i, y_i)} \end{cases}, (x_i, y_i) \in LFP \tag{4}$$

To capture the local texture information, the process of researching key points is performed in some sub-images. In our experiment, we extract key points in each 64×64 block (1/4 sub-image) and 32×32 block (1/16 sub-image). As there are 32 filters (16 channels), we can get 32 key points in each block. Thus, there will be 128 and 512 key points. Fig. 3 shows the 128-dimensional key points in normalized iris image.

The location of these key points (LKP) is gained as the feature vector of an iris, which is defined as:

$$LKP = \{(x_{KP1}, y_{KP1}), (x_{KP2}, y_{KP2}), \cdots, (x_{KPn}, y_{KPn})\} \tag{5}$$

where n is the dimension of the feature vector.
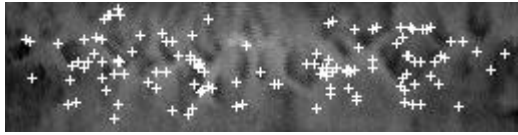
Fig. 3. Key points (marked with +) in a normalized iris image

4   Performance: Experimental Results

The similarity of two irises can be measured by the mean of Euclidean distance between the corresponding key points. Let LKP_1 and LKP_2 be the feature vector of two different iris samples, the mean of Euclidean distance D can be obtained by:

$$D = \frac{\sum_{1}^{n}\sqrt{(x_{1i} - x_{2i})^2 + (y_{1i} - y_{2i})^2}}{n} \tag{6}$$

where $(x_{1i}, y_{1i}) \in LKP_1$, $(x_{2i}, y_{2i}) \in LKP_2$.

In order to test the performance, we collected a set of iris samples. We use 40 irises in our experiment. Eight samples were captured for each iris, thus there were 320 iris images with 768×568 pixels in 256 gray levels. When preprocessed, region of interesting (ROI) is normalized into an image with 256×64 pixels. There are four kinds of experiment schemes: one to four sample(s) of each iris is(are) randomly selected for training and the left samples are used for testing.

Based on the mentioned schemes, the experiment for identification is carried out with 128-dimensional feature vector and 512-dimensional feature vector. The results are listed in Table 1, from which we can know that our method gets the best result with four training sets and 512-dimensional feature vector.

The experiment for verification is based on four training sets and 512-dimensional feature vector. In each matching, a decision of acceptance or rejection is made. If a pair of samples from one iris is rejected or a pair of samples from different irises is accepted, incorrect decisions will be made. These two types of errors are usually described as false reject rate (FRR) and false accept rate (FAR). Fig. 4(a) shows the distributions of the mean of Euclidean distance between authentics, and between imposters. Fig. 4(b) shows curves for the FAR and FRR with different threshold value. When the threshold value is set to 7.58, our method achieves the best result with FAR=0.03% while FRR=1.88%.

Table 1. Identification rate with different feature length and training samples

Feature Length	Training samples			
	1	2	3	4
128	91.429%	95.833%	96.000%	96.250%
512	92.857%	98.333%	99.500%	100.00%
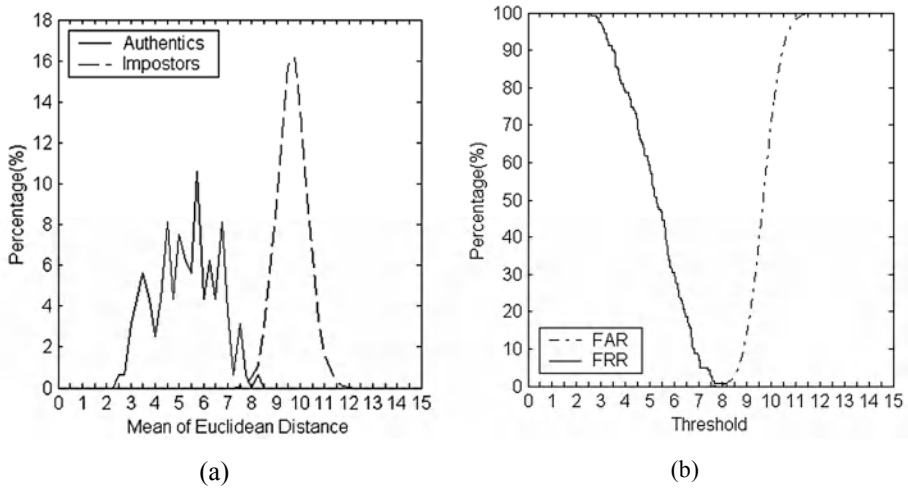
Fig. 4. Experimental results for verification. (a) Authentics and imposters distributions and (b) the FAR and FRR with different threshold

As mentioned above, many approaches for iris recognition have been developed. However, not all methods are tested by plentiful samples. Our method are somewhat similar to the methods proposed by Ma *et al*.[3], [4], [5]. By means of using some spatial filters, Ma *et al.* get an identification rate of 99.43%[5] with 2255 images. From the identification point of view, our method gets higher recognition rate than theirs. But, our samples are limited not only quantitatively but also qualitatively. In addition, due to the quality of our samples, we only use the inner 3/4 of the lower half of an iris, which provides less information. Compared with Daugman's system, Ma's methods, our algorithm for feature extraction is operated on image with comparatively less size, which means the lower requirement for system memory and the less time for computing. Anyway, the experimental results in our method are quite considerable.

5   Conclusions

In this paper, we have presented a novel method for iris recognition based on location of key points. A group of multi-channel 2-D Gabor filters is used to capture the local texture information in some sub-images. Key points are extracted from the filtered sub-image by getting the barycenters of points those have lager coefficients in each channel, then the location of key points is recorded as feature vector. The mean of Euclidean distance between two feature vectors is applied to calculate the similarity of two irises. Based on the proposed method, we get an encouraging recognition rate in our experiments.

Acknowledgements

References

[1]   Jain, L. Hong, and R. Bolle, On-line Fingerprint Verification. IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 19, No. 4, Apr 1997, 302-314

[2]   J. G. Daugman, High Confidential Visual Recognition by Test of Statistical Independence, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15, No. 11, Nov 1993, 1148-1161

[3]   Li Ma, Yunhong Wang, Tieniu Tan, Iris Recognition Based on Multichannel Gabor Filtering, Proceedings of ACCV 2002, Vol. I, 279~283

[4]   Li Ma, Yunhong Wang, Tieniu Tan, Iris Recognition Using Circular Symmetric Filters, the Sixteenth International Conference on Pattern Recognition, Vol. II, 2002, 414-417

[5]   Li Ma, Tieniu Tan, Yunhong Wang, Dexin Zhang, Personal Identification Based on Iris Texture Analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 12, Dec 2003, 1519-1533

[6]   Nicolae Duta, Anil K. Jain, Kanti V. Mardia, Matching of Palmprints, Pattern Recognition Letters 23(2002), 477-485

[7]   R. P. Wildes, Iris Recognition: an Emerging Biometric Technology. Proceeding of the IEEE, Vol. 85, Sep 1997, 1348-1363

[8]   W. W. Boles, B. Boashash, A Human Identification Technique Using Images of the Iris and Wavelet Transform. IEEE Transaction on Signal Processing, Vol. 46(4), No. 4, Apr 1998, 1185-1188

Iris Identification Using Wavelet Packet
for Images in Visible Light Illumination

Emine Krichen [1], M. Anouar Mellakh [2], Sonia Garcia-Salicetti [1],
Kamel Hamrouni [2], Nouredine Ellouze [2], and Bernadette Dorizzi [1]

[1] Institut National des Télécommunications
91011 Evry, France
{emine.krichen,sonia.salicetti,bernadette.dorizzi}@int-evry.fr
[2] ENIT : Ecole nationale d'ingénieurs de Tunis
1002 Tunis, Tunisia
{kamel.hamrouni,nouredine.ellouze}@enit.rnu.tn

Abstract. In this paper, we present a new method for iris identification
particularly convenient for visible light images. It relies on the use of packets of
wavelets [8] for the production of an code-vector. Experiments, conducted on a
database of 700 iris images, acquired with visible light illumination, show an
improvement of 1.5 % of FRR and of 12 % of FAR with the proposed method
relatively to the classical wavelet method.

1   Introduction

Iris scan is a sufficiently mature biometrics so that it can be used for identification
purposes. Indeed, Daugman's approach [1]-[,2], relies on the use of Gabor wavelets in
order to process the image at several resolution levels. An iris-code composed of
binary vectors is this way computed and a statistical matcher (logical exclusive OR
operator) analyses basically the average *Hamming* distance between two codes (bit to
bit test agreement). Some recent works follow this direction [4], [6]. Another
approach, in the framework of iris verification, introduced by Wildes [3], consists of
measuring the correlation between two images using different small windows of
several levels of resolution. Also, other methods for iris verification have been
proposed, in particular relying on ICA [6].
   In this paper, our aim is to perform iris identification on images acquired in visible
light with a standard camera, and in standard resolution and focus conditions. Indeed,
even if we knew that in such conditions the quality of images would be degraded
compared to that of near infrared images (NIR), our aim is to experiment to what
extent iris authentication can still be performed despite of this degradation of the
input images. Moreover, the color information, present in this type of images,
contrary to what occurs in NIR, can be useful to adapt the classifiers or to pre-classify
the different persons, thus improving the global quality of the system. In order to
cope with the particularities of normal light images, we have introduced some
modifications to the classical wavelet approach. In particular, we have found that the
use of Gabor wavelet packets [8] allow a real improvement on this type of images
compared to classical Gabor wavelets. Experiments have been conducted using two

different databases: one from the National Laboratory of Pattern Recognition (NLRP) in China, that is the CASIA Iris Image Database [9] collected by the Institute of Automation of the Chinese Academy of Science, in which the images were acquired with a monochrome camera using NIR illumination; the second database was recorded at Institut National des Télécommunication (INT), in which the images were acquired using a standard camera and visible light illumination.

The content of this paper is as follows. In the next section, we will describe the similarities and differences between our approach (wavelet packets) and the classical wavelet approach. Section 3 presents the results of applying both the wavelet algorithm and the packet algorithm on the different databases described above. Finally, the conclusions and perspectives are provided in section 4.

2   Recognition Method

A hybrid method is used for iris segmentation after binarization; a Hough transform [3], [4], [6] permits to localize the outer boundary of the iris, and integrodifferential operators [1], [2] are used to detect the inner boundary of the iris.

2.1 Classical Approach: Gabor Wavelets

Several variants of the approach introduced by Daugman in [1] have been introduced by other authors, but the general idea remains the same.

After iris localization, a wavelet analysis is performed. A Gabor wavelet, given in equation (1) is used for that purpose, with different width and frequency parameters in order to generate the code vector [2]. In (1), $I(\rho,\phi)$ is the representation of the image in polar coordinates, (r_0,θ_0) represents the polar coordinates of the application point of the wavelet, ω the wavelet frequency, and α and β are the multi-scale 2D wavelet size parameters.

$$\int_\rho \int_\phi e^{-i\omega(\theta_0-\phi)} e^{-(r_0-\rho)^2/\alpha^2} e^{-(\theta_0-\phi)^2/\beta^2} I(\rho,\phi)\rho \ \ d\rho \ \ d\phi \qquad (1)$$

In the following, we call "wavelet's base" a set of wavelets with various windows of analyses. A base is generally composed by one mother wavelet and N daughters wavelets, that is with N levels of resolution (in our case, three levels are consider). A daughter wavelet with resolution level N, analyzes the image with a window which is N times smaller than that of the mother wavelet.

Then, 32 self-similar wavelets, with 4 scales and 8 orientations at each scale, are generated from the mother wavelets [1], [2]. Each wavelet is convolved with 32 different spatial locations leading to a total of 1024 complex coefficients for each iris image. Each complex value is converted into 2 bits depending on the signs of its real and imaginary parts. Thus, each iris pattern is associated with a 2048 bits binary code. All the process of transforming an iris image into an code vector is illustrated in Fig. 1.
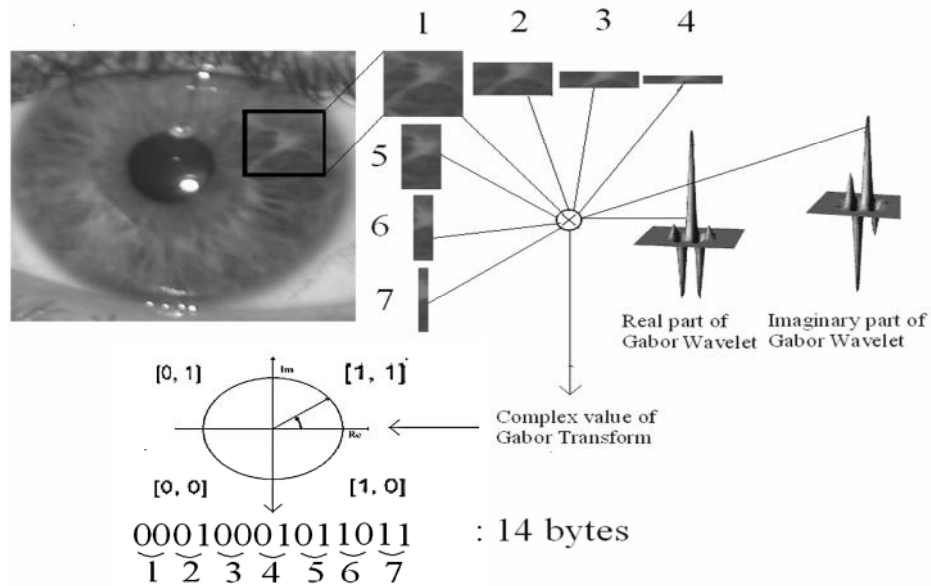
Fig. 1. The classical wavelets approach: each sub-image is considered at 4 different levels of details. At each level, a Gabor wavelet is used, leading to a complex number which is then transformed into a binary code, depending on the value of its phase

2.2 Our Method: Wavelet Packet

On one hand, the classical wavelet approach can be considered as performing more and more precise zooms on the iris. Therefore, if the initial quality of the image is not good enough, it will not be possible to perform a great number of such operations, since the quality of the "zoomed" images will get substantially degraded. A solution to this problem could be to increase the size of the initial window of the mother wavelet, but this would decrease the size of the code, thus implying a greater risk of errors at the recognition phase. For this reason, among others, we preferred to use, instead of classical wavelets, packets of wavelets; indeed, starting from the same analysis window and the same mother wavelet, the packets method involves more wavelets than the classical one, thus leading to an code vector of convenient size (see Fig.s 1 and 2). On the other hand, the whole image of the iris conveys information, even at very small scales; it is thus important to analyze the whole iris image also at low resolutions. The packets method permits to process the whole image at each level of resolution, and this is done recursively by partitioning the image in low and high frequencies (the details) (See Figure 2).

Compared to the classical wavelet analysis, the packets method also exploits equation (3), but involves several application points (r_0, θ_0) when changing of level of resolution, instead of a unique application point. In our case, for a given application point of the mother wavelet, we have 3 application points for the first level daughter
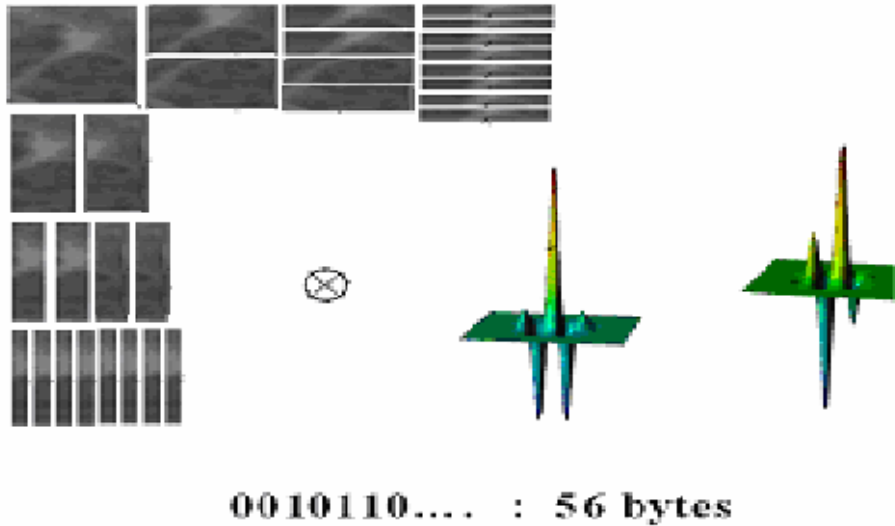
Fig. 2. Wavelet packets decomposition of the sub-images and the corresponding binary code

wavelet, 5 application points for the second level daughter wavelet, and 7 application points for the third level daughter wavelet. We used 832 wavelets with 4 scales and generated this way a 1664 bits code: in one half of the iris image, for each of the 16 levels of r, 26 levels on θ are considered ($26 * 16 * 2 = 832$).

2.3 Recognition Method

Given an iris pattern, our goal is to find the person to which this pattern could be associated. To that end, we have at disposal a database of several iris shots of each person enrolled. We consider for each person a N-template, that is a set of N references (N codes vectors).

To compare a code vector to a given N-Template, we compute a Hamming distance (XOR comparisons) between this code vector and each code vector in the N-template, and we keep the minimum of all these distances. If such distance is lower than a predefined threshold, the pattern is then considered as belonging to the corresponding person.

3   Experimental Setup

We consider a database of K persons and have at disposal L iris shots of each person enrolled. Such database is splitted in two parts, a learning database of K_{LR} persons and a test database made of the remaining persons. Our aim is to learn the decision threshold minimizing the total error rate criterion on the learning database.

For each person in the learning database, we select randomly N irises among the L available, in order to build the corresponding N-template. For the remaining irises of

such person, we compute the distance between each of them and the N-template as explained in section 3.3 (L-N intra-class distances). On the other hand, we also compute the distances between the N-template of such person and all the irises of all other persons in the learning database, considered as impostors (L*(K_{LR}-1) inter-class distances).

This process is repeated for all persons in the training database, leading at this stage to K_{LR}*L*(K_{LR}-1) inter-class distances and K_{LR}*(L-N) intra-class distances. In order to obtain an unbiaised threshold, we balanced the number of client accesses and of impostor accesses. To that end, the random selection of N irises among L to build the N-template is repeated, (L*(K_{LR}-1)/(L-N) times, leading to the same number of client and impostor accesses. To avoid introducing redundancy in the N-template choice, this number must be smaller than C_L^N. For each generated N-template, only the (L-N) intra-class distances are computed.

As for the test protocol, we consider the remaining (K- K_{LR}) persons of the test database, and follow the same scheme above described, but considering only one N-template per person.

3.1 Description of the Databases

In this work, we exploited 2 different databases. The first database (B1) has been realized in our Institute. It contains 700 images from 70 persons (70 classes), and there are between 7 and 23 iris images per person. The images have been taken with a flash set at a distance of 20-30 cms; the camera used has a 23mm focus, and the captured irises have 120-150 pixels of radius. Also, some variations in lighting and position have been introduced in the database. We have chosen to capture only the left iris of each person. As the use of flash introduces a reflection in the right part of the iris, we only exploit the left part of the iris images; this divides of course the code vector by a factor 2 (from 2048 bits to 1024 bits for the classical method, and from 1664 bits to 832 bits for the wavelet packets method).

The second database comes from the National Laboratory of Pattern Recognition (NLRP) in China, that is the CASIA Iris Image Database [9] collected by the Institute of Automation of the Chinese Academy of Science, in which the images were acquired with a monochrome camera using NIR illumination. It is composed of 749 images from 107 different irises. For each iris, 7 images were captured in 2 sessions.

3.2 Results

In our experiments, database B1 was splitted in two parts: 35 people were used to determine the adequate classification threshold (threshold learning base). The remaining persons were used to test the identification system. Ten irises are available per person, and 5 are used to build the N-template. Testing the system thus involves computing 35*5 intra-class distances and 35*34*10 inter-class distances.

Following the same idea, database B2 was also splitted in two parts: 30 persons were used in the learning phase, and 77 persons were used to test the system.

Table 1. Comparison of the different algorithms on database B1 and B2

Databases	B1		B2	
Type of errors	FAR	FRR	FAR	FRR
Classical wavelet method	12.04%	2.%	0.35%	2.08%
Packets method	0%	0.57%	0.2%	1.38%

Results are given in terms of false acceptance rate (FAR) and false rejection rate (FRR). We consider that a false acceptance error was made by the system when an code vector and a N-template corresponding to two different persons lead to a Hamming distance lower than the threshold. In the same way, we consider that a false rejection error was made by the system when a code vector and a N-template corresponding to the same person lead to a Hamming distance higher than the threshold.

We notice (table 1) that on database B1, that is in visible light, the classical wavelet approach performs poorly, as also remarked by other works [7], [10]. Indeed, in most cases, the classical method leads to false acceptances when the compared original iris images are blurred and have the same size. In this case, the packets method is more robust because it does not analyse only high frequencies. Also, in general, the classical method leads to false rejections when the eyelid covers partially the iris; in this case, the packets method is also more robust, since the original analysis window is bigger.

In NIR illumination (database B2), we see that the two methods are equivalent. These results do not reach state-of-the-art error rates because no enhancement was done on purpose, and no filter was applied on the code vector as well [2]. We notice that there are still some false rejections with both methods; this is due to the fact that there are some irises with very low contrast between the iris and the sclera, leading this way to segmentation failures.
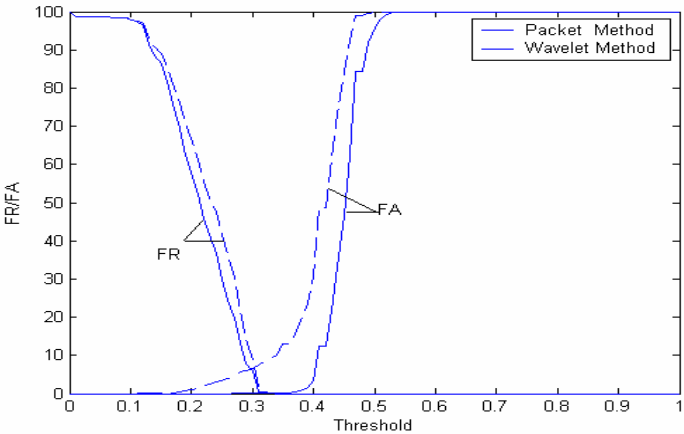
Fig. 3. Variations of the FAR and FRR in function of a threshold for the two different methods applied on the database B1 (Visible light illumination)

4   Conclusion

In this paper, our interest was the processing of iris images acquired in visible light conditions. Very few researchers work in that context as most acquisition devices function in near infrared illumination. We have verified, as other authors [7], [10] that in visible light illumination, the use of the standard wavelet approach is not satisfactory in the context of blurred images and poor iris texture. In that normal light context, we have proposed the use of wavelet packets instead of standard wavelets. Experiments lead to good performances on a database of 70 persons, considering that no enhancement was done on purpose, and no filter was applied on the code vector as well [2]. Relatively good results were also obtained on near infrared images.

In the future, we will introduce image enhancement before the pattern matching phase, which should improve results. Also, we are interested in working on a larger database containing both types of iris images (in visible light and in NIR illumination) as well as studying the limits of the system regarding to image quality degradation.

References

[1]   John Daugman : High Confidence Recognition of Persons by Rapid Video Analysis of Iris Texture. European Convention on Security and Detection (1995) 244 -251
[2]   John Daugman : How Iris Recognition Works. Proceedings of the International Conference on Image Processing, Volume 1 (2002) 33-36, 22-25
[3]   R.P. Wildes : Iris Recognition: an Emerging Biometric Technology Proceedings of the IEEE Volume 85 Issue 9 (1997) 1348 -1363
[4]   Li Ma, Yunhong Wang, Tieniu Tan : Iris Recognition Using Circular Symmetric Filters Proceedings of the 16th International Conference on Pattern Recognition Volume 2 (2002) 414 -417
[5]   Lye Wil Liam, A. Chekima, Liau Chung Fan, J.A. Dargham : Iris Recognition Using Self-Organizing Neural Network Student Conference on Research and Development SCOReD (2002) 169 -172
[6]   Ya-Ping Huang, Si-Wei Luo, En- Yi Chen, "An efficient iris recognition system", Proceedings of the First Conference on Machine Learning and Cybernetics, Beijing, 4-5 November 2002.
[7]   B.V.K Vijaya Kumar, Chunyan Xie, Jason Thornton Iris Verification Using Correlation Filters J. Kittler and M.S. Nixon (Eds.) AVBPA LNCS 2688 (2003) 697-705
[8]   Andrew Laine, Jian Fan : Texture Classification by Wavelet Packet Signatures IEEE Transaction on Pattern Analysis and Machine Intelligence Vol. 15 No. 11 (1993)
[9]   http://www.sinobiometrics.com
[10]   Chul-Hyun Park, Joon-Jae Lee, Mark J.T. Smith, Kil-Houm Park Iris-based Personal Authentication Using Normalized Directional Energy Features AVBPA LNCS 2684 (2003) 224-232

Target Dependent
Score Normalization Techniques
and Their Application to Signature Verification

J. Fierrez-Aguilar, J. Ortega-Garcia, and J. Gonzalez-Rodriguez

Biometrics Research Lab., ATVS
Universidad Politecnica de Madrid, Spain
{jfierrez,jortega,jgonzalez}@diac.upm.es

Abstract. Score normalization methods in biometric verification, which encompass the more traditional user-dependent decision thresholding techniques, are reviewed from a test hypotheses point of view. These are classified into test dependent and target dependent methods. The focus of the paper is on target dependent methods, which are further classified into impostor-centric, target-centric and target-impostor. These are applied to an on-line signature verification system on signature data from SVC 2004. In particular, a target-centric technique based on a variant of the cross-validation procedure provides the best relative performance improvement both for skilled (19%) and random forgeries (53%) as compared to the raw verification performance without score normalization (7.14% EER and 1.06% EER for skilled and random forgeries respectively).

1 Introduction

Previous studies have shown that the performance of a number of biometric verification systems, specially those based on behavioral traits such as written signature [1] or voice [2], can be improved by using user-dependent decision thresholds. Even greater verification performance improvement can be expected through the use of score normalization techniques [3]. These methods (which include the user-dependent thresholding as a particular case) account not only for user specificities but also for intersession and environment changes [4]. The system model of biometric authentication with score normalization is depicted in Fig. 1.

The objectives of this work are: i) to provide a framework for score normalization collecting previous work in related areas, ii) to provide some guidelines for the application of these techniques in real world scenarios, and iii) to provide an example of a successful application of the proposed normalization methods regarding the first international Signature Verification Competition (SVC 2004).
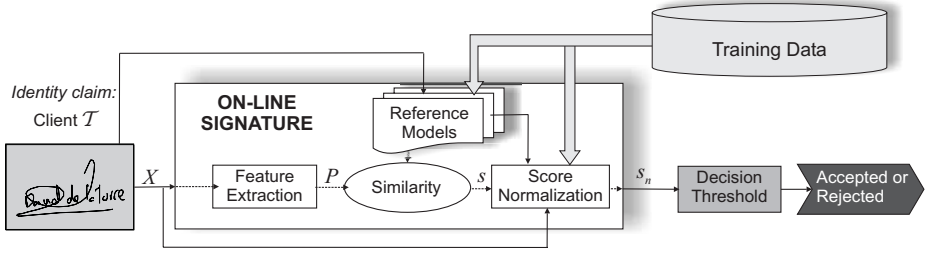
Fig. 1. System model of biometric authentication with score normalization

2 Theoretical Background

2.1 Score Normalization

Given a test sample X the problem of biometric authentication can be stated as a basic hypotheses test between two hypotheses:

$H0$: X is from hypothesized client $\mathcal{T}$.
$H1$: X is *not* from hypothesized client $\mathcal{T}$.

The optimum test to decide between these two hypotheses is a likelihood ratio test given by

$$\frac{p(X|H0)}{p(X|H1)} \begin{cases} > \theta \text{ Accept } H0 \\ < \theta \text{ Accept } H1 \end{cases} \tag{1}$$

where $p(X|H0)$ and $p(X|H1)$ are respectively the probability density functions for the hypotheses $H0$ and $H1$ evaluated for the observed biometric sample X. The decision threshold for accepting or rejecting $H0$ is θ. An equivalent log-likelihood ratio test is obtained transforming (1) into the log domain

$$\log p(X|H0) - \log p(X|H1) \begin{cases} > \log \theta \text{ Accept } H0 \\ < \log \theta \text{ Accept } H1 \end{cases} \tag{2}$$

A common practice in biometric verification (e.g., GMM in case of speaker recognition [5], HMM in case of signature recognition [6], etc.) consists in characterizing each client $\mathcal{T}$ by a statistical model $\lambda^{\mathcal{T}}$ (i.e., the reference model in Fig. 1). In this case, the similarity s is computed as

$$s = \log p(X|\lambda^{\mathcal{T}}) \tag{3}$$

which is an estimation of $\log p(X|H0)$. As a result, the optimal score normalization method for an authentication system based on statistical modeling is given by

$$s_n = s - \log p(X|H1) \tag{4}$$

Worth noting, the normalizing term is affected, in general, by:

Input Information: the input biometric sample X.

Information from Clients: example scores $s_1^{\mathcal{T}}, \ldots, s_{N_{\mathcal{T}}}^{\mathcal{T}}$ from the hypothesized target $\mathcal{T}$.

Information from Impostors: both the models $\left\{ \lambda_1^{\overline{\mathcal{T}}}, \ldots, \lambda_{N_I}^{\overline{\mathcal{T}}} \right\}$ and example scores $\left\{ s_1^{\overline{\mathcal{T}}}, \ldots, s_{N_{\overline{\mathcal{T}}}}^{\overline{\mathcal{T}}} \right\}$ from N_I possible impostors pretending the hypothesized client $\mathcal{T}$.

Estimation of $\log p(X|H1)$ based on the different information involved is nevertheless not a straightforward task. Thus, operational procedures are usually employed. Much effort has been done in order to derive such operational procedures for *score normalization* (also *likelihood normalization*) based on the statistical formalism described above, mainly in the speaker recognition community [3]. These operational procedures aim at designing a function

$$s_n = f(s, X, \left\{ s_1^{\mathcal{T}}, \ldots, s_{N_{\mathcal{T}}}^{\mathcal{T}} \right\}, \left\{ \lambda_1^{\overline{\mathcal{T}}}, \ldots, \lambda_{N_I}^{\overline{\mathcal{T}}} \right\}, \left\{ s_1^{\overline{\mathcal{T}}}, \ldots, s_{N_{\overline{\mathcal{T}}}}^{\overline{\mathcal{T}}} \right\}) \qquad (5)$$

so as to minimize the error rate of the verification task. Linear functions of various statistics of the information involved in Eq. (5) is the prevailing strategy. This is the case of the celebrated: *i*) z-norm, which considers only scores samples from impostors, *ii*) t-norm, based on the test sample and models from impostors, and *iii*) UBM-norm, which considers the test sample and a universal background model characterizing the average target. Other examples can also be found regarding face [7] or signature recognition [8].

In order to simplify the discussion yet providing a powerful framework for score alignment, the main focus of this paper is on considering neither input test information nor models from impostors, i.e.

$$s_n = f(s, \left\{ s_1^{\mathcal{T}}, \ldots, s_{N_{\mathcal{T}}}^{\mathcal{T}} \right\}, \left\{ s_1^{\overline{\mathcal{T}}}, \ldots, s_{N_{\overline{\mathcal{T}}}}^{\overline{\mathcal{T}}} \right\}) \qquad (6)$$

This family of score normalization methods will be referred to as *target dependent score normalization techniques*. Other normalization methods using the test sample and models from impostors will be referred to as *test dependent normalization techniques*.

3   Target Dependent Score Normalization Techniques

3.1   Impostor-Centric Methods

In impostor-centric methods (*IC* for short) no information about client score intra-variability is used. Therefore

$$s_{IC} = f(s, \mathcal{I} = \left\{ s_1^{\overline{\mathcal{T}}}, \ldots, s_{N_{\overline{\mathcal{T}}}}^{\overline{\mathcal{T}}} \right\}) \qquad (7)$$

The following *IC* methods are considered in this work:

IC-1: $s_{IC\text{-}1} = s - \mu_{\mathcal{I}}$
IC-2: $s_{IC\text{-}2} = s - (\mu_{\mathcal{I}} + \sigma_{\mathcal{I}})$
IC-3: $s_{IC\text{-}3} = (s - \mu_{\mathcal{I}})/\sigma_{\mathcal{I}}$

where $\mu_{\mathcal{I}}$ and $\sigma_{\mathcal{I}}$ are respectively the mean and standard deviation of the impostor scores $\mathcal{I}$. Note that the impostor samples scores $\mathcal{I}$ can be, in general, either from casual impostors (cIC) or from real impostors (rIC).

3.2   Target-Centric Methods

In target-centric methods (TC for short) no information about impostor score variability is used. Therefore

$$s_{TC} = f(s, \mathcal{C} = \left\{ s_1^{\mathcal{T}}, \ldots, s_{N_{\mathcal{T}}}^{\mathcal{T}} \right\}) \tag{8}$$

Similarly to the impostor-centric case, the following methods are obtained

TC-1: $s_{TC\text{-}1} = s - \mu_{\mathcal{C}}$
TC-2: $s_{TC\text{-}2} = s - (\mu_{\mathcal{C}} - \sigma_{\mathcal{C}})$
TC-3: $s_{TC\text{-}3} = (s - \mu_{\mathcal{C}})/\sigma_{\mathcal{C}}$

Client scores $\mathcal{C}$ should be obtained from the available training set. In this work, we propose to generate $\mathcal{C}$ by using either the resubstitution or the rotation sampling methods of error estimation [9].

3.3   Target-Impostor Methods

In target-impostor methods (TI for short) information from both client score intra-variability and impostor score variability is used. Therefore

$$s_{TI} = f(s, \mathcal{C} = \left\{ s_1^{\mathcal{T}}, \ldots, s_{N_{\mathcal{T}}}^{\mathcal{T}} \right\}, \mathcal{I} = \left\{ s_1^{\overline{\mathcal{T}}}, \ldots, s_{N_{\overline{\mathcal{T}}}}^{\overline{\mathcal{T}}} \right\}) \tag{9}$$

The two following methods are considered

TI-1: $s_{TI\text{-}1} = s - s_{\text{EER}}(\mathcal{I}, \mathcal{C})$
TI-2: $s_{TI\text{-}2} = s - (\mu_{\mathcal{I}}\sigma_{\mathcal{C}} + \mu_{\mathcal{C}}\sigma_{\mathcal{I}})/(\sigma_{\mathcal{I}} + \sigma_{\mathcal{C}})$

where $s_{\text{EER}}(\mathcal{I}, \mathcal{C})$ is the decision threshold at the empirical Equal Error Rate obtained from $\mathcal{I}$ and $\mathcal{C}$.

4   Experiments

For the experiments reported in this paper, the HMM-based on-line signature verification system from Universidad Politecnica de Madrid [6, 8] competing in the First Intl. Signature Verification Competition (SVC 2004)[1] has been used.

---

[1] http://www.cs.ust.hk/svc2004/

Development corpus of the extended task (including coordinate and timing information, pen orientation and pressure) from SVC 2004 has been used. It consists of 40 sets of signatures. Each set contains 20 genuine signatures from one contributor (acquired in two separate sessions) and 20 skilled forgeries from five other contributors.

Signature data from both sessions is used both for training and testing. Training data consists of 5 genuine signatures for each target. For a specific target user, casual impostor information is extracted from all the remaining targets. Results in which real impostor information is used for computing the normalization functions are also provided. Impostor data for the estimation of the normalization parameters, either casual or real, is used in a leave-one-out fashion, i.e., testing one impostor with a normalization scheme estimated with information from the remaining impostors and rotating the scheme.

A priori score normalization methods are compared in the following. This means that only the information from the training set is used both for the enrollment of the targets and for the estimation of the parameters of the normalization scheme. In order to have an indication of the level of performance with an ideal score alignment between targets, the *a posteriori* target dependent score normalization TI-1 is also given. Only in this case test information is also used for the computation of the normalization functions.

4.1   Results

In Fig. 2 (a) the different impostor-centric methods described in Sect. 3.1 are compared either for skilled (left) or random forgeries (right). Raw verification performance with no normalization (7.14% and 1.06 EER for skilled and random forgeries respectively) is significantly improved by the *a posteriori* normalization scheme (2.79% and 0.01% respectively). Regarding the skilled forgeries test, *a priori* method IC-3 outperforms IC-1 and IC-2. Raw performance is only improved in this case by considering statistics from real impostors. Regarding the random forgeries test, significant improvements are obtained considering statistics either from casual or from real impostors.

Results of different resampling techniques for the estimation of target variability are summarized in Fig. 2 (b) for three different verification systems of decreasing verification performance (from left to right). As it can be observed, the rotation scheme always leads to verification improvements whereas the resubstitution strategy only leads to improvements in the low performance system. This result penalizes the biased estimation provided by the resubstitution scheme in favor of the unbiased rotation procedure.

Verification performance for the target-impostor methods is shown in Fig. 2 (c). As in the impostor-centric experiment, only target-impostor normalization schemes based on real impostor statistics improve verification performance with respect to no score normalization in case of tests with skilled forgeries. With regard to the test considering random forgeries, verification performance improvements are obtained considering either casual impostor or real impostor statistics.
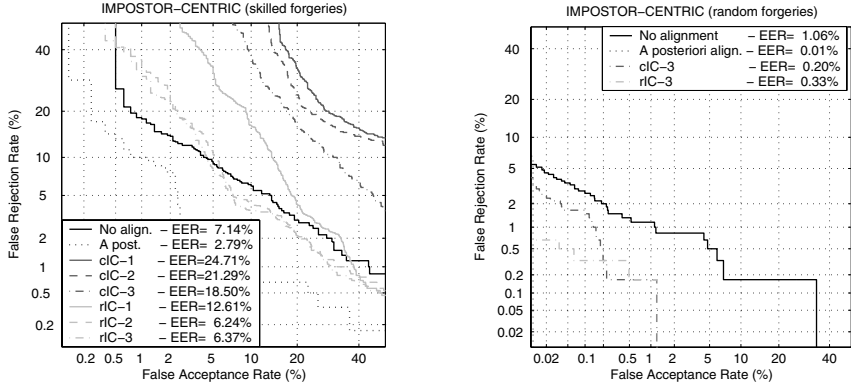
(a) Impostor-Centric: Different impostor-variability estimation methods.
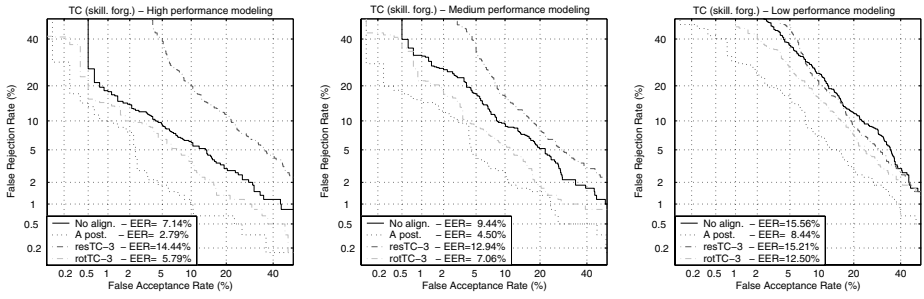
(b) Target-Centric: Different client-variability estimation methods.
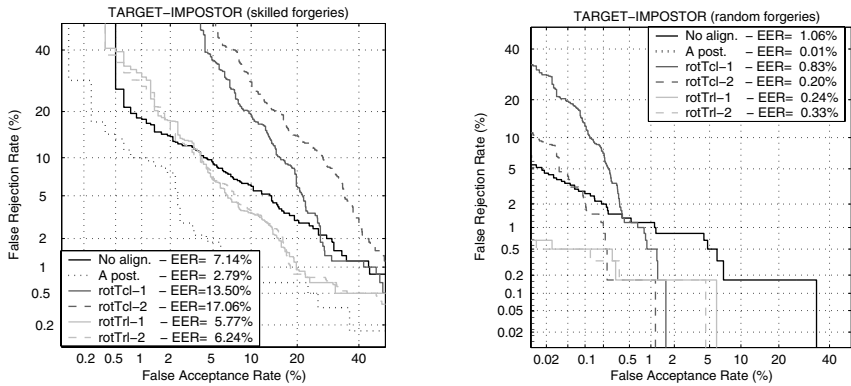
(c) Target-Impostor: casual/real information for impostor-variability estimation.

Fig. 2. Comparison of target dependent score normalization techniques

5   Conclusions

Target dependent score normalization techniques have been reviewed and applied to a HMM on-line signature verification system (7.14% EER and 1.06% EER for skilled and random forgeries respectively) on SVC 2004 signature data and various experimental finding have been obtained. Most remarkably, target-centric techniques based on a variation of the cross-validation procedure provided the best performance improvement both for skilled (5.79% EER) and random forgeries (0.50% EER). Other worth noting experimental findings are: *i*) the use of casual impostor statistics in either impostor-centric or target-impostor methods leads to the highest performance improvement when testing with random forgeries but lowers verification performance in case of testing against skilled forgeries, *ii*) the use of real impostor statistics in either impostor-centric or target-impostor methods leads to verification performance improvements when testing either with random or skilled forgeries, and *iii*) statistics for the estimation of target score intra-variability should be unbiased.

Acknowledgements

References

[1] Jain, A., Griess, F., Connell, S.: On-line signature verification. Pattern Recognition 35 (2002) 2963–2972 498
[2] Matsui, T., Nishitani, T., Furui, S.: Robust methods of updating model and a priori threshold in speaker verification. In: Proc. of ICASSP. (1996) 97–100 498
[3] Auckenthaler, R., Carey, M., Lloyd-Tomas, H.: Score normalization for text-independent speaker verification systems. Digital Signal Processing **10** (2000) 42–54 498, 500
[4] Navratil, J., Ramaswamy, G.: The awe and mystery of t-norm. In: Proc. of ESCA Eur. Conf. on Speech Comm. and Tech., EuroSpeech. (2003) 2009–2012 498
[5] Reynolds, D.A., Quatieri, T.F., Dunn, R.B.: Speaker verification using adapted gaussian mixture models. Digital Signal Processing **10** (2000) 19–41 499
[6] Fierrez-Aguilar, J., Ortega-Garcia, J., Gonzalez-Rodriguez, J.: A function-based on-line signature verification system exploiting statistical signal modeling. Intl. Journal on Image and Graphics (2004) (accepted) 499, 501
[7] Sanderson, C., Paliwal, K.K.: Likelihood normalization for face authentication in variable recording conditions. In: Proc. of ICIP. Volume 1. (2002) 301–304 500
[8] Ortega-Garcia, J., Fierrez-Aguilar, J., Martin-Rello, J., Gonzalez-Rodriguez, J.: Complete signal modeling and score normalization for function-based dynamic signature verification. In: Proc. of IAPR Intl. Conf. on Audio- and Video-based Person Authentication, AVBPA, Springer (2003) 658–667 500, 501
[9] Jain, A., Duin, R., Mao, J.: Statistical pattern recognition: A review. IEEE Trans. on Pattern Anal. and Machine Intell. 22 (2000) 4–37 501

Skilled Forgery Detection in On-Line Signatures: A Multimodal Approach

Anoop M. Namboodiri, Shailesh Saini, Xiaoguang Lu, and Anil K. Jain

Department of Computer Science and Engineering
Michigan State University
East Lansing, MI - 48824, USA
{anoop,shailesh,lvxiaogu,jain}@cse.msu.edu

Abstract. Signature recognition has a long history of usage in authentication of transactions and legal contracts and hence is easily accepted by users in a variety of applications. However, the problem of skilled forgeries is an important challenge that needs to be overcome before signature recognition systems will become viable in unsupervised authentication systems. In this paper, we present a multimodal approach to forgery detection, where a physiological trait, the face of the signing person, is used to validate the signature. Methods of normalizing and combining the matching scores from the individual modalities are investigated. Test results of the system on a database of 100 users is presented. The system achieves an equal error rate of 2.2% in the presence of high quality skilled forgeries and could detect all the skilled forgeries at a genuine acceptance rate of 75%.

1 Introduction

The signature of a person is a well-known biometric attribute. Other biometric attributes, which are commonly used for authentication include iris, hand geometry, face and fingerprints (see Jain et al. [1]). While attributes like iris and fingerprints do not change over time and thus have very small intra-class variation, they require special and relatively expensive hardware to capture the biometric image. An important advantage of signature over other biometric attributes is that it has been traditionally used in authenticating documents and enforcing binding contracts in paper-based documents and hence is socially accepted. However, unlike the physiological biometric characteristics such as fingerprints or iris patterns, signature is an acquired or behavioral trait and could be closely mimicked by an expert forger. The ability to detect such skilled forgeries is extremely important to the success of applications, which use signatures as their primary verification method.

There are two main approaches that are currently used to detect forgeries in signatures. In the case of applications, which do not allow human supervision at the time of signing (e.g., remote verification over a network), we capture additional information about the signature, which is difficult to forge. Signature

verification systems which use velocity, pressure and tilt of the pen for verification [2] belong to this category. In traditional signature verification applications, an additional level of security, such as an identity card, is used to verify the identity of the signee. However, such an approach requires human supervision during the signing process.

In this paper, we present a system, which captures the face image of the user while he is signing to supplement the signature verification process. The combination of face and signature of a user has various advantages. The use of a physiological biometric characteristic makes it difficult to forge the signatures, as the expert forger has to match the physiological traits of a user in addition to his signature. The face image can be captured passively and hence does not impose any additional burden on a genuine user. In addition, one can control the lighting, background and pose of the face [3] in situations such as ATM transactions, since the user will be positioned in a restricted space, and looking at the signing tablet to provide the signature.

There are a variety of techniques available to do signature recognition [4, 5, 6] and face recognition [7, 8, 9]. In this paper, we use two of the commonly used methods for matching signatures and faces, namely dynamic time warping based signature matching and LDA-based face matching. We focus on the problem of combining the two modalities for the purpose of forgery detection [10]. The most commonly used method for fusion in multi-biometric systems is the combination of matching scores produced by the different modalities [11].

2   Signature and Face Recognition

The signature is collected using a tablet, which is equipped with a touch or position sensor. The on-line signature captures the dynamics of signing in addition to the spatial information, which makes it difficult for a person to forge a signature. The signature is recorded as a sequence of points along the trajectory of the pen. A number of features are computed at each sample point in the curve. The feature vector at a point includes the distances in the horizontal and vertical directions to the next sample point and the cosine of the angle formed by the lines connecting the point to its neighbors. The signature is thus represented as a sequence of feature vectors: $S_1 = < f_1, f_2, \cdots, f_n >$, where f_i is the three-dimensional feature vector computed at sample point i. A DTW-based algorithm [12] computes the optimal alignment between the two signatures. The resulting distance measure, falls in the range $[0, \infty)$.

We use the (LDA)-based face classifier proposed by Belhumer et al. [8]. A two-dimensional face image is considered as a vector, by concatenating each row (or column) of the image. Let $X = (x_1, x_2, \cdots, x_i, \cdots, x_N)$ denote the data matrix, where N is the size of the training set. Each x_i is a face vector of dimension n, where n is the total number of pixels in the face image. The Linear Discriminant Analysis (LDA) representation is a linear transformation from the original image vector to a projection feature vector, i.e. $Y = W_{LDA}^T X$, where Y is the $d \times N$

feature vector matrix, d is the dimension of the feature vector $(d \ll n)$ and W_{LDA} is the transformation matrix, derived by

$$W_{LDA} = argmax_W \frac{W^T S_B W}{W^T S_W W}, \tag{1}$$

where S_B is the between-class scatter matrix and S_W is the within-class scatter matrix,

$$S_B = \sum_{i=1}^{c} N_i (x_i - m)(x_i - m)^T; \quad S_W = \sum_{i=1}^{c} \sum_{x_k \in X_i = 1} (x_k - m_i)(x_k - m_i)^T \tag{2}$$

In the above expression, N_i is the number of training samples in class i, c is the number of distinct classes, m is the mean vector of all the samples, m_i is the mean vector of samples belonging to class i and X_i represents the set of samples belonging to class i. LDA derives a low dimensional representation from the high dimensional space, which used as the feature vector. The matching score between two face images is calculated as the cosine value of the angle between their feature vectors. A larger matching score means a better match. The matching scores fall in the range $[-1.0, 1.0]$, which is then scaled and shifted to the range $[0, 1]$.

3   Combining Face and Signature

We combine the individual modalities using the matching scores, which offers more information to the combination process than a binary output decision of match/mismatch. In addition, it allows one to use algorithms, which are appropriate for the individual modalities, as long as they produce a similarity or distance score. Let $\lambda_1, \lambda_2, \cdots, \lambda_n$ be the matching scores computed from individual biometrics. The combined score Λ is computed as a function of the individual matching score, $\Lambda = F(\lambda_1, \lambda_2, \cdots, \lambda_n)$. The most commonly used technique is called the *sum rule*, which computes a weighted sum of the individual matching scores,

$$\Lambda = w_1 \lambda_1 + w_2 \lambda_2 + \cdots + w_n \lambda_n, \tag{3}$$

where w_i are the weights of the individual modalities. We usually impose the constraint $w_1 + w_2 + \cdots + w_n = 1.0$ to normalize the sum. Alternative approaches include the *min* rule, the *max* rule and the *product* rule, which compute the combined score as the minimum, maximum, or product of the individual matching scores. One might also treat the set of matching values as a feature vector and use classifiers such as KNN, decision tress, neural networks [13] etc. to arrive at a decision. In our experiments, the sum rule gave the best results for combining the scores of the signature and face verification algorithms.

Before combining the individual scores using any of the combination techniques, it is desirable to normalize the scores to a common range. Our signature recognition algorithm outputs a distance measure, d_1, which falls in the range
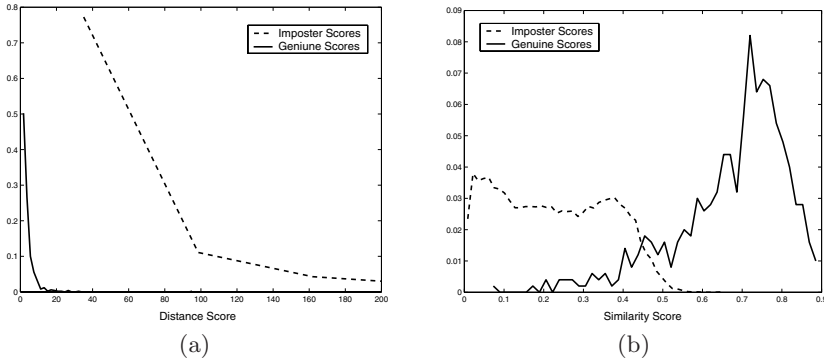
Fig. 1. Distributions of (a) distance scores and (b) the corresponding similarity distribution from the signature matching algorithm

$[0, \infty)$. We transform the distance score to a similarity score, λ_1 in the range $(0, 1]$ using the function:

$$\lambda_1 = \frac{1}{1 + \alpha.d_1}, \tag{4}$$

where α controls the rate of decay of similarity score as the distance score increases. The value of α was determined experimentally and set to 0.15. However, the result of the combination is not very sensitive to the value of $\alpha \in [0.05, 0.45]$. Figure 1 shows the distribution of the distance scores computed by the signature matching algorithm, and the corresponding similarity score distribution.

The combination of the similarity scores, using the sum rule, requires additional parameters, the weights of individual biometrics (see equation 3). The choice of these parameters determine the significance of the individual matching algorithms, which contribute to the combination. Let w_1 be the weight of the signature score. The weight of the face score is set to $1 - w_1$. Since the objective of using multiple modalities in this work is to detect the skilled forgeries of signature, we assign higher importance to the face matching algorithm, compared to the signature matcher.

4   Experimental Results

The database consisted of signatures from 100 users collected in 2 sessions, over a period of 3 weeks. Each person signed his name 5 times in each session, generating a total of 10 samples per person. The signatures are sampled by the signing tablet at a rate of 100 samples per second. We divided the database of signatures, by randomly choosing three of the signatures for enrollment and the remaining (7) signatures for testing. The face database consisted of 120 users, each user providing 10 samples of his face. The faces were captured in different lighting conditions. We selected 100 users from this set and randomly paired
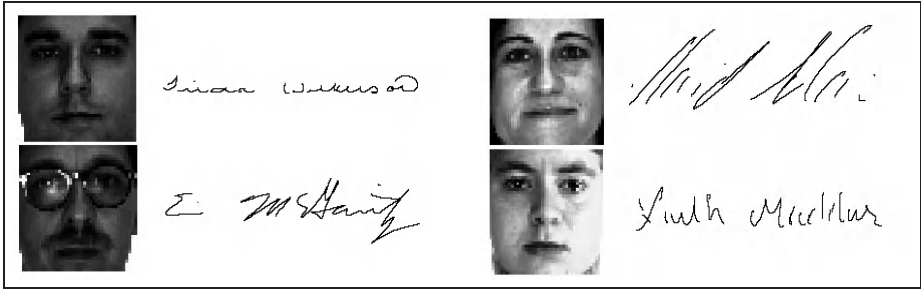
Fig. 2. Examples of face-signature pairs from the database

them with the users in the signature database. The samples from the remaining 20 users were used to pair up the skilled forgeries described below. From the 10 face samples, we use 5 samples for training and 5 samples for testing. Figure 2 shows a few face-signature pairs from the database used in this work.

Forgeries in signature are divided into skilled and random forgeries, depending on the information available to the forger about the signature, which he is trying to imitate. Random forgeries assume no knowledge of the target signature. These are usually used to compute the probability of a genuine user being identified incorrectly, while signing his or her own name. We use the genuine signatures of each user as random forgeries for the rest of the users. In the case of skilled forgeries, we assume that the forger has complete information about the target signature, and has practiced and perfected the signing process. One of the main challenges in developing a system for detecting skilled forgeries of signatures, is the lack of availability of skilled forgers to train and test the system. We overcome this difficulty by assigning a portion of the genuine signatures from a user as skilled forgeries. The assumption is that a skilled forger can at best be as good as the genuine user in signing. In practice, the forgeries will generate lower similarity scores than the ones selected by this process, and hence the results of the system in practice will be better than what is reported here. We label two of the test signatures from each user as forgeries, generating 200 skilled forgeries. Thus each user has 5 genuine signatures, 2 skilled forgeries and 495 random forgeries in the database.

The parameters of the signature matching algorithm were estimated from an independent set of training samples. An estimate of w_1, the weight assigned to signature, was computed from the training (enrollment) samples of signatures and faces and was set to 0.1. The value of w_1 depends on the proportion of skilled forgeries that are present in the system. The database described above contains 2 skilled forgeries and 5 genuine signatures per person, which is a very high ratio of skilled forgeries for a practical system. Hence the performance of a practical system will be better than what is reported here.

Assigning a lower weight to the signature modality (w_1) reduces the effect of skilled forgeries on the system. Figure 3 shows the ROC curve of the system in the presence of random forgeries and both random and skilled forgeries. The

(a)   (b)
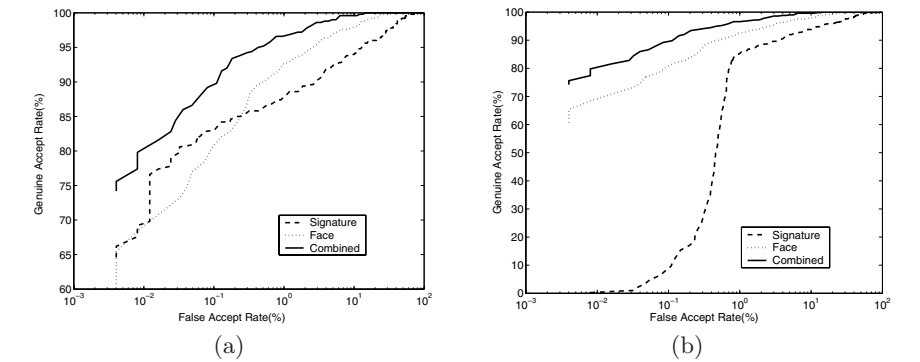
Fig. 3. ROC curves of the combination (a) in the presence of random forgeries and (b) in the presence of random and skilled forgeries

(a)   (b)
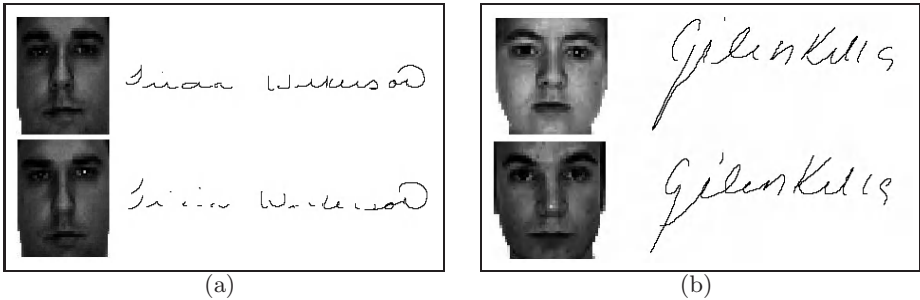
Fig. 4. Improvement in matching performance. (a) shows a rejected genuine signature, which was accepted when combined with the face matcher. (b) shows a skilled forgery, which was detected when paired up with the face image of the imposter

equal error rate of the combined system in the presence of random and skilled forgeries is 2.2% and in the presence of only random forgeries is 2.1%. We achieve a genuine acceptance rate of 75% with no false acceptance for the combined system. However, in practice one might want to operate at a higher false acceptance rate to achieve a more user friendly genuine acceptance rate. Figure 4 (a) shows a genuine signature and its template along with the corresponding face images. The user was not accepted by the signature matcher alone, at a FAR of 2.0%. However, the multimodal system was able to accept the user at the same FAR. Figure 4 (b) shows a genuine signature and its skilled forgery, which was detected by the multimodal system.

5   Conclusions and Future Work

We have presented a biometric authentication system, which can be deployed cheaply and effectively in many practical situations such as ATMs, checkout counters in stores and access control to enhance security. Information from signatures effectively complements a physiological trait such a face, and the combined system performs considerable better than the individual biometric modalities. The system can also limit the effect of skilled forgeries with a weighted combination of the matching scores from the individual modalities. The results presented are overly pessimistic due to the quality and quantity of skilled forgeries in the database. The system presented here assumes the use of an inexpensive touch-screen sensor to capture the signature. However, in cases where a more sophisticated sensor is used, one can capture additional information of the signature such as pen pressure and tilt. We are currently investigating the effects of using additional information from the signature sensor on the performance of dace and signature multimodal system.

References

[1] Jain, A. K., Pankanti, S., Bolle, R., eds.: BIOMETRICS: Personal Identification in Networked Society. Kluwer (1999)   505
[2] Ohishi, T., Komiya, Y., Matsumoto, T.: On-line signature verification using pen-position, pressure and inclination trajectories. In: Proc. ICPR. (2000) 547–550   506
[3] Adini, Y., Moses, Y., Ullman, S.: Face recognition: The problem of compensating for changes in illumination direction. IEEE Trans. on PAMI **19** (1997) 721–732   506
[4] Gupta, G., McCabe, A.: A review of dynamic handwritten signature verification. `http://cay.cs.ju.edu.com/~alan/Work/HSV-Lit_rev.html` (1997)   506
[5] Dolfing, J. G. A., Aarts, E. H. L., van Oosterhout, J. J. G. M.: On-line signature verification with hidden markov models. In: Proc. ICPR. Volume 2. (1998) 1309–1312   506
[6] Huang, K., Yan, H.: On-line signature verification based on dynamic segmentation and global and local matching. Optical Engineering **34** (1995) 3480–3487   506
[7] Zhao, W., Chellappa, R., Phillips, P., Rosenfeld, A.: Face recognition: A literature survey. ACM Computing Surveys **35** (2003) 399–458   506
[8] Belhumeur, P. J., Hespanha, J. P., J., K. D.: Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. IEEE Trans. on PAMI **19** (1997) 711–720   506
[9] Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience **3** (1991) 71–86   506
[10] Kittler, J., Hatef, M., Duin, R. P., Matas, J. G.: On combining classifiers. IEEE Trans. on PAMI **20** (1998) 226–239   506
[11] Ross, A., Jain, A. K.: Information fusion in biometrics. Pattern Recognition Letters **24** (2003) 2115–2125   506
[12] Jain, A. K., Griess, F. D., Connell, S. D.: On-line signature verification. Pattern Recognition **35** (2002) 2963–2972   506
[13] Lee, D. S., Srihari, S.: A theory of classifier combination: the neural network approach. In: Proceedings of the 3rd ICDAR. (1995) 42–46   507

Writer Identification Using Dynamic Features

Kun Yu, Yunhong Wang, and Tieniu Tan

Centre for Biometric Authentication and Testing
NLPR, Institute of Automation, Chinese Academy of Sciences
P. O. Box 2728, Beijing, 100080, P. R. China
{kyu,wangyh,tnt}@nlpr.ia.ac.cn

Abstract. Automatic handwriting identification is an interesting research topic in the community of biometrics. In this paper, we propose a new approach to writer identification based on dynamic features, namely the features associated with gestures and movements of the writers during writing. The feature vectors defined here are mainly extracted from the writing pressure, velocity and pen inclination, whose components are selected according to respective reliability. Experimental results demonstrate that the proposed algorithm performs well, especially in tackling the degraded documents with little loss of accuracy.

1   Introduction

As one of the most traditional methods for personal identification, handwriting has been in extensive research [1][2][3][4]. Due to the diversity of circumstances of writing and various writing styles of different writers, handwriting identification is regarded as an art in most cases. So far, the most active research domains include text-dependent document analysis, text-dependent document authentication, etc.

Many handwriting identification systems require the writer to write documents of indicated contents [1][2][5], and the same ones are written for both classifier training and writer identification. To weaken this constraint, Yasushi proposed an eclectic scheme [6], which was text-dependent in nature.

Text-independent handwriting identification has an apparent advantage over text-dependent cases in practical applications. Specifically, people can write text of different contents at will, where great difficulty exists in reproducing handwriting of the indicated style. Presently researchers adopt global and local features to discriminate the handwriting. Extracting global features for writer authentication have been proved to be feasible [7], but a relatively large character set is necessary. Other researchers focused on local information, and the hinge angles [10][14] and strokes [11][13] are generally considered.

Our proposed algorithm is implemented on characters written at will by the subjects. Promising identification results with the accuracy 80% are gained when the number of Chinese characters is around 120. Compared with some traditional methods [1][5][6], the proposed method bears the inspiring advantage of weakened demand for character number, as well as low computational cost. The introduction of feature vector with selected components has further optimized the overall performance.

The remainder of the paper is organized as follows. In Section 2, the handwriting data set is introduced. In Section 3, we describe the proposed algorithm. In Section 4, we detail relevant experiments and evaluations. The conclusion is drawn in Section 5.

2   Handwriting Data Set

To demonstrate the effectiveness of the proposed algorithm, we established the NLPR handwriting data set. The samples are collected using a Wacom tablet. 40 Chinese subjects were required to write for three different pages including approximately 600 Chinese characters. Under standard conditions, the samples were collected in a period from one week to two months.

3   The Proposed Method

Traditional pattern recognition framework is adopted in our handwriting identification system, and the flow chart is shown in Fig. 1.

3.1 Preprocessing

The writing samples are preprocessed stepwise as follows and the final level of feature extraction is fixed on the stroke segments.
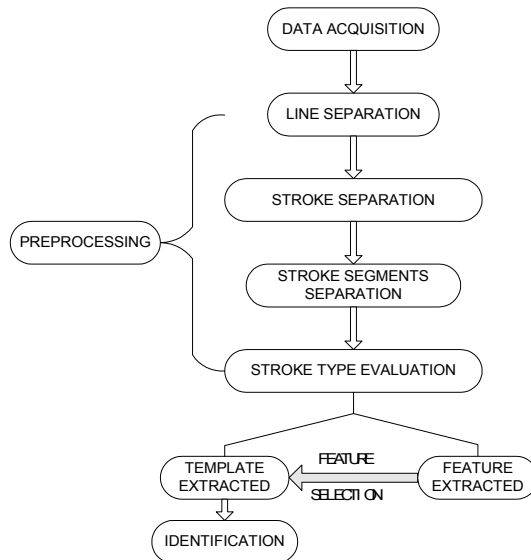
Fig. 1. Schematic diagram of the proposed algorithm

Line Separation. We start with the writing lines. The result of line separation for the sample in Fig. 2 (a) is shown in Fig 2 (b). We adopted the geometry moments to describe a line. For a given image, its geometric central moment μ_{pq} is calculated as follows:

$$\mu_{pq} = \sum_{i=s1}^{e1} \sum_{j=s2}^{e2} (x_i - x_0)^p (y_i - y_0)^q f(x_i, y_i) \tag{1}$$

where $f(x_i, y_i)$ is the pixel at (x_i, y_i) in the image, and (x_0, y_0) denotes the centroid position of the image. s1, s2, e1 and e2 denote the range of the points within the image respectively. The inclination θ may be calculated by

$$\theta = \frac{1}{2} \tan^{-1} (\frac{2\mu_{11}}{\mu_{20} - \mu_{02}}) \tag{2}$$

Stroke Separation. Here a stroke is defined as the dots ranging from the moment the penpoint lands on the tablet surface to the moment the penpoint takes off from the surface. Fig. 2. (c) shows the result of stroke separation, where dark dots denote the end points of the strokes.

(a)   (b)

(c)   (d)
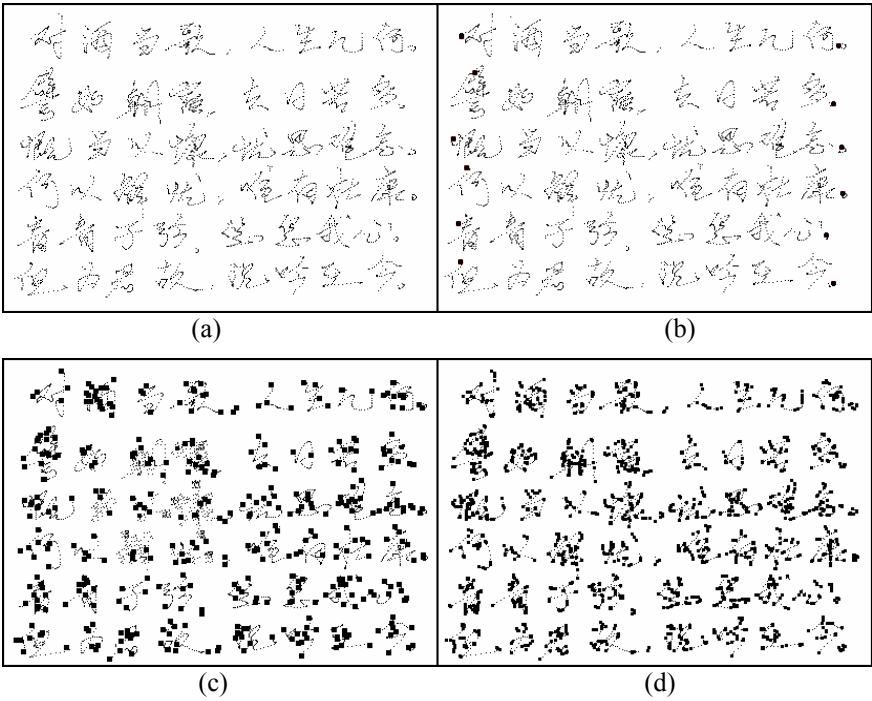
Fig. 2. Preprocessing. (a) An input sample. (b) The document after line separation. The dark dots represent the start and end points of a segment. (c) The strokes. (d) The stroke segments

Table 1. Primary stroke types in Chinese characters

Serial Number	1	2	3	4	5	6	7	8	9	10	11	12	0
Segment Type													Others
Structural Proportion	13%	36%	10%	4%	<1%	6%	14%	<1%	4%	4%	<1%	4%	>2%
Sample Proportion	8%	22%	9%	3%	<1%	4%	9%	<1%	5%	3%	<1%	4%	>20%

Stroke Segments Separation. The strokes constitute the cornerstone of the whole algorithm. Each stroke is separated into one or more stroke segments according to its configuration. The stroke segments are short trajectories with delimited orientation and without self-overtracing. The result is shown in Fig. 2 (d).

Stroke Segment Type Evaluation. According to the structural constituents of the Chinese characters [12], 12 primary stroke segment types are extracted, as shown in Tab. 1, and the arrow around each trajectory indicates the writing direction. The structural proportion and the sample proportion are also listed below respective segments.

The segment types arranged in the order of their saliency. Each segment is assigned by referring to its orientation and its correlation with the pertinent stroke, and then merged into the corresponding cluster. The segments from type 8 to type 12 are arcs frequently found in cursive Chinese characters. We take recourse to time information to allocate stroke segments into these types. To identify the type of an arc, the cross product of two vectors S_1 and S_2 formed by the start point, the middle point and the end point are adopted. From the sign of the coefficient of k, the rotation of an arc is decided.

$$\overline{S}_1 \otimes \overline{S}_2 = \begin{pmatrix} i & j & k \\ a_1 & b_1 & 0 \\ a_2 & b_2 & 0 \end{pmatrix} = (a_1 b_2 - a_2 b_1)k \tag{3}$$

3.2 Feature Extraction

Local features are extracted from the first 12 types of stroke segments. The histograms of the segments indicate that their attributes could be modeled as Gaussian distribution. Therefore, the pressure, the azimuth, the velocity and the altitude are selected to represent the writing gestures and movements. The velocity, which varies with writing pressure, is considered within the strokes. Because no explicit relationship between the writing pressure and writing velocity is found [13], they are handled separately. Four Gaussian models are constructed for each type of stroke segment, and the feature vectors are extracted from the Gaussian Models as templates.
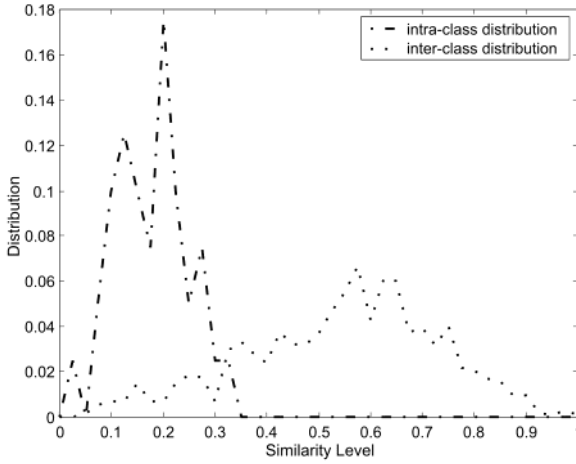
Fig. 3. Distribution of intra-class and inter-class similarity level

3.3 Identification

We use the method of template matching to identify the writer. In case the writing sample does not have enough strokes for statistic analysis and feature extraction, we select the dimensionality of the feature vectors adopting the following rule:

$$V(X) = \arg\min_{j \in J} \{ \sum_{i=1}^{96} w_i \cdot c_i(X) \cdot d(X_i, S_{ij}) \} \tag{4}$$

where $V(X)$ denotes the matching score, w_j is the weight parameter, and $c_j(X)$ is the piecewise non-linear function for the evaluation of the feature component, $d(X_i, S_{ij})$ is the discrimination function between X_i and S_{ij}, and n denotes the total number of stroke segments. According to the proposed algorithm, the distributions of intra-class and inter-class similarity levels are shown in Fig. 3.

4   Experiments and Evaluations

Experiments are carried out on the NLPR data set. Among the three pages of handwriting sample, two pages are taken out for template extraction, and the remaining page of approximately 180 characters for test.

In the preliminary experiments, we adopted the method without feature reduction. Two pages of handwriting are adopted for template extraction, and the identification accuracy is 62.5%. With the introduction of feature reduction, the algorithm improves the accuracy to 80% (32 out of 40).

Feature reduction bears the advantage of decreased computational cost with improved accuracy. Previous research [15] on dynamic analysis has adopted the

weight coefficients correlated with the number of writers [13], which means high time cost for template extraction and frequent updates of the feature templates. The proposed method avoids such process by adopting the selected feature vectors. Compared with other stroke-based methods [11][13], the stroke segments extracted have been proved to be representative of the Chinese characters with good performance.

The performance of the proposed method is also tested with increasing number of characters, and the best result is obtained when the number of writing lines is beyond 6, including approximate 120 characters. More characters bring no negative effects on the performance of the proposed algorithm. From the result shown in Fig. 4 we can see that pressure-based writer identification is not reliable, so pressure information plays an auxiliary role in the overall performance of the algorithm. The writing angle embodies the writing style better than other features when the number of characters is small. But as the number of writing character increases, the combination of all the features leads to the best result. Meanwhile, some proficient writers are identified with fewer characters, and in the least case only 18 characters are written.

5 Conclusions

The main contribution of the current work lies in the validation of the proposed strategy for text-independent handwriting analysis and writer authentication using selected features. We extracted features concerning writing pressure, velocity and pen inclination from 12 primary types of stroke segments. The performance of the method is tested on Chinese characters, but the techniques are likely productive for analyzing other Asian characters. Moreover, tests on different curves in Western characters are also worthwhile [16].
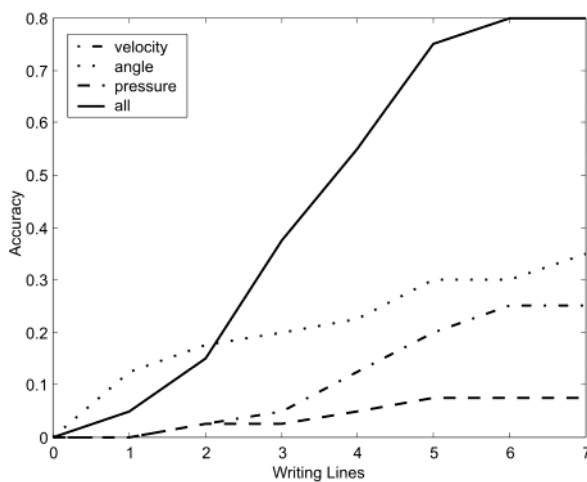
Fig. 4. Performance comparison of different features

Acknowledgements

References

[1]   Rejean Plamonton and Sargur N. Srihari, "On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey", IEEE Trans. on PAMI, Vol. 22, No. 1, Jan. 2000.
[2]   Mario E. Munich and Pietro Perona, "Visual Identification by Signature Tracking", IEEE Trans. on PAMI, Vol. 25, No. 2, pp. 200-217, Feb. 2003.
[3]   Veljko Potkonjak, Mirjana Popovic, Milailo Lazarevic and Jelena Sinanovic, "Redundancy Problem in Writing: From Human to Anthropomorphic Robot Arm", IEEE Trans. on Systems, Man, and Cybernetics, Vol. 28, No. 6, pp. 790-805, 1998.
[4]   Veljko Potkonjak, Dragan Kostic, Spyros Tzafestas, Mirjana Popovic, Mihajlo Lazarevic and Goran Djordjevic, "Human-like Behavior of Robot arms: General Considerations and the Handwriting Task—Part II : the robot arm in handwriting", Robotics and Computer Integrated Manufacturing, 17 (2001) 317-327.
[5]   Graham Leedham and Sumit Chachra, "Writer Identification Using Innovative Binarised Features of Handwritten Numerals", ICDAR'03.
[6]   Yasushi Yamazaki, Tsuyoshi Nagao and Naohisa Komatsu, "Text-indicated Writer Verification Using Hidden Markov Models", ICDAR'03.
[7]   Yong Zhu, Yunhong Wang and Tieniu Tan, "Biometric Personal Identification Based on Handwriting", Proc. 15th International Conference on Pattern Recognition, Vol.2, pp. 801-804.
[8]   L. Schomaker, M. Bulacu and M. van Erp, "Sparse-Parametric Writer Identification Using Heterogeneous Feature Groups", ICIP 2003.
[9]   Mi-Gyung Cho and Hwan-Gue Cho, "A Script Matching Algorithm for Oriental Characters on PDAs", ICPR02, Vol. 2, pp. 957-960.
[10]   Ithipan Methasate and Sutat Sae-Tang, "On-line Thai Handwriting Character Recognition Using Stroke Segmentation with HMM", Applied Informatics International Symposium on Artificial Intelligence and Applications, Feb. 2002.
[11]   Yasushi YAMAZAKI, Yousuke MIZUTANI and Naohisa KOMATSU, "Extraction of Personal Features from Stroke Shape, Writing Pressure and PenInclination in Ordinary Characters", ICDAR'99.
[12]   http://www.chancezoo.com/hz/hzbhtjtx.htm.
[13]   L. R. B. Schomaker and R. Plamondon, "The Relation between Pen Force and Pen-Point Kinematics in Handwriting", Biological Cybernetics, 63, 277-289, 1990.
[14]   In-Jung Kim and Jin-Hyung Kim, "Statistical Character Structure Modeling and Its Application to Handwritten Chinese Character Recognition", IEEE Trans. on PAMI, Vol. 25, No. 11, Nov. 2003.
[15]   Vishvjit S. Nalwa, "Automatic On-Line Signature Verification", Proceedings of the IEEE, Vol. 85, No. 2, Feb. 1997, pp. 215-239.
[16]   A. Lawrence Spitz, "Determination of the Script and Language Content of Document Images", IEEE Trans. on PAMI, Vol. 19, No. 3, Mar. 1997.

Searching for an Optimal Reference System
for On-Line Signature Verification
Based on (x, y) Alignment

Juan J. Igarza, Lorea Gómez, Inma Hernáez, and Iñaki Goirizelaia

Department of Electronics and Telecommunications
University of the Basque Country
Alameda Urquijo s/n, 48013 Bibao, Spain.
{jigarza,lorea,inma,igoirizelaia}@bips.bi.ehu.es

Abstract. The handwritten signature is the expression of the will and consent in daily operations such as banking transactions, access control, contracts, etc. However, since signing is a behavioral feature it is not invariant; we do not always sign at the same speed, in the same position or at the same orientation. In order to reduce the errors caused in verification by these differences between original signatures we have introduced a new concept of reference system for the (x, y) coordinates from experimental results. The basis of this concept lies on using global references (centre of mass and principal axes of inertia) instead of local references (initial point and initial angle) for a recognition system based on local parameters. The system is based on the hypothesis that signing is a feedback process, in which humans react to our own signature while writing it following patterns stored in our brain.

1   Introduction

Traditionally, the dynamic handwritten signature verification process by means of Hidden Markov Models (HMM) consists on the following stages: digitization of the signature, processing, features extraction, verification and finally decision.

From the graphics tablet the (x, y) coordinates that describe the movement of the signer's hand along the paper are obtained. These coordinates show the position of the pen every 10ms, obtaining a sequence of samples. Due to signature variability, the signal extracted from the tablet can differ for the same signing person, depending on the speed of writing, the size of the signature, its position and orientation.

The causes of this variability can be divided into two groups. On one hand the number of samples of the sequence depends on the length of the signature and on its size [1], since the writer does not always last the same in signing and the physical space provided to sign may vary. On the other hand, the coordinates must be processed to avoid variability due to the position and orientation of the signature since from the corrected (x, y) coordinates several characteristics will be extracted, such as speed or acceleration, to be used in the verification process.

In fact, the signature may be made in different positions of the tablet at different instants so coordinates should be referenced to an alignment point. Moreover, due to

rotations in signer's hand or in the tablet the orientation of the signature is variable, so the angle of the reference axes is also needed.

In this sense, some authors [2] propose the use of the first sample of the signature as a reference in order to reduce position variability and some others suggest the initial angle [3] to avoid rotation variability.

This paper discusses the variability of the (x, y) coordinates and proposes a new method for processing the coordinates by means of an alignment based on global references. Two different aspects are researched in more detail: the use of the center of mass or centroid of ink as the center of the reference system in the alignment and the use of the principal axes of inertia to find the axes that best align the signatures.

2   Reference Point for (x, y) Coordinates Alignment

As stated, the signatures must be aligned for the enrollment and verification processes to make the (x, y) coordinates independent of the position of the tablet. We must find the best reference point in order to get the best alignment between the signatures.

a) Alignment using the initial sample of the signature as a reference
When we use the first point of the signature (x_0, y_0) as the center of the (x, y) coordinates reference system the alignment is carried out as follows:

$$x' = x - x_0 \qquad y' = y - y_0 \tag{1}$$

b) Alignment using the center of mass as a reference
The (x, y) coordinates of the centroid of ink or centre of mass of the signature (N-particle system) are:

$$x_{cm} = \frac{1}{N}\sum x \qquad y_{cm} = \frac{1}{N}\sum y \tag{2}$$

In this case, the alignment is carried out as follows:

$$x' = x - x_{cm} \qquad y' = y - y_{cm} \tag{3}$$

The center of mass is calculated applying the Breshingham algorithm to every two consecutive samples of the signature. This leads to involve in the computation not only the pen-down samples but also intermediate points that are calculated by means of this segment approach algorithm.

Figure 1 shows the signatures of two users and the result of superimposing ten signatures referred to the coordinates system centered in the initial point (Fig. 1b) or the center of mass (Fig. 1c). As can be seen in the figures, the global dispersion is lower when the signatures are referred to the center of mass. Namely, we get a better alignment of the signatures with little deviation in x and y.

3   Reference Angle for the (x, y) Coordinates Alignment

The variability in the orientation of the signature produces noise in the verification. As stated before, the rotation may be caused by the writer's hand or by the position of the graphics tablet. This is why once the center of the new reference system is

defined; we should find a suitable reference angle to rotate the system axes. Once calculated, the coordinates will be rotated by means of a rotation matrix as follows:

$$x' = x\cos\theta - y\,sen\,\theta , \qquad y' = x\,sen\,\theta - y\cos\theta \qquad (4)$$

where x and y are the original coordinates, x' and y' the new ones, and θ the reference angle.

a) Alignment using the initial angle as reference
In this case the angle of reference is calculated as follows:

$$\theta_0 = arctg\left(\frac{y_1 - y_0}{x_1 - x_0}\right) \qquad (5)$$

From experimental results in previous studies [4] we know that this is a noisy angle. Figure 2 shows three signatures of the same user rotated by the initial angle and superimposed (the left side of the figure shows a zoom of the origin). It can be seen that the rotation by the initial angle does not get the correct alignment. Quite the opposite, it introduces noise since the initial angle might be random.

b) Alignment using the angle of minimum inertia as reference
The principal axes of inertia of a rigid body are a set of axes passing through the object's center of mass and relate to the object's mass distribution such that angular momentum imparted about a principal axis is not transferred to any other axis. The symmetry axis of a pencil is one example of a principal axis. An asymmetric two-dimensional signature has two mutually perpendicular principal axes, which can always be calculated unambiguously. One of them is known as the minimum moment of inertia.
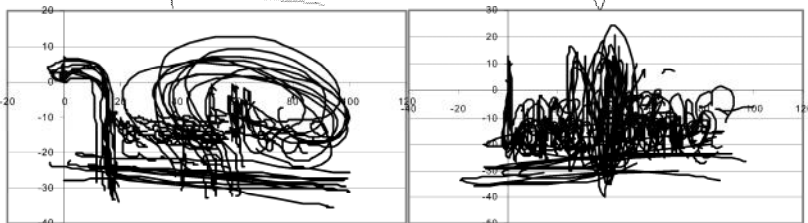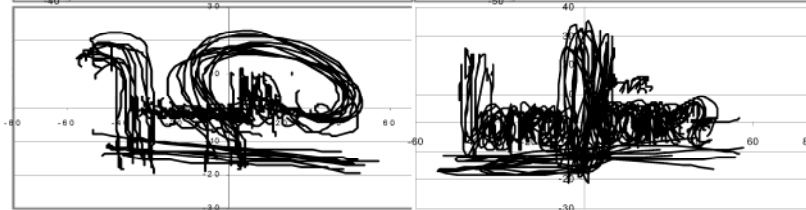
Fig. 1. 1a two signatures of two users. **1b**. Graphical superimposition of ten signatures referred to the coordinates system centered in the initial point. **1c**: Graphical superimposition of ten signatures referred to the coordinates system centered in the mass center
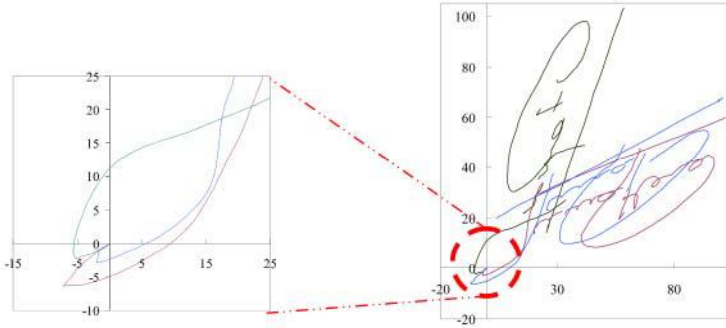
Fig. 2. Three signatures of the same user rotated by the initial angle and superimposed. The left side of the figure shows a zoom of the origin

The angle of minimum inertia gives an indication of the orientation of the signature, and there is a direct relationship between the angle of inertia and the writing trend.

To avoid variability in orientation in the process of matching the acquired data with the reference data we should rotate the (x, y) coordinates of every signature by an angle that equals the angle of minimum moment of inertia but in the opposite direction. Namely, we align all the signatures so as to make the verification independent of the writing direction and eliminate noise.

The angle of minimum inertia θ is obtained using the formula below:

$$\theta = \frac{1}{2}\arctan\left[\frac{2I_{xycm}}{I_{ycm} - I_{xcm}}\right] \tag{6}$$

where I_{xcm}, I_{ycm} and I_{xycm}, are the moments of inertia referred to the axes parallel to the origin of coordinates and crossing the center of mass. They are calculated using the following procedure:

   I. First, the moments of inertia referred to the origin of (x, y) coordinates and the crossed moment of inertia are calculated as follows:

$$I_{x0} = \sum y^2 , \; I_{y0} = \sum x^2 , \; I_{xy0} = \sum xy \tag{7}$$

  II. Secondly, these moments must be referenced to the previously calculated center of mass. This problem is solved by means of the Steiner theorem:

$$I_o = I_{cm} + d^2 N \tag{8}$$

where,

- Icm is the moment of inertia referred to an axis that crosses the center of mass, I_o is the moment referred to an axis parallel to the previous one and d is the distance between the two parallel axes.
- N is the number of pen down samples that form the signature and the intermediate points calculated by the Breshingam algorithm.

In this way, the moments of inertia referred to the center of mass can be calculated using:

$$I_{xcm} = I_{x0} - Ny_{cm}^2 , \; I_{ycm} = I_{y0} - Nx_{cm}^2 , \; I_{xycm} = I_{xy0} - Nx_{cm}y_{cm} \tag{9}$$

4   Results

We have performed the evaluation of the proposed references over a 100 user's database [5]. This database contains 25 original signatures from each user, and 25 skilled forgeries of each user. The verification process is carried out by means of the Hidden Markov Models. We have chosen a Left to Right topology with a fixed number of states (six) so as to obtain satisfactory results in generating the model and in the verification process.

The experiments involve the use of different sets of parameters:

a) Training of the HMM's using as a parameter set only (x, y) coordinates referred to the center of mass and to the initial sample.
b) Training of the HMMs using only the (x, y) rotated coordinates referred to the center of mass and not rotated but referred to the center of mass.
c) Training of the HMMs using all the available parameters (rotated coordinates and seven more parameters: pressure, azimuth, acceleration, inclination, acceleration in x axis, linear and angular speed).

Carrying out the verification using only the (x, y) coordinates referred to the center of mass and the initial sample we get the EER (Equal Error Rate) values shown in Table 1. As can be seen, an important improvement is achieved when using the center of gravity as reference.

Training the system with the rotated coordinates has given different results depending on the kind of signature:

Case 1 Small variation in the EER

For the majority of the users (72%) the EER has no varied (+-4% deviation using the rotated or non-rotated coordinates in the verification).

Table 1: Comparison of the EER for the verification using only (x, y) coordinates referenced to the center of mass (cm) or the initial point (ip)

	EER cm (%)	EER ip (%)	ΔEER (%)
Mean value	24.4%	42.16%	-17.76%

Fig. 3. As an example of EER improvement by means of coordinates rotation we show one signature of a user where the angle of minimum inertia varies in the range [-2.72⁰, 1.62⁰] among the 25 original signatures in the database
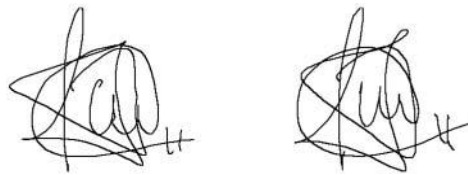
Fig. 4. Two signatures of the same user for which the deviation of the minimum angle of inertia is 77.11° ([-38.77°, 38.34°]) among the 25 signatures of the database

Case 2 Improvement in the EER

The rotation of the coordinates leads to a remarkable improvement of the EER for those signatures that have a nearly constant orientation (that is, the angle of minimum inertia presents little deviation). The improvement is especially important for those signatures with a horizontal trend, since for these ones the user has a precise reference and he/she is able to keep the same orientation at every time he/she signs. Figure 3 shows one example of this case.

Case 3 Worsening of the EER

There are some signatures for which the instability in writing provokes a wide deviation in the angle of minimum inertia. For those signatures, if we use rotated coordinates in the verification the results are worse than when we use non-rotated coordinates. Figure 4 shows an example of this case. The signatures of this user show a deviation of 77.11° ([-38.77°, 38.34°]) among his 25 signatures of the database. This is due to the instability of the signature's shape.

The instability found in the signature can be related with the EER found in the verification process using only the (x,y) coordinates. Considering that a user presents a stable performance if he or she gets an EER less than or equal 20%, we find that approximately half of the signers (52%) are stable in our database. Using only this set of stable signers, the EER does not show an improvement, as shown in table 2.

Table 2. Comparison of the EER for the verification using only (x, y) coordinates rotated the initial angle and the angle of minimum inertia

	EER with rotated (x, y)	EER with non-rotated (x, y)
Mean value	14.3%	14.7%

Table 3. Final results of the project using 9 local parameters of the signatures using global references: (x, y) coordinates referred to the canter of mass and rotated the minimum angle of inertia, pressure, azimuth, acceleration in x axis, linear and angular speed

Mean EER	Mean FAR	Mean FRR
4.66%	0%	9.64%

Nevertheless, a worsening in the EER of the system trained only with the rotated coordinates does not imply a loss of effectiveness in the verification as long as for those kind of signatures the other parameters are used, and the verification success

lies on those parameters. We trained the system with all the available parameters (coordinates referred to the center of gravity and rotated to the angle of minimum inertia, pressure, azimuth, acceleration in the x axis, and linear and angular speed).

5   Conclusions

As stated herein in spite of the fact that the way of signing is a behavioral feature, it presents certain variability. In fact, at first the signer places the hand on the tablet randomly; however, as long as he writes the signature becomes more stable in size, shape, speed… In other words, signing is a feedback process. This feedback is mainly visual, since if we try to sign with our eyes closed or with an inkless pen will make an amorphous signature.

In order to profit from this feedback, we propose the signature alignment based on global references. We have evidenced that using the center of gravity as the origin of the reference system instead of the initial sample improves the mean EER in 17.76%. Furthermore, using as reference the axes obtained by rotating the main axes a quantity equal to the angle of minimum inertia an added improvement in the EER of 0.4% is obtained.

References

[1]   V. S. Nalwa, "Automatic on-line signature verification", Proceedings of the IEEE vol 85 No 2, Feb 1997 pp. 215-239.
[2]   Javier Ortega-Garcia, Joaquin Gonzalez-Rodriguez, Danilo Simon-Zorita y Santiago Cruz-Llanas, "From Biometrics Technology to Applications Regarding Face, Voice, Signature and Fingerprint Recognition Systems". "Biometric Solutions for Authentication in an e-World" Chapter 12. Ed: Kluwer Academic Publishers, 2002 pp. 312-319.
[3]   L. Yang and B.K. Widjaja and R. Prasad, "Application of Hidden Markov Models for signature verification", Pattern Recognition, Vol 28, No 2, 1995.
[4]   Juan J. Igarza, Iñaki Goirizelaia, Koldo Espinosa, Inmaculada Herná z, Raúl M éndez, Jon Sánchez. "On-line Handwritten Signature Verification Using Hidden Markov Models" Progress in Pattern Recognition, Speech and Image Analysis. LNCS Vol. 2905 Springer November, 2003 pp. 391-399.
[5]   J. Ortega, J. Fierrez, D. Simon, J. Gonzalez, I. Hernaez, J.J.Igarza, C. Vivaracho, D. Escudero, Q.I. Moro "MCYT baseline corpus: a bimodal biometric database", IEE Proceedings Vision, Image and Signal Processing. Vol. 150(6) December, 2003 pp 395-401.

Reduction of Feature Statistics Estimation Error for Small Training Sample Size in Off-Line Signature Verification

B. Fang and Y. Y. Tang

Department of Computer Science, Hong Kong Baptist University
Kowloon Tong, Hong Kong
{fangb,yytang}@comp.hkbu.edu.hk

Abstract. In this paper, we propose two methods to tackle the sparse data problem in off-line signature verification. The first one is to artificially generate additional training samples from the existing training set by an elastic matching technique. Feature statistics are estimated by using the expanded training set. The second approach applies regularization technique to the sample covariance matrix to overcome the problem of inverting an ill-conditioned covariance matrix and obtains stabilized estimation of feature statistics. Experimental results showed that both techniques are able to produce significantly improved verification accuracy when implemented with a set of peripheral features.

1 Introduction

The problem of insufficient samples is prominent in off-line signature verification due to the private and confidential nature of the signatures which further makes it a formidable task [1]. A considerable amount of effort has been devoted to design a classifier particular in small sample size situations [2]. Nonparametric classifiers like a nearest neighbor classifier may be believed to be effective [3] to avoid unreliable estimation of statistical parameters. However, the off-line signature verification is usually considered to be a specific case of the typical two-class classification problem with only genuine training samples of one class available to model the classifier [4]. Hence, non-parametric approaches are not suitable for off-line signature verification application and threshold classifiers based on distance such as Mahalanobis distance are widely adopted. The maximum likelihood (ML) estimator is used to estimate feature statistics involved in computation of the distance. However, estimation error can be a significant problem for the limited amount of training samples which renders the estimation of the statistics unreliable [5]. Furthermore, when the number of training samples is less than the dimensionality of the feature space, the sample covariance matrix is singular and its inverse does not exist. One ways to overcome the problem is to employ feature extraction or selection to reduce the feature dimensionality. However, the between-class scatter matrix as selection criterion used by these approaches cannot be computed due to the special characteristics of signature verification that training samples come from genuine class only. Hence, these approaches are not applicable in the study to estimate stable feature statistics.

Fig. 1. (a) Samples of genuine signatures. (b) Samples of forgeries

In this paper, we first propose to employ an elastic distortion model to artificially generate additional training samples to compute stable feature statistics. Another approach is also presented which adopts matrix regularization technique to sample covariance matrix. Experimental results showed that both techniques produced significantly improved verification accuracy when implemented with a set of peripheral features and a Mahalanobis distance threshold classifier.

2   Artificially Generated Samples for Training

In this section, we describe a method which uses 2-D elastic structure matching algorithm to generate artificial signature patterns. Examples of original signature patterns are illustrated in Fig. 1(a). These generated samples represent more variations in handwriting than the original. Hence, model statistics are more representative of the true sampling distribution by using the generated samples together with the original patterns for statistics estimation.

A guideline on how much distortion applied to original signatures must be set up to make sure that the generated samples lie inside the normal variations of the authentic signatures. Our guideline is to identify corresponding strokes between two signatures, measure the distortions in terms of displacement vectors and generate additional samples within the space spanned by the displacement vectors. The rationale is that the generated signatures are then within the bounds of variations of the original authentic samples. The key role is to identify corresponding strokes by elastic matching algorithm [6] briefed as follows.

Let the two signatures to be elastically matched be called the '*Template*' and the '*Input*'. Thinning is performed and the one-pixel wide lines are approximated by fitting a set of straight lines which are further divided up into smaller segments of approximately equal lengths. Each of these smaller segments is referred as an 'element'. Thus the signature pattern is in turn represented by a set of elements. The two sets of elements are put on top of each other and the elements of the *Template* are then gradually attracted towards elements of the *Input* in successive iterations so that the corresponding elements finally meet. The objective is to achieve maximum similarity between the resulting patterns while minimizing the deformation. This is

achieved through the minimization of a cost (or energy) function used to guide the movements of the *Template* elements towards the *Input* elements [7].

   At the end of matching two sets of elements, the corresponding strokes of the two signatures should hopefully be nearest to each other and a list of matches between *Input* elements and *Template* elements is drawn up. Assuming mid-point and two end points of each *Template* element as feature points, the positional displacement vector for each feature points can be computed by referring to the original undistorted *Template* and *Input* signatures. To generate an artificial signature, each pixel of the *Template* signature are moved halfway along its move vector towards the corresponding stroke of the *Input* signature. The move vector for each pixel of the *Template* signature can be defined as follows.

$$\Delta M = \sum_{j=1}^{N} w_j \mathbf{P}_j \tag{1}$$

   N is the size parameter of the Gaussian window which establishes neighborhoods of correlation, w_j are the correlation weights in the Gaussian correlation window where $w_j = \exp\left(-(j-1)^2/(2 \times N^2)\right)$ and $j = 1,2,\ldots,N$. $\mathbf{P}_j$ is the positional displacement vector of feature point j in the Gaussian window sorted in the order of increasing distance from that pixel. $\mathbf{P}_j = \mathbf{T}_j - \mathbf{I}_{i(j)}$, where $\mathbf{T}_j$ is the position vector of feature point j of the *Template* signature and $\mathbf{I}_{i(j)}$ is the position vector of the feature point of *Input* signature which is matched to j. Finally, a morphological operation of closing is applied to bridge gaps and fill holes in the distorted pattern using a circular structural element with the smallest radius of one pixel. Some examples of the artificially generated additional signature patterns are illustrated in Fig. 2.

(a)   (b)   (c)

Fig. 2. (a) The generated additional signature patterns, (b) overlapped images of the original *Template* signature (black) and *Input* signature (grey), (c) overlapped images of the generated additional signature pattern (dark gray) and the original *Template* and *Input* signatures

3   Covariance Matrix Estimation by Regularization Technique

There are different approaches to tackle the problem of having not enough samples for training. The approach proposed in Section 2 is to expand the training set by generating additional samples from existing samples. Another possible approach is to apply varying degrees of regularization to the sample covariance matrix directly from the training samples to get a stable estimation of the covariance matrix. Friedman [8] proposed a covariance matrix estimation method called "Regularized Discriminant Analysis"(RDA) which is a two dimensions optimization over mixtures of the sample covariance matrix, common covariance matrix and the identity matrix multiplied by a scalar. RDA regularizes the covariance estimate toward a diagonal matrix to stabilize the covariance estimate. Hoffbeck et al. [9] proposed another covariance matrix estimator called "Leave-One-Out Covariance"(LOOC). The LOOC takes the form of mixture of sample covariance, diagonal sample covariance, common covariance matrix and diagonal common covariance matrix. The mixing parameter is selected so that the corresponding mixture of covariance matrices maximizes the average leave-one-out log likelihood of each class for the training samples. However, signature verification is a special two-class classification task with only information of the genuine pattern class being provided to estimate feature statistics. The common covariance does not exist. Therefore, straightforward applications of these techniques are not suitable to signature verification. To cope with the situation, a ridge-like estimator for covariance matrix estimation is proposed in this study.   This regularization technique called LOOC1 is to add some constant values to diagonal elements of the sample estimate of the covariance matrix to overcome the inverse of an ill-conditioned covariance matrix and stable the covariance matrix estimation. The form of LOOC1 is as follows,

$$\hat{\Sigma}_i(\alpha_i) = \alpha_{i1} \frac{tr(\Sigma_i)}{p} I + \alpha_{i2} \Sigma_i \qquad (2)$$

where Σ_i is the sample covariance matrix, p is the feature dimensionality and $tr(\Lambda)$ is the trace of matrix Λ. The elements of the mixing parameter $\boldsymbol{\alpha}_i = [\alpha_{i1}, \alpha_{i2}]^T$ are restricted to $\alpha_{i1} + \alpha_{i2} = 1$.

The strategy of maximizing the average class leave-one-out likelihood is adopted as the criterion to select the appropriate model parameters because the number of training samples is small. The value of the mixing parameter $\boldsymbol{\alpha}_i$ is selected to maximize the average likelihood of omitted samples for the training samples. The technique is to remove a sample, estimate the mean and covariance matrix from the remaining samples, then compute the likelihood of the left-out sample using the mean and covariance matrix estimates. Each sample is removed over in turn, and the average log likelihood is computed over all the left out samples. This computation is repeated for several values of $\boldsymbol{\alpha}_i$ being sampled between 0 and 1 with a fine step of 0.1 subject to $\alpha_{i1} + \alpha_{i2} = 1$. Mixing parameter $\boldsymbol{\alpha}_i$ with the highest score of average log likelihood is selected to compute the final estimation of covariance matrix using all training samples.

4   Experimental Results

The proposed methods were evaluated on a database of 1320 authentic signatures and 1320 forgeries. The authentic signatures were collected from 55 volunteers over a period of one month, with each person contributing 24 signatures. Twelve other persons were recruited as forgers. Each forger produced 110 forgery signatures (two forgeries for each of the 55 volunteers). The forgers were instructed to briefly study each genuine signature before imitating it. The forgeries could be considered as semi-skilled forgeries. All signatures were written into a 1 inch × 3 inch white rectangular box with light yellow color rims. The signatures were digitized by a scanner at a resolution of 300 dots per inch. Preprocessing is done to extract the signature image from the box and fit a virtual frame to enclose the image. The slant and baseline information of signature patterns is considered important features for individual writer and may provide discriminatory measure for relative feature extraction such as peripheral features in classification. Hence, writer specific slant and baseline normalization was not conducted in our study. The peripheral features ET1 and DT12 are employed in the study to represent the signature patterns because they can provide more discriminatory power than other local and global features [10]. They are normalized to be shift and scale invariant.

Two types of errors are defined to evaluate verification accuracy: the Type I error rate or the false rejection rate (FRR), and the Type II error rate or false acceptance rate (FAR). The average error rate is defined to be the average of FRR and FAR. Leave-one-out approach is used for determination of the false rejection rate. 23 samples were used for training and one sample was used for testing. For the method to produce additional samples, 2 samples were randomly drawn from this set of 23 training samples for performing elastic matching and one additional sample was generated. Feature statistics of mean $\hat{\mu}$ and covariance matrix $\hat{\Sigma}$ are then computed out using expanded training data set which consists of the artificially generated samples and the original 23 samples. For regularization approach LOOC1, mean $\hat{\mu}$ and covariance matrix $\hat{\Sigma}$ were estimated only by using these 23 genuine samples.

In the test phase, given a test signature with feature vector $\mathbf{F}_t$, the Mahananobis distance $d_m(\mathbf{F}_t, \hat{\mu}, \hat{\Sigma},)$ is used to measure the dissimilarity between the input test signature and the set of training signatures according to:

$$ d_m = \sqrt{(\mathbf{F}_t - \hat{\mu})^T \hat{\Sigma}^{-1} (\mathbf{F}_t - \hat{\mu})} \qquad (3) $$

The test signature is accepted as authentic if the dissimilarity is less than a given threshold, otherwise, it is considered as a forgery.

The ET1 and DT12 features described in [10] were extracted with different quantities of N_p which is the number of strips used to segment the signatures vertically and horizontally to calculate the values of ET1 and DT12, N_p = 1,2,3,5,7. When the number of training samples (N = 23) is less than the feature vector dimension ($8 \times N_p \geq 24$), the sample estimation of covariance matrix is singular. Hence the Mahalanobis distance d_m in Equation (3), which involves the non-existent inverse of the matrix, is not computable. To enable the distance to be computed, the off-

diagonal elements of the estimated covariance matrix were set to zero. Only the variances of each vector dimension were used and the covariance between different dimensions was ignored. This is equal to adopt a weighted Euclidean distance for threshold classifier.

Experiments results are shown in Table 1. With higher feature dimensionality $(8 \times N_p \geq 24)$, the Mahalanobis distance can not be computed only using the original training samples due to the failure to convert the covariance matrix where the number of training sample is equal to 23. On the other hand, weighted Euclidean distance classifier works. When feature dimension exceeds 24, the covariance matrix is estimated through the use of either expanded data set or the matrix regularization technique proposed in this study. It is clear that the verification error rates were reduced from 16.8% (by using only the small training sample set) to 11.4% and 12.6% by employing the additional training samples and LOOC1 respectively when feature dimension equal to 24. This demonstrates that both methods proposed are able to reduce estimate errors in feature statistics and as a result to produce significantly improved verification accuracy.

Although the method of applying additional samples to train the verification system leads to better performance, it is time-consuming to artificially generate samples between pairs of genuine signatures. The average execution time is 171 minutes to generate 552 additional samples for each of the 55 writers. However, it is fast to extract the ET1 and DT12 features and compute their statistics which can be completed in a few minutes (195 seconds/writer). Thus the computation complexity for training using additional samples mainly depends on the process to generate samples. The verification can be done within a minute upon already trained system. On the other hand, the matrix estimator LOOC1 can finish model parameter selection in an average time of 35 seconds for one writer, thus the computation load for training which also involves feature extraction is comparably low (44 seconds/writer). Verification is as fast as in the method using addition samples.

Table 1. Average verification error rates with feature dimensions ranging from 8 to 56 (N_p = 1,2,3,5,7)

Training method	Form of covariance matrix	Average error rate for various feature dimensions (%)				
		8	16	24	40	56
Original set	Full matrix	19.2	18.1	N/A	N/A	N/A
Original set	Diagonal	21.5	19.5	16.8	16.4	15.6
Expanded set	Full matrix	15.1	12.9	11.4	12.3	13.3
Expanded set	Diagonal	19.0	16.6	14.8	14.5	14.4
LOOC1	Full matrix	17.2	14.8	12.6	13.8	14.9

5 Conclusions

The sparse data problem in off-line signature verification was tackled. In this study, we propose two methods to reduce the statistics estimation error. The first proposed method is to artificially generate additional training samples from the existing training set by an elastic matching technique. The method differs from existing ones in that it is more suitable for the generation of signature samples. Features statistics are estimated by using the expanded training data set. The second proposed method applies regularization technique to the sample covariance matrix optimization techniques to overcome the problem of inverse of an ill-conditioned covariance matrix and obtains stabilized estimation. Experimental results showed both techniques produced significantly improved verification accuracy when implemented with a set of peripheral features and a Mahalanobis distance threshold classifier.

References

[1] F. Leclerc and R. Plamondon, "Automatic signature verification: the state of the art— 1989-1993," *Int. J. Pattern Recognition and Artificial Intelligence*, vol. 8, no. 3, pp. 643-660, 1994.

[2] S.J. Raudys and A.K. Jain, "Small sample size effects in statistical pattern recognition: Recommendations for practitioners," *IEEE Tran. Pattern Recognition and Machine Intelligence*, vol. 13, no. 3, pp. 252-264, 1991.

[3] K. Fukunaga, *Introduction to statistical pattern recognition*, Second Edition, Boston: Academic Press, 1990.

[4] N. Murshed, R. Sabourin and F. Bortolozzi, "A cognitive approach to off-line signature verification," *Automatic bankcheck processing*, Singapore: World Scientific Publishing Co., 1997, pp. 339-363.

[5] F. O'Sullivan, "A statistical perspective on ill-posed inverse problems," *Statistical Science*, vol. 1, pp. 502-527, 1986.

[6] B. Fang, C.H. Leung, Y.Y. Tang, P.C.K. Kwok, K.W. Tse, Y.K. Wong, " Offline signature verification with generated training samples*," IEE Proceedings-Vision, Image and Signal Processing*, vol. 149 no. 2, pp. 85–90, April, 2002.

[7] C.H. Leung and C.Y. Suen, "Matching of complex patterns by energy minimization," *IEEE Transactions on Systems, Man and Cybernetics*, Part B, vol. 28, no. 5, pp. 712-720, 1998.

[8] J.H. Friedman, "Regularized Discriminant Analysis," *Journal of the American Statistical Association*, vol. 84, no. 405, pp. 165-175, 1989.

[9] J.P. Hoffbeck and D.A. Landgebe, "Covariance matrix estimation and classification with limited training data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 7, pp. 763-767, 1996.

[10] B. Fang, C.H. Leung, Y.Y. Tang, P.C.K. Kwok, Y.K. Wong, "A peripheral feature based approach for off-Line signature verification," in *Proc. of the 5th Joint Conference on Information Sciences*, Atlantic City, USA, March, 2000, pp.236-239.

Biometric Personal Authentication
Based on Handwritten Signals

Miguel G. Lizárraga and Lee L. Ling

School of Electrical and Computer Engineering, State University of Campinas
Albert Einstein Avenue, 400, PO Box 6101, Postal Code 13083-970, Campinas, SP, Brazil
{lizarrag,lee}@decom.fee.unicamp.br

Abstract. In this work is proposed a personal authentication method based on images composed of handwritten strokes that may represent symbols, words, signatures or any kind of drawings. This method is efficient, robust and can be easily implemented, requiring low CPU processing power and needs only five handwritten signal samples for enrollment. The authentication method has been evaluated under both verification and identification processes. In verification experiments random and skilled forgeries were used, while in identification experiments three different database configurations were setup. Performance of 0.7 % equal error rate and that of 93.7 % correct classification rate were achieved in the verification and identification processes, respectively.

1   Introduction

The personal authentication through more general handwritten signals is a natural extension of signature images authentication process. In this context, we define a handwritten signal as any image composed of freehand written strokes, like drawings, emblems, handwritten letters, symbols, characters etc. Thus, all signatures are special cases of handwritten signals, although not any handwritten signal can be considered a signature [1].

An authentication method based on any handwritten signal and not on signatures is motivated by the following observations:

§   Instead of using our own signature we can use any set of strokes that can be more simple and quick to be written.

§   It is not desirable that a signature change too much over the time, since they have to be always alike the ones previously registered in a notary. Handwritten signals are free to change at any time.

§   If we use the image of a handwritten signal as login in a computer system, it can be easily changed whenever it is needed, by performing a re-enrollment with a new handwritten signal

Motivated by these facts, we introduce the concept of biometric graphical passwords involving the utilization of handwritten signals for personal authentication.

A biometric graphical password is understood as any image of handwritten strokes that can be used for verification or identification. The biometric graphical password must be frequently and easily reproduced by a person keeping low variability over

form and style. It is expected that a biometric graphical password, as classical passwords based on typing a text in a keyboard, serves as a biometric key granting access to the resources of a particular system [2].

2   Database

The images stored in the database for our experimental investigation are composed of handwritten signals divided in two types. The first type is called "Signatures". This type of handwritten signal was sub-divided in two groups: "Genuine Signatures Group" and "False Signatures Group". The second type is called "Symbols" and it was also sub-divided in two groups: "Genuine Symbols Group" and "Skilled False Symbols Group", some images that belong to the Symbols are shown in Figure 1.

   The groups that the Signatures and Symbols were divided will be detailed as follow:

§   Genuine Signatures Group (SG): This group has a total of 1200 images. These signatures were collected from 40 individuals, each one contributing with 30 genuine signatures.
§   False Signatures Group (FG): This group is composed by a total of 50 signatures acquired from 50 different people who do not take part of the group of people having contributed with their genuine signatures.
§   Genuine Symbols Group (GSG): This group is composed of 600 images collected from 20 persons each contributing with 30 samples of one handwritten signal.
§   Skilled False Symbols Group (SFG): This group is composed of 150 images acquired from 15 different people, equally divided in 5 groups. Each group is given one different sample image from the GSG and each person in the group was asked to produce 10 forgery samples.

3   Proposed Method

In this section we describe the modules that implement the proposed authentication method: preprocessing, feature extraction and classifier.
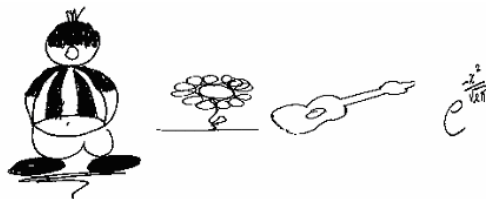
Fig. 1. Image samples of handwritten signals Symbols
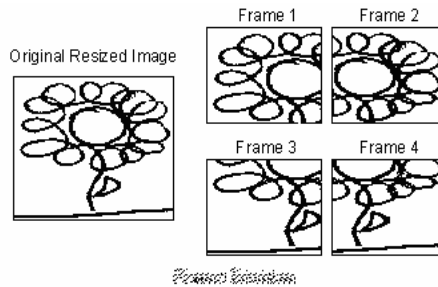
Fig. 2. An image divided in four frames

3.1 Image Preprocessing

With the intention of enhancing the images and simultaneously minimizing the intra-class variability and maximizing the inter-class variability, we divide the image preprocessing procedure in three stages. The first stage deals with the image bounding, the second one with the size normalization and finally the third partitions image into frames.

Image Bounding. The image bounding consists in eliminating all the blank space around the signatures or handwritten symbol images that does not contribute to the authentication process. This is done by a minimum rectangle cornering and limiting all handwritten strokes.

Size Image Normalization. In this work the resolution scale is associated with the number of pixels in the horizontal and vertical axis of the image. Each handwritten signal will result in two rescaled images that will be used in the feature extraction process. The size normalization is done in two stages. In the first stage, the image is only rescaled in the horizontal axis and in the second stage, the image is resized only in the vertical axis.

Frame Division. In this work we present a new approach where the image is divided into four frames, however with 50% overlapping the adjacent frames. The frame division has the objective of splitting the image in small regions. Each one of these regions will be used in the feature extraction process to generate vectors that will have the local information of the images. Fig. 2. shows the division scheme where the image is divided horizontally and vertically [3].

3.2 Features

For the proposed personal authentication method, three feature vectors are involved:

1. The slope of the handwritten signal image contour line
2. The slope of the handwritten signal image strokes
3. Correlation data between a template and test images

The extraction of the first and second feature vectors are more detailed in [4]. The correlation feature vector will be explained as follow.

Correlation Feature. The correlation feature vector is composed of 5 elements per frame. The reference image, called template, is previously acquired, during the enrollment process, by adding 5 images of handwritten signals from one person. The template is then divided in four template frames as explained in 3.1. The value of each element of the correlation feature vector is calculated performing the overlapping between the frame test over a template frame. The composition of those elements are described as follow:

1. The number of black pixels of the test frame
2. The number of pixels performing the overlapping between test and template frames are white in both of them
3. The number of pixels performing the overlapping between test and template frames are black in both of them
4. The number of pixels performing the overlapping between test and template frames are black in the test frame but are white in the template frame
5. The number of pixels that performing the overlapping between test and template frames are white in the test frame but are black in the template frame

4   Experiments

The experiments performed in this work used a simple minimum distance classifier. The experiment 1 was performed using the set of 40 genuine signature classes in conjunction with the 50 false signatures. The goal of this experiment is to measure the discriminant power of each of the three proposed feature vectors individually. The results of experiment 1 can be seen on table 1.

The experiment 2 was performed using the 40 genuine signature classes in conjunction with the 50 false signatures. This experiment has the goal of determine joint discriminating power of slope of the contours and the slope of the strokes based on (1), and the results can be observed on table 2:

$$p_{final} = p_{contour} \cdot p_{strokes} \tag{1}$$

Table 1. Average FRR_0, FAR_0 and EER of Experiment 1

Characteristics	FRR_0 (%)	FAR_0 (%)	EER (%)
Contours	7.4	5.6	3.0
Strokes	19.2	11.2	5.9
Correlation	10.3	7.2	6.0

Table 2. Average FRR_0, FAR_0 and EER of experiment 2

FRR_0 (%)	FAR_0 (%)	EER (%)
4.8	3.6	1.5

Table 3. Average FRR_0, FAR_0 and EER of experiment 3

FRR_0 (%)	FAR_0 (%)	EER (%)
2.4	1.2	0.7

Table 4. Average FRR_0, FAR_0 and EER of experiment 4

FRR_0 (%)	FAR_0 (%)	EER (%)
1.9	1.5	0.9

Table 5. Average FRR_0, FAR_0 and EER of experiment 6

FRR_0 (%)	FAR_0 (%)	EER (%)
2.5	1.9	0.7

The experiment 3 was performed using the 40 genuine signature classes in conjunction with de 50 false signatures. The goal of this experiment is to determine joint discriminating power of the 3 feature vectors, the slope of the contour, the slope of the strokes and the correlation data based on (2), and the results can be observed on table 3.

$$p_{final} = p_{contour} \cdot p_{strokes} \cdot p_{correlation} \tag{2}$$

Based on the results of the experiment 3, it was verified that the classification error rates of handwritten signals using the equation 6 were considerable low. Thus, we decide to use the same classifier in experiments 4, 5, 6 and 7.

The experiment 4 was performed using the 20 handwritten symbol classes in conjunction with the 57 handwritten symbols that do not belong to any test class. The result of this experiment it is shown on table 4.

The experiment 5 was performed using the 20 handwritten symbol classes in conjunction with the 50 false signatures. In this experiment all the error rates were equal to zero, it means that it was possible to separate totally the handwritten signal classes from the false signature classes.

The experiment 6 was performed using 20 handwritten symbol classes and the 40 genuine signature classes in conjunction with the 57 handwritten symbols that do not belong to the test class and the 50 false signatures. The result of this experiment is shown on table 5.

The experiment 7 was performed using 5 classes chosen randomly from the Genuine Symbol Group (GSG). These 5 classes were the ones used as templates for the skilled forgeries. The result of this experiment can be seen on table 6.

In the experiment 8 we studied the performance of our method when it is applied to an identification process. Three tests were performed. In the first one, genuine signatures are compared to the Reference Information of all the persons that belong to the Genuine Signature Group (SG). As result of the comparisons, a set of distances was calculated (based on Equation 2), one distance measure for each of the classes that belong to SG. If the smaller distance calculated over the genuine signature of person k belongs to its own class, it is set as correct identification. In the second test the same methodology was applied, but only samples from the Genuine Symbols Group (GSG) were used. In the third test, samples from SG and GSG were mixed and the methodology was re-applied [5].

Table 6. FRR_0, FAR_0 and EER of skilled forgeries

						Average Error Rate
FRR_0 (%)	3	25	45	25	15	22.0
FAR_0 (%)	3	43	43	25	35	45.3
EER (%)	3	5	5	35	3	5.1

Table 7. Identification Performance

Database Universe	Correct Classification (%)
Signatures vs. Signatures	83.8
Symbols vs. Symbols	93.7
Entire database	87.1

The correct classification rate, that measures the number of correct classified handwritten signals, was the parameter used to evaluate the identification performance of the method. The results of the experiment 8 over these three tests are presented on table 7.

5   Conclusion

We proposed a method for personal authentication through unrestricted handwritten graphical signals, which can be viewed as generalization of personal signature authentication techniques.

We introduced the concept of graphical passwords, that they can either substitute or work in conjunction with classical textual passwords, improving the degree of security in computer systems logins.

This method is efficient, robust and can be easily implemented, requiring low CPU processing power and only five handwritten signal samples are need for enrollment.

From the performed experiments we conclude that this approach is efficient against random forgeries and also over skilled ones where the impostor knows previously the details of the genuine graphical password.

In this paper is proposed two feature extraction algorithms. The first is based on mathematical morphology operations to detect the slope of the contours and the strokes of the handwritten signal, and the second based on the pixel correlation between an image template and the image sample test.

The average error rates of 2.5% FRR_0 and 1.9% FAR_0 are in the inferior range of the set of values between 2 % and 5 % that Plamondon and Srihari in [6] affirm that are the error rates presented in most of the automatic personal identification systems based on signature images.

A simple statistical linear classifier was applied to separate the genuine signals from the forgeries. Thus, the proposed method take only few seconds to be trained, meanwhile other methods like Markov Models, Neural Network and Genetic Algorithms may take a very long time to perform the enrollment task [7][8].

Acknowledgement

References

[1]   Fairhurst M. C.,"Signature verification revisited: promoting practical exploitation of biometric technology", *Electronics and Communication Engineering Journal*, December, (1997) 273−280

[2]   L. L. Lee, Miguel G. Lizárraga, "Biometrics on internet: Security Applications and Services", Chapter book - Biometrics Solutions for Authentication in an E-World − Klewer Academic Publisher, (2002)

[3]   Miguel G. Lizárraga. "Um sistema biométrico de identificação pessoal via internet com ênfase em assinaturas estáticas", Ph. D. Thesis. UNICAMP, (2000).

[4]   L.L. Lee and M.G. Lizárraga, "An off-line method for human signature verification", ICPR (1996) 195 - 198

[5]   Mario E. Munich, Pietro Perona, "Visual Identification by Signature Tracking", *IEEE trans. On Pattern Analysis and Machine Intelligence*, Vol. 25. No. 2, (2003) 200 − 217

[6]   Plamondon, Rejean; Srihari, Sargur; "On-line and offline Handwriting Recognition: A comprehensive survey"; *IEEE Trans. on PAMI*, Vol 22, No. 1, (2000) 63 − 84

[7]   Cardot, H.; Revenu, M.; Victorri, B.; Revillet, M., "A static signature verification system based on a cooperative neural network architecture", *Int. Journal of Pattern Recognition and Art. Intelligence*, Vol 8, No 3, (1994) 679 − 692

[8]   Jain, A.; Duin, R.; Mao, J. "Statistical Pattern recognition: A review", *IEEE trans. On Pattern Analysis and Machine Intelligence*, Vol 22, No. 1, (2000) 4 − 37

On-Line Signature Verification
Based on PCA (Principal Component Analysis)
and MCA (Minor Component Analysis)

Bin Li[1], Kuanquan Wang[1], and David Zhang[2]

[1] Department of Computer Science and Technology
Harbin Institute of Technology
Harbin, China
`(lbn,wangkq)@hit.edu.cn`
[2] Department of Computing, The Hong Kong Polytechnic University
Kowloon, Hong Kong, SAR
`csdzhang@comp.polyu.edu.hk`

Abstract. On-line signature verification is still an active topic in the field of biometrics. This paper proposes a novel method based on PCA (Principal Component Analysis) and MCA (Minor Component Analysis). Different from the application of PCA in other fields, both principal and minor components are used to signature verification, and MC plays a very important role. Comparing with DTW and the discriminance of Euclidean distance, the method based on PCA and MCA is better. With 1215 signatures contributed by 81 signers of which numbers of reference signatures, genuine signatures and forgeries (skilled) are 5 respectively, the EER is about 5%.

1   Introduction

Signature verification is commonly used to approbate the contents of a document or to authenticate a financial transaction. With the rapid development and wide application of network, on-line signature verification becomes a hot topic in the field of Internet security. Many people engage in the research of on-line signature verification and a wide range of methods are presented in recent ten years.

Dr. Nalwa presented a very detail approach by relying primarily on the pen dynamics during the production of the signature instead of the detailed shape of a signature [1]. Jain et al. introduced an on-line signature verification system based on string matching and writer-dependent threshold [2]. Martens et al., Munich et al., and Yong et al. discussed an on-line signature verification system based on Dynamic Time Warping (DTW), which is originated from the field of speech recognition [3], [4], [5]. Other methods such as Hidden Markov Models (HMMs), Artificial Neural Networks (ANN) and Genetic Algorithm (GA) applied to signature verification are also introduced in some literatures.

In this paper, a novel on-line signature verification method based on principal component analysis (PCA) and minor component analysis (MCA) is introduced. We divide a signature into several segments according to predefined rules and search an optimal path in which two signatures can be well corresponded by using DTW.

Reference signatures are used to produce PC (principal component) and MC (minor component) with K-L transform. Signature verification will be based on PCs and MCs. Contrasted with other applications of PCA, both PC and MC are used in this paper, and MC plays a very important role to verification especially.

2   Signature Processing and Segmentation

In my system, a common and cheap tablet is used as a capture device. With a fixed sample frequency, a signature can be described by a series of points (x_i, y_i, t_i). An original signature captured by a general device is shown in Fig. 1.

We can divide a signature into two sequences, (x_i, t_i) and (y_i, t_i), corresponding to x- and y- coordinates. Since the noise coming from the capture device and hand-shake, it is necessary to normalize the signature and smooth with a sliding window. In this paper, the Gaussian function is used to smooth x- and y-curves of a signature. Curves about x- and y-axis after preprocessing are shown in Fig. 2.

Now we define a set of features to describe these curves of a signature. The sequence of inflexions in troughs and crests are detected and marked in each curve, and then a series of features for each pair of neighboring inflexions to describe a curve is defined as follows:

The length of these two neighboring inflexions in x-coordinates is:

$$l_x^i = x_i - x_{i-1} \tag{1}$$

The obliquity of the line connecting these neighboring inflexions is:

$$\theta^i = \arctan(\frac{y_i - y_{i-1}}{x_i - x_{i-1}}) \tag{2}$$

The position of the inflexion is:

$$p^i = \frac{x_i - x_1}{L_s} \tag{3}$$

The mean velocity between these two neighboring inflexions in x-coordinates is

$$V_{mean}^i = \frac{l_x^i}{T_i} \tag{4}$$

Deviation of Velocity is:

$$V_d^i = \sqrt{\frac{\sum_{j=1}^{M_i} (V_j^i - V_{mean}^i)^2}{M_i - 1}} \tag{5}$$

where (x_i, y_i) is the ith inflexion, x_1 is the x-coordinate of the first inflexion and L_s is the length of the whole signature in x-coordinates. M_i is the number of sample points of ith segment.

Now, we have a sequence of vectors to describe a whole signature about the x-axis.

$$H_x = (h_1, h_2 ..., h_N), \quad h_i = (l_x^i, \theta^i, p^i, V_{mean}^i, V_d^i) \tag{6}$$

The sequence of vectors about y-axis can be deduced in the same way.
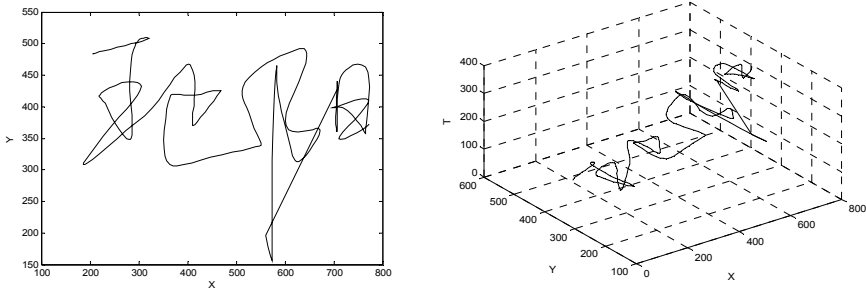
Fig. 1. An original signature is shown in 2-D and in 3-D with time information
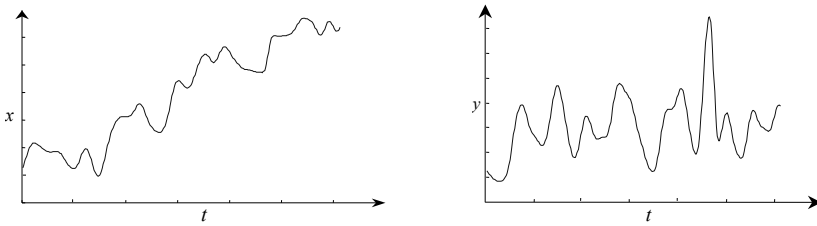
Fig. 2. Curves of a signature about x and y-axis after preprocessing

3   Flexible Matching and Stable Segments Extraction

As we know, the segment numbers of two signatures produced by a same signer are often different. So we must try to match the segments of a signature with another one correctly. Dynamic Time warping (DTW) is a technique that is well suitable for this matching. Using DTW, we get an optimal path in which the summation of distances of all corresponding segments between two curves is minimum. Yet, not all features of h_i (defined in formula 7) are suitable in this matching process, because dynamic features (V_{mean}^i and V_d^i) are more unstable than static features (l_x^i, θ^i and p^i). So we define a sequence S of static features for DTW.

$$S = (s_1,...,s_N), \quad s_i = (l_x^i, \theta^i, p^i) \tag{7}$$

To find the optimal path of two sequences $S^p = (s_1^p,...,s_I^p)$ and $S^q = (s_1^q,...,s_J^q)$ which describe the static information of two 1-D signature curves, a DTW algorithm is introduced as follows:

$$D_{i,j} = \min \begin{pmatrix} D_{i,j} \\ D_{i+1,j} + d_{2,1} \\ D_{i,j+1} + d_{1,2} \end{pmatrix}, \ 1 \le i \le I, \ 1 \le j \le J \tag{8}$$

where $D_{i,j}$ is the distance of static features between s_i^p and s_j^q, $d_{1,2}$ and $d_{2,1}$ are punishments. After DTW matching, the optimal path is recorded (Fig. 3).
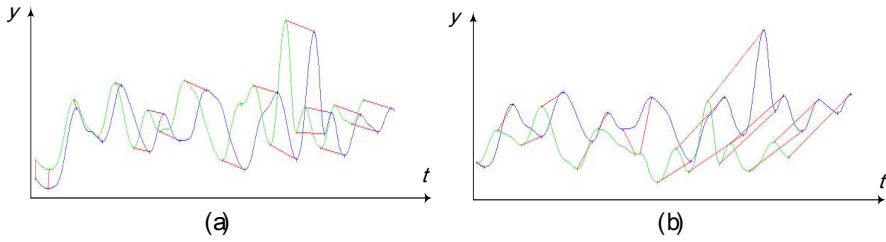
Fig. 3. Matching two signatures about y-axis by DTW. (a). Matching between two genuine. (b). Matching between a genuine signature and a skilled forgery

Since K-L transform needs all vectors with the same length, we must make the segment number of each signature be same. For this reason, after matching each signature with others in reference, we search those segments that appear in each matching paths and mark them as stable segments. The vectors of stable segments in each reference signature will be the same length. With both static and dynamic features of these stable segments, a feature vector for a reference can be described as:

$$H^m = (h_1^m, h_2^m ..., h_n^m), \ m = 1,..., M \tag{9}$$

where M is the number of reference signatures.

To a test signature, we also search its stable segments in the same way. If there is a matched segment in the test, it will be marked and both static and dynamic features of it will be added into a feature vector H', otherwise, a predefined null segment will be added into H'. After searching all stable segments in references, a feature vector H' of test signature whose length is the same with the reference will be produced.

4   Principal Component Analysis and Minor Component Analysis

Principal component analysis (PCA) is an essential technique for data compression and feature extraction, and has been widely used. In most applications of PCA, people usually throw away minor components and only care about principal components, as principal components contain most information of the reference data.

In this paper, both principal components and minor components are used. Even the minor component plays a more important role than the principal component. The feature vector $H = (h_1, h_2 ..., h_n)$ of each signature is reshaped to a 1-D vector s whose length is $N = 5 \times n$ (5 characters are used to represent a segment). Then M reference vector $(s_i^r - \bar{s})$, $i = 1,..., M$ are combined in a $N \times M$ reference matrix S:

$$S = (s_1^r - \bar{s}, s_2^r - \bar{s},..., s_M^r - \bar{s}), \ \bar{s} = \frac{1}{M} \sum_{i=1}^{M} s_i^r \tag{10}$$

where M is the number of reference signatures.

The eigenvector u_i and eigenvalue λ_i, $i = 1,...,5 \times N$ of the covariance matrix Σ of S can be deduced by K-L transform. The space constituted by fore $M - 1$

eigenvectors contains all the information of reference signatures. We call these eigenvectors with large eigenvalues as principal components (PCs) and eigenvectors with very small and zero eigenvalues as minor components (MCs).

We separate eigenvectors $U = \{u_i, i = 1,...5 \times N\}$ into two parts, one is U_P constituted of PCs and the other is U_M constituted of MCs.

$$U_P = \{u_i, i = 1,...M-1\}, \; U_M = \{u_i, i = M,...,5 \times N\} \tag{11}$$

Considering the effect of different eigenvector in U_P, large coefficients are given to the eigenvectors with small eigenvalues, and small coefficients are given to the eigenvectors with large eigenvalues. U_P is non-linearly transformed into:

$$\hat{U}_P = \{\frac{u_1}{\sqrt{\lambda_1}}, \frac{u_2}{\sqrt{\lambda_2}},..., \frac{u_{M-1}}{\sqrt{\lambda_{M-1}}}\}, \; u_i \in U_P \tag{12}$$

where u_i is the eigenvector with λ_i.

The reference s_i^r and the test s^t can be transformed into new space of $\hat{U}_P$.

$$\hat{s}_i^r = s_i^r \cdot \hat{U}_P, \; \hat{s}^t = s^t \cdot \hat{U}_P \tag{13}$$

where $\hat{s}_i^r$ and $\hat{s}^t$ is new vectors in the space of $\hat{U}_P$, and their length is $M-1$.

Now we turn to introduce MCA into signature verification. We give a concept: energy of a signature in U_M. Since all the information of references is contained in the space of U_P, the energy of references in the space of U_M is very small or zero. So we can judge a test signature as genuine signature or forgery by its energy in the space of U_M. The less energy of a test signature in the space of U_M is, the more similar with the references it is. The energy G in the space of U_M can be defined as:

$$G = \| (s^t - \bar{s}) \cdot U_M \| \tag{14}$$

where s^t is the test signature and $\bar{s}$ is the mean vector of references.

From above applications of principal and minor components, a distance to judge a test signature as genuine signature or forgery can be defined as:

$$Dis = \frac{1}{M} \sum_{i=1}^{M} \| s^t - s_i^r \| \cdot C_1 + \| (s^t - \bar{s}) \cdot U_E \| \cdot C_2 \tag{15}$$

where C_1 and C_2 are weights of these two parts which come from PCA and MCA.

The effects of PC and MC in signature verification can be understood as follows. In the feature space of reference signatures, some parts are stable which can be used to represent genuine signature well, and other parts are unstable which represent the inner varieties of reference signatures. With K-L transform and resizing the space of PCs non-linearly (formula 12), the inner variety can be restrained well. Furthermore, since the energy of reference signature in the space of MCs is very small or zero, the less energy of a test signature in the space of MCs is, the more similar with references it is. Taking above advantage, PCA and MCA can be well applied into on-line signature verification.

5   Experiment Results

The proposed method has been implemented and evaluated with 1215 signatures. Based on a database containing 810 genuine signatures and 405 skilled forgeries of 81 signers, the experiments were carried out. Each signer was asked to write his/her signature 10 times of which 5 signatures were used as references and the other 5 signatures were used for testing. Observed the whole producing course of genuine signature, some forgers were asked to imitate 5 signatures of each genuine signer.

To compare my method with normal DTW, in my experiments, we only use the local features of signature defined in section 2 and 3. The tradeoff curve using DTW and the discriminance of Euclidean distance is shown in Fig. 5.(a), and the tradeoff curve using DTW and PCA/MCA is shown in Fig. 5(b). The EER(equal error rate) with DTW and the discriminance of Euclidean distance is about 10%, and the EER of my method is about 5%.

6   Conclusion and Future Work

An on-line signature verification method based on DTW and PCA/MCA is proposed in this paper. Taking advantage of PCA and MCA, The stable and unstable information of reference signatures can be well analyzed and applied in signature verification. During this course, the unstable parts are restrained and the stable parts are impelled. The minor component plays a very important role, though it is often ignored in other applications.

It is still an open question to how signatures of a signer can be well divided into same segments, as K-L transform needs vectors with the same length. In future work, all kinds of feature comparison and the relation between PC and MC need to be analyzed detailedly in on-line signature verification based on PCA and MCA.
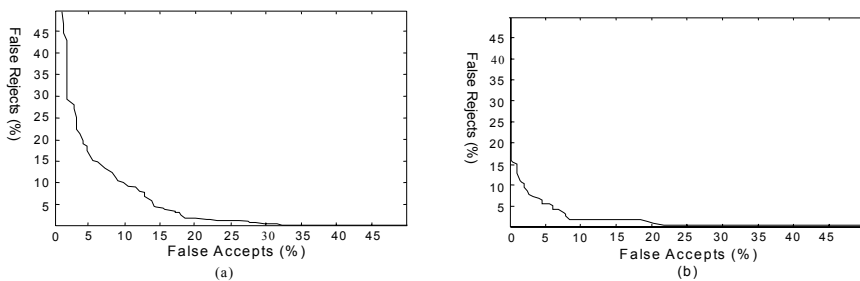
Fig. 4. Error tradeoff curves of experiments (a). Based on normal DTW and the discriminance of Euclidean distance. (b) Based on DTW and PCA/ECA

References

[1]   Vishvjit S. Nalwa. Automatic On-line Signature Verification, Proceeding of The IEEE, vol. 85, no. 2, Pages 215-239, February 1997.
[2]   A.K. Jain, F.D. Griess and S.D. Connell, On-line Signature Verification, On-line Signature Verification. Pattern Recognition, Pages 2963~2972, 2002
[3]   Ronny Martens and Luc Claesen. Dynamic Programming Optimization for On-line Signature Verification, Proceeding of 4th ICDAR '97. Pages 653-656, 1997.
[4]   Munich, M.E.and Perona, P. Continuous dynamic time warping for translation-invariant curve alignment with applications to signature verification Computer Vision. The Proceedings of the Seventh IEEE International Conference on, pages 108 -115, 1999
[5]   Jin Yong and Liu Jian, On-Line Handwriting Signature Verification Based on Elastic Matching of3 D Curve, Journal of Huazhong University of Science. and Technology, vol. 7 No. 5 pages 14-16, 1999

Biometric User Authentication on Smart Cards by Means of Handwritten Signatures

Olaf Henniger[1] and Katrin Franke[2]

[1] Fraunhofer Institute for Secure Telecooperation
Rheinstr. 75, 64295 Darmstadt, Germany
henniger@sit.fraunhofer.de
[2] Fraunhofer Institute for Production Systems and Design Technology
Pascalstr. 8–9, 10587 Berlin, Germany
franke@ipk.fraunhofer.de

Abstract. This paper describes a biometric method for user authentication on smart cards. Smart cards are chip cards with the ability for data processing directly on the card. They are not only usable for storing biometric reference data, but biometric user authentication methods can also be performed on card in order to protect security-relevant functions or data on the cards. The biometric data under consideration are handwritten signatures captured by means of a graphic tablet and a special pen. The feature-matching algorithm is a variant of dynamic time warping, taking the limited resources of smart cards into account. It is implemented as an operating prototype on two types of smart cards.

1 Introduction

Smart cards are chip cards with an integrated microprocessor chip. For smart cards providing security-relevant functions (like the creation of electronic signatures) or carrying values (like an electronic purse) or sensitive data (like medical data), the verification that the person who wants to use the card is the legitimate card holder should take place inside the card itself. On-card matching prevents the feigning of a positive result to the smart card. Moreover, on-card matching effects that the sensitive reference data remain safely stored inside the smart card. The function protected by on-card matching can only be used after successful user authentication. Fig. 1 illustrates the division of labor between the smart card service system and the smart card in case of on-card matching. On-card matching implementations are available for fingerprint biometrics, but so far not for handwritten signatures.

The major advantage of handwritten signatures over other biometric features is their high level of user acceptance. Handwritten signatures have long been accepted in many places as a means for authenticating persons. Moreover, a handwritten signature is regarded as evidence of a deliberate decision on the part of the signer because, in general, a signature is not delivered coincidentally or unintentionally.

There is a multitude of methods for analyzing on-line signatures, i.e. handwritten signatures captured by means of a graphic tablet and/or a special pen [1, 2]. Since smart cards possess only a limited memory capacity and their computing power falls far short of that of PC's, algorithms for on-card matching must be chosen carefully and must be adapted to these conditions. In this paper, we discuss the selection of a

signature recognition method that is suitable for implementation on smart cards. In a feasibility study, the selected method has been implemented as an operating prototype on two types of smart cards. The remainder of the paper is arranged as follows: In Section 2, the side conditions for the on-card matching implementation are described. Section 3 deals with design decisions. Section 4 contains first experimental results obtained with our prototype implementations. Section 5 summarizes the results and gives an outlook.

2   Side Conditions

2.1 Capture Devices

The on-card matching method should be widely independent of the capture device used for data acquisition, so that the smart card can be used in different environments.

Graphic tablets provide time series for the pen position (x and y coordinates). Some tablets also provide time series for the pen-tip pressure and the pen orientation, which may support signature matching. However, because not every kind of tablet provides this information, pen-tip pressure and pen orientation are left out from our on-card matching method. Even if different tablets provide time series for the pen-tip pressure, their uniform calibration is difficult.

Since the signature dynamics of a person vary slightly from signature to signature and signatures must be accepted anyway within a certain range of tolerance, for signature capture an approach "as accurate as necessary" and not "as accurate as possible" is needed. We assume a resolution of at least 1,000 dpi and a sample rate of 100 per second to be sufficient not to lose substantial information about on-line signatures. The time series produced should be equidistant, so time stamp information need not be recorded.

The graphic tablets deployed should record the pen position not only if the pen touches the tablet (pen-down strokes), but also if the pen is close-by above the tablet (pen-up strokes) since these movements may bear signer-specific information.
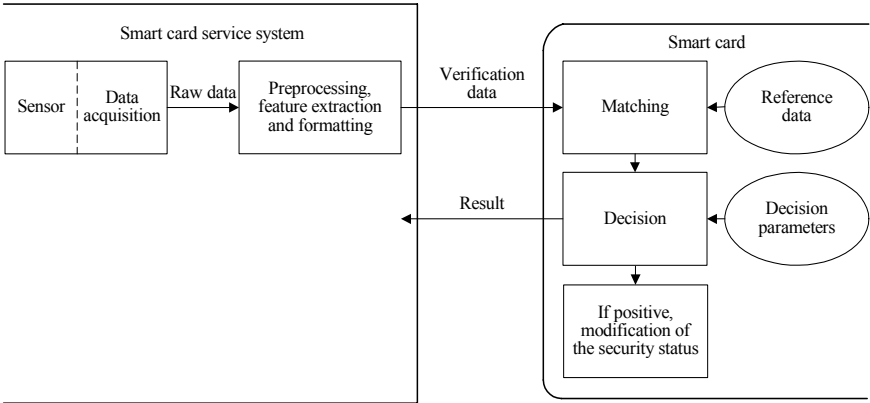
Fig. 1. Biometric user authentication with matching on card

2.2 Implementation Platforms

Java Cards. Java cards are smart cards with an interpreter (Java Card Virtual Machine) [3] for the execution of processor-independent byte code. Code development for Java cards is based on a subset of Java and Java development tools.

Java cards provide only limited memory space and their computing speed is limited because the Java byte code is interpreted at run-time. On Java cards, only a subset of the Java language is available. Java cards support the data types *Boolean*, *byte*, *short*, and optionally also *int*; the data types *char*, *double*, *float*, and *long* are not available. Only one-dimensional arrays are supported. By default, only basic arithmetic operations and no mathematical libraries are available. There is no garbage collection and no *finalize*(). Objects and arrays once created cannot be deleted; their storage location remains occupied. Therefore, all necessary objects and arrays have to be created at the installation of a Java card applet and be reused later. Dynamic loading of classes is not supported; all necessary classes must be brought onto the Java card at production time or during the installation of a Java card applet.

We selected rather powerful Java cards as implementation platform (CPU word length: 16 bit, 2 Kbytes RAM, 64 Kbytes ROM, 32 Kbytes EEPROM, default clock rate: 3.5712 MHz) [4]. However, their computing power falls far short of that of today's PC's.

The shortage of RAM causes considerable problems. There would be sufficient EEPROM available to use it also as main memory; however, writing into EEPROM cells is substantially slower than writing into RAM, and the number of write cycles for EEPROM is limited.

Java cards are well suited for the rapid development of prototype smart card applications. That's why the signature recognition approach described in this paper has first been implemented on Java cards. However, their drawback is that the interpretation of Java card applets by a Java Card Virtual Machine is rather slow.

Native-Code Cards. Program execution on smart cards running native code is considerably faster than on Java cards. Therefore, native-code cards would be a better choice for developing an end-product rather than a prototype. However, the memory space on native-code cards is as limited as on Java cards. Code development for native-code cards is based on cross-compiling programs in higher programming languages.

The signature recognition approach described in this paper has also been implemented as an operating prototype on a smart card running native code, viz. on a Funcard 2. The Funcard 2 is available e.g. from hobby electronics shops and is well suited for applications requiring rapid development and small volume roll-outs. It is a programmable multi-chip smart card with an AVR-based RISC microcontroller (Atmel AT90S8515A: CPU word length: 8 bit, 512 bytes RAM, 8 Kbytes programmable flash, 512 bytes EEPROM, default clock rate: 3.5712 MHz) [5] and a separate 8 Kbytes EEPROM chip.

3   Design of the On-Card Matching Method

3.1 Signature Recognition Approach

Methods for analyzing on-line signatures can be roughly classified into methods for the statistical, spatial, temporal, and spectral analysis and combinations thereof. The statistical analysis is too weak alone. The spatial analysis considers features perceptible in the shape of on-line signatures. For a temporal analysis, the features to be compared are time series, i.e. sequences of values at successive points in time. The spatial or temporal analysis could be implemented on smart cards. For a spectral analysis today's smart cards do not have sufficient resources.

With standardization of the feature data format in mind, it stands to reason to use directly the time series for x and y, possibly supplemented with time series for velocity, acceleration, pen orientation, etc., as biometric features. Therefore, we have chosen the temporal analysis of on-line signatures as on-card matching algorithm.

3.2 Preprocessing, Feature Extraction, and Formatting

Preprocessing the Captured On-Line-Signatures. The goal of preprocessing is to suppress insignificant information that is expression of random variation, but not of individual signature dynamics, and to even out differences between the data captured with different graphic tablets. Preprocessing steps include translation and rotation of the signature, removal of the linear component from the x-time regression line, normalization and scaling.

If the time series for the x and y coordinates had a normal (GAUSSian) distribution, then about 70% of the normalized x and y coordinates lay in the interval between -1 and 1, almost 100% of them lay in the interval between -4 and 4. Due to the lack of floating-point arithmetic on Java cards, floating-point numbers are not suitable for on-card processing. Therefore, the x and y coordinates are scaled such that they each can be coded in a signed byte, i.e. as an integer in the interval from -128 to 127. In order that almost all values lie in the desired interval, the normalized x and y coordinates are scaled by multiplying with $128/4 = 32$ and rounding off towards the nearest integer. In case a value lies outside the desired interval after scaling (which is possible, but very improbable if the values were normally distributed), it is replaced by -128 or 127, respectively. Rounding off to integers entails a quantization error. We assume it to be negligibly small compared to the random variation among signatures of the same person.

Fig. 2 (a) shows pen-down and pen-up strokes of a sample signature. Fig. 2 (b) shows the preprocessed time series of the x and y coordinates of the sample signature.
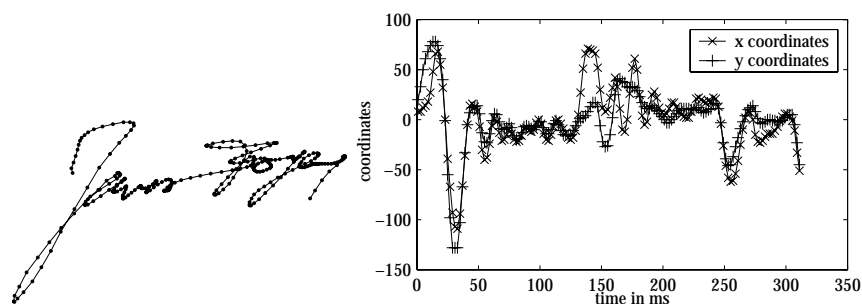
Fig. 2. (a) Sampled signature; (b) Preprocessed pen-position time series

Table 1. Format of biometric data objects for signature feature data

Tag	Length	Value
5f 2e (Card Holder Biometric Data [6])	1–3 bytes, coded according to the ASN.1 Distinguished Encoding Rules [8]	OCTET STRING $x_1 y_1 ... x_n y_n$

Feature Extraction. The features to be compared are the preprocessed time series of the x and y coordinates and time series for the velocity in x and y direction derived from the time series of the x and y coordinates by forming difference sequences. If the derivation of the difference sequences takes less time on the card than their derivation outside the card plus the transmission to the card, then the difference sequences should be derived on card.

Format of the Signature Feature Data at the Smart-Card Interface. Reference data are transmitted to the smart card in the command data fields of CHANGE REFERENCE DATA APDU's; verification data are transmitted in the command data fields of VERIFY APDU's. Both, reference data and verification data are required to be packed in a biometric data object [6, 7]. The structure of biometric data objects used in the prototype implementation is described in Table 1.

In general, only 255 bytes can be transmitted in the data field of a smart-card command. Because the feature data may be longer than this, we use command chaining for the APDU's [9] and send the data belonging to a signature in subsequent APDU's.

3.3 Feature Matching on Card

The feature-matching algorithm compares the verification data presented by the user with the reference data stored on the card and determines their "distance" as a measure for their dissimilarity. If the distance is larger than a threshold value, the presented signature is regarded as a forgery; otherwise, it is regarded as a genuine signature. The threshold depends linearly on the length of the reference data. The reference signature is one signature selected from several signatures captured during the enrolment phase.

The "distance" of two signatures can be determined by nonlinear time warping of the time series of the signature to be verified to the time series of the reference signature [2, 10, 11]. Nonlinear time warping is achieved by dynamic programming. The goal is to determine a mapping between the time axes of the signatures to be compared (distortion path) minimizing the distance of the two signatures and observing certain side conditions. The side conditions are that the start and end points of verification and reference data must be mapped one on the other, that the temporal ordering of the sample points must be maintained, and that no sample point be omitted. The distance between verification data and reference data is the sum of the local distances along the optimal distortion path. The local distance between the feature vectors of verification data and reference data (consisting of x and y coordinates as well as velocities in x and y direction at a sample point) can be defined in different ways. On the smart card we use the absolute value norm of the difference vector as a distance measure because its computation requires less resources than more sophisticated distance measures.

Dynamic programming algorithms are described in the literature (e.g. [11]) and are not repeated here. Dynamic programming is a time-consuming procedure. The computing time is reduced by looking for the optimal distortion path only within a limited band (Sakoe/Chiba band [12]).

4   Experimental Results

For testing purposes, the prototypes reveal the distance values of each matching attempt. In normal operation, the distance values should not be revealed to avoid hill-climbing attacks (where an attacker systematically modifies the verification data to obtain smaller and smaller distances until the decision threshold is met).

The false acceptance rate (FAR, probability that an unauthorized user is falsely accepted) and the false rejection rate (FRR, probability that a legitimate user is falsely rejected) can be estimated experimentally with a certain statistical significance using large test databases [13]. We carried out a preliminary technology evaluation of our prototype implementations by testing against a signature database collected within our research group. The signatures were collected during several sessions from 5 colleagues, all male right-handers under 40 years of age. The signature data were captured on a WACOM tablet. The database contains a total of 50 genuine signatures and 69 forgery attempts. For a statistically more significant technology evaluation, a larger database will be needed.

For the forgery attempts, the impostors had the original signatures to be forged available on paper (a quite realistic scenario) and were allowed to produce forgeries to the best of their ability. They were allowed to practice the signatures to be forged, to look at the original while forging, and even to retrace the original.

Fig. 3 summarizes the preliminary performance results in a receiver operating characteristic curve [13] plotting the relative frequency of correct acceptances over the relative frequency of false acceptances at varying threshold values. The equal error rate (where FRR equals FAR) lies at about 8%.
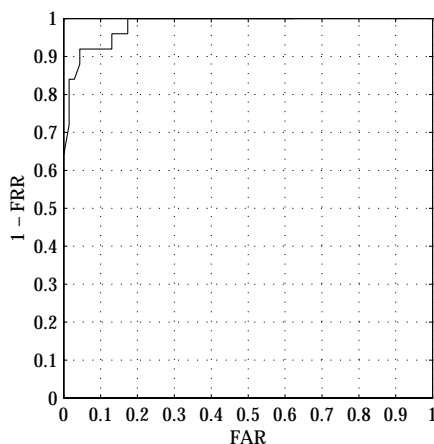
Fig. 3. Receiver operating characteristic curve

The time needed for transmitting verification data to the card and for on-card matching grows linearly with the size of the verification data. For the shortest on-line signature in the database consisting of 70 sample points (i.e. writing it took 0.7 seconds), the computing time is about 7 seconds on the Java card and about 2.5 seconds on the native-code card. For an on-line signature consisting of 250 sample points, the computing time is about 25 seconds on the Java card and about 8.5 seconds on the native-code card. On the native-code card computing takes about one third of the time it takes on the Java card. For practical applications, less waiting time would be desirable.

5   Summary and Outlook

Biometric user authentication on smart cards by means of handwritten signatures is feasible. We have implemented first prototypes of an on-card matching algorithm for handwritten signatures. The challenge consisted in implementing the feature-matching algorithm using only the limited resources available on smart cards.

The approach we selected for implementation on smart cards is the temporal analysis of signatures. The features to be compared are the time series of the x and y coordinates, which are captured by a graphic tablet and a special pen and pass through some preprocessing steps, and the time series for the velocity in x and y direction, which are derived from the time series of the x and y coordinates. By means of dynamic programming, a nonlinear time warping of the signatures to be compared is performed and their distance is determined as a measure for their dissimilarity.

A more thorough statistic evaluation of the attainable error rates based on larger signature databases is pending. An improvement of the performance is expected from dissecting the signatures into pen-down and pen-up strokes, as discussed in [11]. The recognition performance may be further improved by taking into account more features, like pen orientation. The computing time needed for on-card matching will decrease as smart cards become more and more powerful.

Acknowledgements

References

[1]   R. Plamondon and G. Lorette. Automatic signature verification and writer identification – The state of the art. *Pattern Recognition* 22 (1989), pp. 107–131

[2]   C. Schmidt. *On-line Unterschriftenanalyse zur Benutzerverifikation*. RWTH Aachen, Germany, PhD Thesis, 1998, in German

[3]   *Java Card 2.1.1 Virtual Machine Specification*. Sun Microsystems, Revision 1.0, May 2000

[4]   *Reference manual of the Java card operating system Sm@rtCafé 2.0*. Giesecke & Devrient, edition 12/01, 2001

[5]   Atmel 8-bit AVR microcontroller with 8 Kbytes in-system programmable flash – AT90S8515. Datasheet, 2001

[6]   Information technology – Identification cards – Integrated circuit(s) cards with contacts – Part 6: Interindustry data elements. International Standard ISO/IEC 7816-6

[7]   Information technology – Identification cards – Integrated circuit(s) cards with contacts – Part 11: Personal verification through biometric methods. International Standard ISO/IEC 7816-11

[8]   Information technology – ASN.1 encoding rules – Part 1: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER). International Standard ISO/IEC 8825-1

[9]   Information technology – Identification cards – Integrated circuit(s) cards with contacts – Part 4: Interindustry commands for interchange. International Standard ISO/IEC 7816-4

[10]   J.B. Kruskal. An overview of sequence comparison, In D. Sankoff, J.B. Kruskal (eds.), *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Addison-Wesley, 1983, pp. 1–44

[11]   B. Wirtz. *Segmentorientierte Analyse und nichtlineare Auswertung für die dynamische Unterschriftsverifikation*. TU Munich, Germany, PhD Thesis, 1998, in German

[12]   H. Sakoe and S. Chiba. Dynamic programming optimization for spoken word recognition. *IEEE Trans. Acoustics, Speech and Signal Processing* 26 (1980), pp. 623–625

[13]   A.T. Mansfield and J.L. Wayman. *Best practices in testing and reporting performance of biometric devices*. Version 2.0, Aug. 2002

Writer Identification Method
Based on Forensic Knowledge

Marino Tapiador[1] and Juan A. Sigüenza[2]

[1] IBM Global Services and Universidad Autónoma de Madrid
Escuela Politécnica Superior, Spain
marino_tapiador@es.ibm.com
[2] Universidad Autónoma de Madrid
Escuela Politécnica Superior, Spain
jalberto.siguenza@ii.uam.es

Abstract. Police corps have been extensively used Forensic techniques to perform criminal identification. One of these techniques is questioned document examination. Forensic document examiners can identify individuals in big populations using a classification of the forms of manuscript characters, i.e. a handwriting formulation. This paper presents a method that defines a handwriting formulation that allows high identification accuracy minimizing the amount of data used and the sample size. Thus the method improves the query performance in a writing specimens database and it reduces the storage requirements. Experiments with results of 100% accuracy on the identification of 20 criminals in a real forensic database are presented.

1 Introduction

Questioned document examination is a usual technique used by police corps around the world since the early 1600's. Different methods have been developed by forensic scientists in order to verify or identify criminals using multiple manuscript features. Methodological aspects of handwriting identification have been intensively studied time ago by forensic document examiners as described in [1]. The individual's handwriting, and the identification of authors of questioned handwritten documents has great importance on the criminal justice system from two points of view: verification and identification. The work described in this paper was focused on off-line identification procedures (due working with confiscated manuscripts) and it was developed in collaboration with a forensic laboratory of a Spanish police corps.

Handwriting identification can be included as a particular kind of dynamic biometrics ([2]) where the shapes and relationships of writing strokes are used as biometric features for authenticating an identity i.e. we try to measure the individual's behavior. These features are used in the case of forensic handwriting analysis to identify a criminal in a group of suspects.

Several works related to off-line writer identification can be found in biometrics e.g. [3]-[5], and the most important text-dependent study about handwriting individuality -due the great volume of samples used: 1000 volunteers- that we are aware was performed by Srihari et al. [3]. The identification algorithm used was the traditional Nearest Neighbor (NN) rule [6] and best accuracy was achieved using

character level features (GSC, Gradient-Structural-Concavity features) using 8 characters of a fixed word in a fixed text (text-dependency). That work revealed a decreasing performance from 98% accuracy with 2 writers to 72% accuracy using 1000 writers (1000 different classes).

Other important research was developed by Said et al. [4], with the main objective of developing a text-independent handwriting identification system. It used texture analysis for writer identification from non-uniformly skewed handwriting images. An identification accuracy of 96% was obtained with 150 test documents.

Finally, another relevant work was done by Cohen et al. [5], due to its focus on character examination. This work was mainly centered on B-Spline curve representation and as an example, it covered a couple of use cases where one of them was writer-screening identification related to affine invariant matching and classification of 2-D curves. The author's experiment consisted of 10 writers, each of which has written the characters of the alphabet 10 times. A 100% classification rate was achieved using the full alphabet i.e. 26 characters (English).

Considering all this previous references, the goal of the work described in this paper was to design a method in order to build an automatic forensic system with as high accuracy as the work of Cohen [5] or higher, and to use it for criminal identification i.e. not using the friendly samples described by Cohen, and with a reduced sample size which enables efficient search and storing of samples in a large-scale forensic database.

2   Our Method

2.1 Acquiring the Forensic Knowledge

The writer identification method used by the forensic group participating in this work is based on manually reviewing an archive of handwritten documents confiscated to several criminals. Therefore forensic document examiners cannot have a perfect full-alphabet sample from suspects, they usually have to work with very limited handwriting samples e.g. a couple of lines in a notebook confiscated, a photocopy of a contract, or so forth. The method used by the experts in order to compare two different questioned documents consist of generate a document classification based on the forms of the characters (in our terms, formulation). This method increases the manual identification speed and is based on several kinds of relevant characters, so each manuscript can be formulated in terms of the kind of characters it contains. For example, if one of the characters present in the formulation is the letter "K" for this class of letter the forensic expertise says there are only three main shape variations: these subclasses have associated three formulation codes -"K1", "K2", and "K3"- and thus if the document contains the "K" letter it also will have in its formulation one of these three formulation codes. Collecting all the formulation codes of characters contained into the document, we will have the formulation that label that documents e.g. let X be a questioned document, the formulation could be $F(X)=\{A2,B4,D2,G4,K3,M2,a3,f1,r3,z1\}$. Having the questioned documents formulated in these terms, writer identification consist of formulating the questioned document and comparing the list of codes for each document in order to find the more similar document of a registered criminal.

Because each handwritten document could be labeled with this special formulation (list of letter codes), a database was built using the thousands of documents in the archive and registering just the formulation associated to each document. The idea was to be able to perform several queries over this database using the formulation in order to find documents with the same or similar formulation originated by new questioned documents. But this type of database revealed to be insufficient when the formulation process started due to the problem of subjective interpretations. Different experts can classify the same character sample in different classes by visual analysis: for instance, if an expert E1 formulates a questioned document from author X and searches it in the database where there is the genuine document from author X formulated by another expert E2, the document could not be found because the formulations can be different as they were provided from different experts. For example, figure 1 shows the case of an 'A' character sample from an anonymous writer that could be classified in a different category by different experts. After visual examination, expert 1 could decide to formulate the 'A' as type A4 i.e. 'A with centered tilde and curved head', but expert 2 could decide to classify it as type A7 i.e. 'A with centered tilde and squared head'.
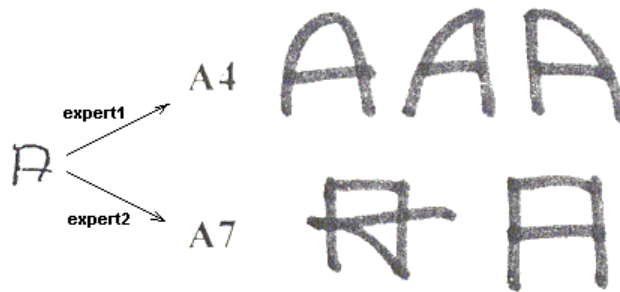
Fig. 1. Different experts can formulate in a different way

Table 1. Letter Formulation

Uppercase	#Variations	Lowercase	#Variations	Digits	#Variations
A	8	a	7	4	4
B	6	b	5	8	4
D	6	d	6		
E	6	f	7		
G	3	g	4		
J	6	m	4		
K	3	p	4		
M	5	q	4		
Q	2	r	4		
R	5	t	5		
		z	2		

Our method does not consider the subclasses included in the formulation for each kind of letter in order to avoid these interpretation problems but it uses the forensic

knowledge about which are the letters with more shape variation, and for this reason the more valuable for classification. The forensic knowledge was condensed in a new formulation by reviewing an archive of 5,000 criminal documents (manuscripts). The forensic examiners summarized the letters with more shape variation in this formulation shown by table 1.

2.2 Data Collection System

The system works with digitized images of the original confiscated/authenticated documents or with photocopies of them (A4 size), provided by the Spanish forensic lab collaborating in the project. The documents are scanned into digital form using a resolution of 300 DPI in a PC with Microsoft Windows. The full document image is stored in the system's hard disk in bitmap image format (BMP) using 24 bpp. With the digitized full-document ready from an individual, the next stage is the character segmentation. A standard digital image-manipulating tool is used, extending it with an automatic action (macro): the system operator has to draw a character selection using the computer mouse, and to label it. All these letter samples are stored in a database where there are samples and users ('questioned' or 'genuine'). Because the forensic procedure we were attempting to automate was based just on character examination we included in the system the character level features used in the study of Srihari [3], and we designed an intra-character level comparative study that will be discussed later in this paper.

Next stage is the normalization process: the bitmap image is converted to gray scale, and an average spatial filter is applied to smooth the character image -noise elimination-. Finally the character is binarized to black and white format using the Otsu's binarization algorithm [7] and the margins are dropped by means of calculating the bounding box of the image. The feature extraction process is applied to the resulting image from the normalization. The digital features are registered in the system's database. The features are GSC and geometric (character level). GSC is a group of features that consist of gradient, structural and concavity features, and the geometric features are related to several geometric properties of the letter shape. These features are the total number of black pixels, the height-width ratio, the centroid-height ratio, the centroid-width ratio, and a 9-spatial sensor (see [3]).

All these features are compiled in a total binary feature vector. This vector is stored into the system's database and is used during the identification process in order to decide the similarity between character samples of individuals.

2.3 Identification System

The data collection module is used to capture character samples only for the letters contained in the formulation considered. Maybe not all the letters in the formulation can be included in the database because samples come from criminal documents confiscated in real life, and only some letters are in them.

The identification method used by the system is the traditional Nearest Neighbor (NN) algorithm [6]. This algorithm has been applied in a similar way as in the Said work [4] using the Euclidean distance in order to identify a particular unknown individual. Considering a total of N genuine documents, given all the handwriting

images (character samples) in a document D_i , i N for a genuine individual in the database, the total binary feature vector is computed as described in section 2.2 for each character image. Each handwritten genuine document D_i is thus described by the set of character vectors c_j it is made of:

$$D_i = \{c_j, j \quad card(D_i)\}, i \quad N. \tag{1}$$

New character samples from a new questioned or unknown individual follow the same process and their feature vectors are also computed. Let be $L(c_j)$ the letter corresponding to the character sample of c_j , a distance measure DIS between a questioned handwritten document Q and a genuine document D_i for a particular type of letter $C \in \{Formulation\}$ in the database can be defined according to the following relation (2). It means that to identify the unknown user, all his/her character samples registered are considered, and for each character its feature vector is compared with all the characters from genuine users in the database. Only the same kinds of letters are compared ('A' vs. 'A', 'b' vs. 'b' and so forth in the formulation) and in general they come from different texts due the text-independence requirement of our experiments (it means they are characters not included in the same word).

$$DIS(Q, D_i) = \quad F^{-1} \Sigma_C^F \ M^{-1} \Sigma_p^M Min(\ dis(x_p, y_q) \ ), \text{ where } L(x_p) = L(y_q) = C$$
$$x_p \in Q_C \ , \ y_q \in D_{iC} \ , \ M = card(Q_C) \ , \ C \in \{Formulation\}, F = \tag{2}$$
$$card(\{Formulation\})$$

Comparing two binary feature vectors consists of generating a distance vector with a questioned character and a genuine character. The distance vector consists of several components -sub distances- for the different types of features. Each component is computed using the Euclidean distance $dis(x_p, y_q)$ between the corresponding feature vectors of the unknown and genuine character images. Given the binary feature vectors of two characters A and B, the distance measure is:

$$dis(V_a, V_b) = \sqrt{ \left(\Sigma_i^n \ (V_a^i - V_b^i)^2 \right) } \quad , n = card(V_a) = card(V_b) \tag{3}$$

Therefore, two documents will be all the closer as this measure will be close to zero. The writer of document Q will be determined as the writer of the closest document in the database:

$$Writer(Q) = Writer(\ Arg_i \ min \ (\ DIS(Q, D_i) \ ) \ ), D_i \ \text{for } i \quad N = card(\{database\}) \tag{4}$$

3 Experiments

Using the method previously described a forensic database was created using documents confiscated to real criminals or authenticated in presence of a policeman. It means the documents were acquired in real forensic conditions. This is an important point compared with the experiments of the other works cited in the introduction of this paper: in that works the writing samples were obtained with the collaboration of volunteers under controlled conditions.

Considering only characters in the formulation proposed in table 1 allowed capturing and digitizing a reduced volume of 1,400 character samples. A total of 20 genuine individuals and 20 questioned individuals were created in the database

associated to the samples. For a particular individual, the tool operator (forensic expert) digitized an authenticated document and created two users in the database: a genuine user and a questioned/unknown user. The 80% of the samples are used to register the genuine individuals and the other 20% is used in order to test the identification accuracy of the system.

The identification method described in section 2.3 was used with all the questioned users registered in the database and the experimental results are summarized in figure 2. This figure describes a comparison between the identification accuracy obtained for the different types of character level features used in the process. The graph shows in a visual way the ordered accuracy levels (%) with each kind of features associated to each graph block.

4   Conclusions

This writer identification system shows a 100% accuracy with 20 individuals using only gradient features (figure 2), better than the 96% accuracy for 10 individuals provided by Said et al. [4], as the more important reference for a text-independent identification system (not considering the basic experiments of Cohen [5] with full-alphabet 'ideal' samples and with the computational complex B-splines technique). Compared with the text-dependent experiments of Srihari [3], the accuracy is also better because in that work accuracy has already gone down to 96% with only 10 individuals in the experiments.

The analysis of our results suggests that the basic idea of the formulation technique described in the paper can be used in order to reduce the sample size and also in order to have a reduced number of vectors in the database. For the formulation presented in table 1, the identification method allows to use gradient feature vectors of 192 bits instead of GSC feature vectors of 512 bits: a 62% information reduction. These are significant advantages in writer identification systems in the area of forensic or criminal identification where the questioned individuals belong to huge populations that imply searching and storing data in large-scale databases.

Future experiments will be focused on increasing the database volume to get more accurate conclusions; due using only 20 individuals is a limited amount of data.

Acknowledgements

1 = gradient;   2 = structural;   3 = all;
4 = U/D/L/R/H;   5 = large stroke;   6 = geometric;
7 = coarse pixel;
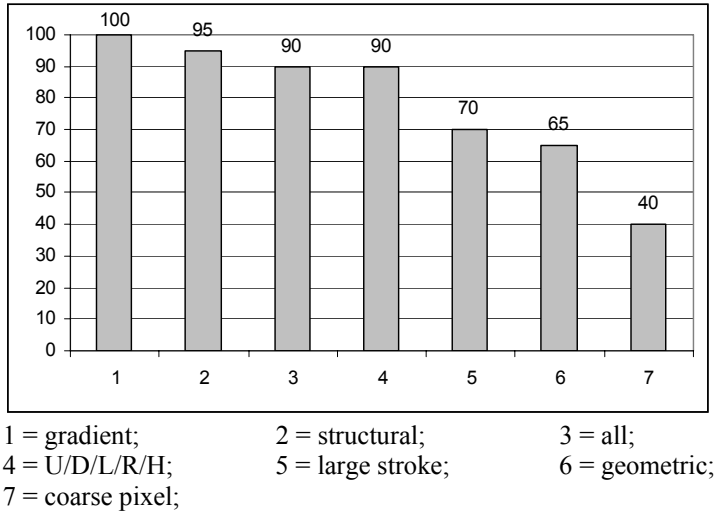
Fig. 2. Identification Accuracy (%) by Type of Features

References

[1]   R. A. Huber and A. M. Headrick, *Handwriting identification: facts and fundamentals*. Ed. CRC Press (1999).

[2]   A. K. Jain, R. Bolle, and S. Pankanti, *Biometrics. Personal identification in networked society*. Ed. Kluwer Academic Publishers (1999).

[3]   S. N. Srihari *et al*., "Handwriting identification: research to study validity of individuality of handwriting and develop computer-assisted procedures for comparing handwriting," Tech. Rep. CEDAR-TR-01-1, Center of Excellence for Document Analysis and Recognition, University at Buffalo, State University of New York, Feb. (2001), 54 pp.

[4]   H. E. S. Said, G. S. Peake, T. N. Tan and K. D. Baker, "Writer identification from non-uniformly skewed handwriting images," in *Proc. 9th. British Machine Vision Conference*, (2000) pp. 478-487.

[5]   F. S. Cohen, Z. Huang and Z. Yang, "Invariant matching and identification of curves using b-splines curve representation," *IEEE Transactions on Image Processing,* vol. 4, no. 1, pp. 1-10, Jan. (1995).

[6]   R. O. Duda, P. E. Hart, *Pattern Classifications*. 2nd. Ed. John Wiley & Sons (1973).

[7]   N. Otsu, "A threshold selection method from gray-scale histogram," *IEEE Transactions System, Man and Cybernetics,* vol. 9, pp. 62-66, Jan.(1979).

A Window-Based Hybrid Signature Verification System

Alessandro Zimmer[1,2] and Lee Luan Ling[1]

[1] UNICAMP - Universidade Estadual de Campinas. CEP 13081-970, Campinas, SP, Brazil
`lee@decom.fee.unicamp.br`

[2] UNICENP – Centro Universitário Positivo. CEP 81280-330, Curitiba, PR, Brazil
`azimmer@pobox.com`

Abstract. In this work we present a hybrid handwritten signature verification system where the on-line reference data acquired through a digitizing tablet serves as the basis for the segmentation process of the corresponding scanned off-line data. Local windows are determined over the image through a self-adjustable learning process and are used to focus the feature extraction step. The positions of the windows are determined according to the complexity of the underlying strokes given by the observation of a handwritten reproduction model. Local feature extraction is bounded by the windows formed and it is used with global primitives to feed the classifier. The overall performance of the system is then measured.

1   Introduction

In any Signature Verification System, the feature extraction procedure represents a major challenge on the verification process. Global features, such as the overall direction of the signature, dimensions, pixel distribution, etc. are usually not enough to discriminate forgeries, on the other hand, significant local features are extremely hard to pinpoint. In order to focus the feature extraction process, great efforts were made toward the robust extraction of basic components called "strokes" from the original skeleton of the words [1] [2] [3]. The exact definition of such entities varies according to the segmentation method used as well as the kind of data extracted (dynamic or static) and sometimes can even be based on the semantics of the languages on which data is written.

In this work the local features are extracted from windows formed around the individual strokes where the definition of a stroke is done inspired by a psychomotor handwritten generation model [4]. This model states that complex neuromuscular movements, such as the ones generated by the arm-forearm-hand system can be decomposed into simple impulses, or strokes. Depending on the temporal superposition degree of such impulses we can classify the resultant strokes according to their complexity. From this observation we can consistently label and consequently segment any sequence of points produced by ballistic trained movements such as the ones used during the handwriting process of the signature.

The windows (or foci of attention) used during the local feature extraction processes will be located over simple stroke regions, as suggested by the handwritten

generation model studied. The size of those windows will be dynamically determined during the learning process, based on an intraclass minimization criterion.

In order to be able to determine the precise location of the individual simple strokes we have to be able to follow the contour. This will be done with the help of the on-line dynamic data of the signature (position versus time) acquired by a digitizing tablet.

On-line systems generally require the presence of the author during both the acquisition of the reference data and the verification process limiting its use to certain kind of applications [1] [2]. In this work a hybrid on/off line signature system was used, limiting the use of a digitizing tablet to the acquisition of the reference data. The verification process can be done directly over the desired document, without necessarily the presence of the author of the signature. The on-line reference data serves as a basis for the localized feature extraction process and for segmenting the off-line test data to be acquired during the verification step.

The system can be divided into the acquisition and training module (on/off-line), and the verification module (off-line). The first one is responsible for the processing of the on-line reference data, generating the thresholds needed during the verification step. The second one inputs the test image to the system, extracting similarity measurements between the reference and the test data, reaching a verdict about the authenticity of the signature.

2   Segmenting by the Stroke Complexity

Complex movements such as the ones used during the reproduction of the handwriting can be viewed as the result of a system of simple push-pull impulses produced by the neuromuscular networks involved [4]. The segmentation of the signatures will be done based on the decomposition of the individual movements forming each of the strokes producing the signature.

2.1   Modeling the Handwriting

The delta-lognormal model studied [4] sees the handwriting as a temporal vectorial superposition of individual strokes, where each of them is the result of a synchronous activation of a lognormal (asymmetric bell-shaped) impulse response system representing a synergy of agonist and antagonist neuromuscular movements that will determine, for a sequence of n strokes, the speed and the position of the writing instrument through time.

A simple stroke is characterized by a constant curvature, an initial direction and a corresponding speed profile. By the determination of the curvature profile of the signature and searching its constant curvature regions it is possible to separate simple from complex strokes in a consistent fashion, completing then the segmentation of the original dynamic signal [3].

Constant curvature regions are normally straight or semi-straight elements (segments with a little curvature), where the pen attains maximum speed due to the lack of the restraint antagonist muscular impulse. In order identify and segment those regions we will label the local maxima of curvature and execute a research between

those points. The maxima can be easily determined by the use of a thresholding process.

2.2   Preprocessing

First of all the sampled data from the digitizer is filtered in order to separate the individual components (sections where the pen touches continuously the digitizer), and to eliminate spurious points of the trace such as the ones generated when the pen rests at the same position during the sampling cycle as well as points produced where the pen moves away from the surface of the tablet. The one pixel width ideal skeleton is then formed through multiple interpolations by splines of the filtered data.

In order to reduce the digitalization noise we have to assure that the pixels are 8 connected (the pixels should be connected to only one of its eight possible neighbors). From this sequence of generated points we can proceed with the calc ulus of the curvature.

2.3   Curvature Profile

The mean direction of the strokes will be determined by the use of a differential technique known as DOS (Difference of Slopes) [5], proceeding then with a research of the segmentation points within the curvature maxima obtained.

The directional calculus is done by the displacement of a pair of segments of size W, secants to the trace and separated by a distance m, measuring the angular variation θ between them. The specific values of DOS parameters are application dependent and are generally determined heuristically. For the specific case of the signatures used, we found the values of $W=2\%$ of the size of the component and $m=2$ pixels to be appropriated.

The application of DOS over discrete angular variations generates high frequency noise. We chose to attenuate such influence through the application of a discrete variable resolution convolutive Gaussian filter, with the resolution set to W and a kernel of seven elements, reducing in this way the intrinsic noise of the method without dislocating the constant curvature regions.

2.4   Constant Curvature Regions

The research procedure for the constant curvature regions is iterative, aiming at the minimization of the standard deviation around the mean value of the curvature between two maxima. The algorithm works as follows: We initially determine the mean value of all curvature points within two curvature maxima. Next we verify the value of each extremity of the interval. If the difference between the value of this extremity and the mean is greater that a threshold we eliminate the extreme point and repeat the procedure until the standard deviation is equal or less than the desired threshold. The value for the threshold was chosen based on the greatest possible value of θ for a straight eight-connected segment [5] and varies with the length of every component.
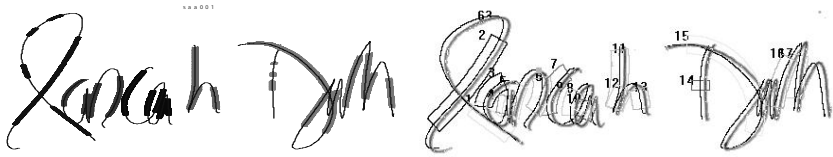
Fig. 1. Segmenting and constructing the windows around simple stroke regions

2.5   Windowing Process

Windows are created around the simple stroke regions, following the general outline of the strokes. The length of each window is equal to the length of the correspondent simple stroke and the width will be determined during the learning process. In order to form the windows the skeleton and the original image are overlapped, centering both from their respective center of gravity. The center of gravity of the image is calculated by first eliminating the background through a threshold operation over the histogram of the gradient [6], calculated by a Sobel mask operator, and then by using the pixel intensity values as weights. Generally the overlapping procedure is not perfect due to background noise of the image and due to rounding errors but the differences are compensated during the learning phase.

Each window is individually constructed and labeled forming an envelope around the signature that can be easily accessed during the local feature extraction process.

3   System Architecture

3.1   On-Line/Off-Line Hybrid Module

This module of the system is composed by six processing steps, as follows:

Step 1.: Acquisition: The reference signature on/off line data is read.

Step 2.: Preprocessing: The input data is filtered by a low-pass filter in order to eliminate spurious noise inherent to the acquisition process [2], and it is pre-segmented into strokes produced by pen-up/pen-down movements.

Step 3.: *Step 3:* Recursive sampling: A skeleton of the signature is generated by recursive sampling of the resulting points by splines.

Step 4.: *Step 4:* Segmentation into strokes: The skeleton is segmented according to the complexity of the underlying strokes [3] (Fig. 1).

Step 5.: *Step 5:* Windowing: Local windows – or foci of attention - are created around the simple stroke regions, following the general outline of the strokes (Fig. 1).

Step 6.: *Step 6:* Learning: During this step the size of the windows will be adjusted by a process aiming at reducing the within-class distance between any given pair of reference signatures acquired. The resulting signature skeleton with its personalized windows envelope will be used during the verification module of the system. Meanwhile the mean and standard deviation of all local and global features used by the system are computed in order to be fed later into the classifier. They are as follows:

Global features:

- The aspect ratio of the signature, calculated by the ratio of the standard deviation of the pixels in relation to the center of gravity of the signature on both X and Y axes.
- The proportion of pixels localized outside the rectangle formed by the standard deviation of the pixels previously determined.
- The slant of the signature.
- The distribution of the pixels around the X and Y axes (standard deviations).

Local Features:

- The overall direction of each stroke represented by the circular mean of the directional probability density function calculated over each window.
- The proportion of pixels inside each window over the total number of pixels.

The direction of each individual signature stroke plays a fundamental role in the proposed learning process. The distance between a given pair of reference signatures is calculated as the total difference of direction of each individual stroke of the signature. A small distance represents a great similarity between both signatures. The reference signatures are analyzed in pairs, with the envelope of one signature overlapping the image of another. For a three reference set this represents a total of 6 possible combinations.

For any given two reference signatures $S1$ and $S2$, the learning process is as follows:

1. Initially the fixed width window envelope of $S1$ (generated previously) is put on top of the image of the same $S1$ signature in order to calculate the mean direction of its individual strokes for the given window size configuration, producing $\theta S1_1 \cdots \theta S_n$ where n is the number of windows.
2. The envelope of $S1$ is now put over the image of the second signature $S2$, and the corresponding $\theta S2_1 \cdots \theta S2_n$ mean directions are calculated.
3. The total distance between both signatures is given by:

$$DistS1S2 = \sum_{i=1}^{n} \sqrt{\left(\theta S1_i - \theta S2_i\right)^2} \qquad (1)$$

4. The width of the first window is changed and steps 1 to 3 are repeated. If the new calculated distance is smaller than the previous one, the new window size is kept, if not, the original width remains. Step 4 is repeated for the entire range of desired width values for each window. Experiments were done using several different ranges and the values of 80, 100 and 120 pixels worked the best for the database available, so Step 4 is to be repeated three times for each window of the envelope.
5. The result of Step 4 above will be a customized set of windows that generates the minimal distance between the two signatures $S1$ and $S2$. Next, another pair of signatures is selected and the process repeats itself until there are no more combinations available.

After the training there will be a set of optimized windows for all of the reference signatures. In order to allow within-class variations, the signature that presents the maximum calculated distance among the optimized windows will be chosen as the prototype that will represent all the signatures of the class.

3.2   Off-Line Module

The off-line module will process the image of the test signature, comparing it against the reference data.

Step 1.: Acquisition: The test signature image is fed into the system.

Step 2.: Preprocessing: The input data is filtered in order to extract it from the background.

Step 3.: The correspondent window-formed skeleton is read from the database and is put over the image of the test signature.

Step 4.: The extraction of the local and global features takes place, delimited by the windows of the reference skeleton.

Step 5.: The decision over the authenticity of the test image is taken upon the comparison between the local and global features extracted versus the thresholds stored during the learning phase.

4   Experimental Results

The overall performance of the system was assessed by the means of the Equal Error Rate (ERR). In order to test the system two databases were used. The first one (hybrid) was composed by both on-line and off-line data from the same authors. The second database contained only images of signatures and was used mainly to further validate the false acceptance rate of the system.

The hybrid database was made of 400 signatures, with 20 signatures per writer (20 writers). The online data was acquired through a digitizing tablet and the authors were instructed to write over a framed white sheet of paper placed on top of the tablet to produce the corresponding off-line data. The off-line data was then gathered by scanning the correspondent images (using grayscale). The second off-line only database was formed by 500 signatures of 50 other writers (10 signatures per person).

In the first experiment, multiple Euclidean Distance Classifiers working on a vote scheme were used to assess the performance of the system. A vote was issued if the value of a given feature was within the threshold calculated during the learning phase. The number of votes issued by all the seven processed features determined the acceptance or refusal of the signature.

Initially three randomly chosen signatures were used as reference, with the remaining 17 signatures used as test. In this case the system produced an equal error rate of 8.54% (considering 5 favorable votes). With five signatures as references the results improved, with an ERR of 5.30%.

In the next experiment the 500 unseen signatures from the second image-only database were used as random forgeries instead of the remaining signatures of the first database. The system produced an ERR 7.77%. With five signatures as references an ERR of 4.20% was obtained.

Finally, a single Euclidean Distance classifier was used, where the similarity measurement was done over the addition of all the distances produced by the individual features. The first database produced an ERR of 10.8% with 3 reference signatures, 5.16% with 5 signatures and 1.19% with 10 references.

5   Conclusion

On-line systems have good performance but have restricted use due to the need of the dynamic input. Off-line systems don't have such restriction but have a poorer performance and usually base the extraction of local features almost at random, or at most using well known grid-like segmentation schemes [7]. In this work a hybrid on/off line window based segmentation scheme is proposed where the local feature extraction is done within the windows constructed around signature regions, determined from the analysis of a psycho-physical handwritten reproduction model. The segmentation is done in a robust and consistent way, focusing the feature extraction process. The reference data is acquired by a digitizing tablet but only the image of the test signature is needed during the verification process.

The proposed system was able to correctly segment the signatures producing equal error rates of about 5% for 5 reference signatures and about 1% for 10 reference signatures by the use of common Euclidean Distance classifiers. Further improvements on the results could be achieved by the analysis of the regions around complex strokes and by using a more efficient classifier.

6   Acknowledgements

References

[1]   F. Leclerc and R. Plamondon, Automatic Signature Verification: The State of the Art - 1989-1993, In: International Journal of Pattern Recognition and Artificial Intelligence, Vol. 8, No. 3 (1994) 643-660

[2]   J-J. Brault and R. Plamondon, Segmenting Handwritten Signatures at Their Perceptually Important Points, In: IEEE Transactions on PAMI, Vol. 15, No. 9 (1993) 953-957

[3]   A. Zimmer and L. L. Ling, Preprocessing: Segmenting by Stroke Complexity, In: Proceedings of the VI Iber-American Symposium on Pattern Recognition, Brazil (2001) 89-94

[4]   W. Guerfali and R. Plamondon, The Delta LogNormal Theory for the Generation and Modeling of Cursive Characters, In: Proceedings of the ICDAR, Vol. 2 (1995) 495-498

[5]   L. O'Gorman, Curvilinear Feature Detection from Curvature Estimation, In: Proceedings of the 9th International Conference on Pattern Recognition (1988) 1116-1119

[6]   N. Otsu, A Threshold selection Method from Grey Level Histograms, In: IEEE Transactions on Systems Man and Cybernetics, Vol. 9 (1979) 62-66

[7]   R. Sabourin and G. Genest, An extended Shadow-Code Based Approach for Off-Line Signature Verification: Part I, In: Proccedings of the IAPR, Jerusalem, Israel (1994) 450-455

On-Line Signature Verification
by Explicit Solution
to the Point Correspondence Problem

Jakob Sternby

Centre for Mathematical Science, Lund University
Box 118, S-221 00 Lund, Sweden
`jakob@maths.lth.se`

Abstract. This paper presents a novel tool to be used for signature verification. The method is based on the *vector context matrix* concept that enables robust and precise extraction of the point correspondences between a sample and a database model. Using a digitizing tablet as a source for input, information of the pressure, speed and pen inclination is obtained in addition to the coordinates of the sampled stroke. The knowledge of the point correspondences enables studying these features pointwise and thereby increases the discriminatory power of verification systems. A simple system based on this tool has been tested on the samples of one signature from the SVC 2004 dataset with a perfect verification rate.

1   Introduction

Automatic identity authentication by signature verification is a comparatively old topic with research papers dating as far back as the 1980s [7]. Recent progress in digitizing tablet development has produced tablets which, in addition to the normal time and positioning data, also produce data with the pressure and tilt of the pen on the tablet [4]. Regardless of the features used for the signature verification most implementations use a probabilistic approach to solve the problem. So far there seems to have been mainly two strategies for the signature verification problem [5]. Either comparison of global features such as the total time spent and average speed, or functional features such as coordinate position, speed and acceleration versus time along the signature. For the latter part two approaches have received a lot of attention. The first, named Hidden Markov Models (HMM), is a powerful tool since it can avoid problems of comparing signatures with varying length and height [3]. The other method which, like HMM has also proved successful for handwriting recognition, is Dynamic Time Warping (DTW). This method allows successive matching of point configurations with a varying number of points [8]. A problem with HMM is that one in general has little insight into the recognition process. DTW on the other hand has a weakness in that it requires pre-processing that normalizes size and position of each sample into the coordinate system of the model.

This paper presents a new strategy made possible by introducing the notion of the *vector context matrix* (VCM). With the help of the VCM the point correspondence problem between a template and sample signature is solved. Using point contexts for solving the correspondence problem has been successful in the past [1]. Having solved the correspondence problem one can compare the features from the digitizing tablet evaluated at corresponding points and thereby increase the discriminatory power of the system. Previous experiments have shown that these features are sufficient to detect forgers that are skilled at imitating the shape of a signature. By defining the comparison of each of these features (including the distance measure from the correspondence problem) as a separate binary classifier one obtains a multiple expert system. Multiple expert systems have previously been shown to improve performance of traditional signature verification methods [2].

2 Core Points and the Vector Context Matrix

2.1 Core Points

Digitally sampled handwritten symbols in general contain a lot of excess information. In symbol recognition one is primarily concerned with points that have a crucial impact on the shape of a symbol. Points that lie on a nearly straight line between two points are in general uninteresting. We make the following definition of these crucial points that we call **core points**.

Definition 1. *Let y be a symbol and let $\mathbb{S}_y$ be the space of all sequences of points $X = (x_1, x_2, \ldots, x_n), x_i \in \mathbb{C}$ that represent y. A minimal subsequence of points*

$$C_q^*(X) = \{c_i\}_{i=1}^q \subseteq X, \ X \in \mathbb{S}_y$$

*that still contain the discriminatory features of a shape of class y and that have corresponding points for all elements of $\mathbb{S}_y$ are called **core points** C_q^* of X. The shape, or the image, of a set of **core points** will be referred to as a **core shape**. In the database the **core shape** of the template will be referred to as the **core** of the database model.*

Extraction of Potential Core Points Due to the deformations inherent to handwritten characters it is nor possible to construct a method that extracts the **core shape** of a sample from a symbol class without utilizing information about the **core** of that class. However, one may utilize the fact that **core points** are significant points to extract a subset of the original points which is smaller but still contains *all* **core points**. One may for instance disregard all points between the endpoints of a straight line segment. To aid in our further explanations we make the following definition:

Definition 2. *By **potential core points** C_p we mean a subset of a sample character X that contains at least all of the **core points** $C_p(X) = \{c_i\}_{i=1}^p, C_q^*(X) \subseteq C_p(X) \subseteq X$.*

We will show how the set of **potential core points** can be used to obtain an estimate of the **core** of a class. There are several ways of constructing algorithms that extract potential core points from a given sample. The only requirement on these algorithms is that they should extract *at least all* of the **core points**. In this paper we have limited the **core shape** to contain variations in the direction orthogonal to the writing direction. Making an analogy with the compass, we let the set of **potential core points** be the local extremepoints in the directions NW, N, NE as well as SW, S, SE given that the writing direction is from W to E.

2.2   Matching Corepoints with VCM

The Context of a Core Point As stated in the definition, the core points of a sample of a certain class should represent the discriminative features of that shape. To aid in determining which subset of the **potential core points** that actually define the **core** of a sample, we define the **vector context of a core point**.

The **vector context of a core point** will here be defined as the set of vectors to all the other core points together with a normalization constant. The reason to include a normalization constant here is of course to assure the scale invariance of these vectors.

In this paper we have used a normalization constant that is defined by the max distance between subsequent core points. We will refer to this normalization constant D as the max vector length and it will be defined as $D = \max_i \|c_i - c_{i-1}\|, c_i \in C_p$.

Optimization by Vector Context Distance We will use the vector context above to find the subset of potential core points that best correspond to a **core shape** of a class. This is done by defining a distance measure for the vector context defined above. To seek the best corresponding subset of potential core points we then seek the ordered subsequence of q points in C_p with indexes I_q^* (note that there are $\binom{p}{q}$ such subsequences) that realizes $\min_{I_q \in P_q(C_p)} d(M, C_p(I_q))$, where $M = \{m_i\}^q$ is a **core shape** of $\mathbb{S}_y$ and $P_q(C_p)$ is the set of all ordered subsequences of length q from $\{1, \ldots, p\}$. In other words if the sample, denoted X, belongs to $\mathbb{S}_y$, M should optimally contain the mean **core shape** extracted from the set $\mathbb{S}_y$. From here on M will be referred to as the model (which could in other applications be a submodel of the multistroke class y). The distance measure d above should be such that it is small for natural deviations in **core shape** and large when matching other shapes. One may therefore want the model M to be flexible, but this has not been studied in this paper.

Vector Context Matrix As the heading insinuates one can form the vector contexts of all points in a model into a matrix. For a model M with q normalized

points $\{m_i\}_{i=1}^q$ as before we define the vector context matrix $\overline{\mathbf{V}}_{\mathbf{M}} \in \mathbb{C}^{\mathbf{q-1} \times \mathbf{q-1}}$ as

$$\overline{V}_M = \begin{bmatrix} m_2 - m_1 & m_3 - m_1 & \dots & m_q - m_1 \\ 0 & m_3 - m_2 & \dots & m_q - m_2 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & m_q - m_{q-1} \end{bmatrix}.$$

Similarly this matrix can be created for the potential core points and we define $\overline{\mathbf{V}}_{\mathbf{C_p}} \in \mathbb{C}^{\mathbf{p-1} \times \mathbf{p-1}}$ as

$$\overline{V}_{C_p} = \begin{bmatrix} (c_2 - c_1)/D & \dots & (c_p - c_1)/D \\ 0 & \dots & (c_p - c_2)/D \\ \vdots & \vdots & \vdots \\ 0 & \dots & (c_p - c_{p-1})/D \end{bmatrix}.$$

A distance measure (for a fixed set of indexes I_q) for the vector context may then be obtained by taking the Frobenius norm of $[\|I_b(I_q)\overline{V}_{C_p} I_a(I_q) - \overline{V}_M\|_F$, where $I_b \in N^{q-1 \times p-1}$ is a matrix extracting rows with index $\{I_q(k)\}_{k=1}^{q-1}$ and $I_a \in N^{p-1 \times q-1}$ is a matrix extracting columns with indexes $\{I_q(k) - 1\}_{k=2}^q$.

Combinatorial Optimization The problem of choosing the subsequence of the potential core points that best matches to the core points in the model is equal to the problem of finding the submatrix of $\overline{V}_{C_p}$ that is most similar in a Frobenius-norm sense. Since the inclusion or exclusion of a point affects the vector contexts for all other core points it is not possible to solve this problem by conventional fast methods like DTW. However, the set where the distance function is defined is discrete and finite (although sometimes very large) and hence there will be a minimal value attained in this set.

This value will be attained if we apply the branch and bound method to solve the problem (see for example [6]). Convergence speed of the branch and bound method will be discussed briefly in the result section of this paper. It may be desirable in some cases to add a penalization function to avoid a minimum at global structures similar to the sought local structure.

3   A Simple Verification System Based on Core Points

An automatic system consisting of a training procedure (database generation) and a test procedure has been constructed. An algorithm for generating the common **core shape** of a set of samples of a signature has been developed. Statistically modelling the features evaluated at the common **core** of the training samples is then straight forward. In this implementation we ignored the number of strokes and concatenated all strokes into one model. If users comply to maintain a fixed number of strokes the discriminatory power against forgeries could be augmented further.

3.1 Automatically Building Core-Shape Models

In this section we present an algorithm that automatically estimates the **core** of a set of training samples. It is an iterary procedure that gradually decides whether a point in the **potential core point** set is a real **core point**.

Algorithm 1 *Automatic extraction of the largest common **core shape** of a set of t training samples. Let the samples be indexed by $I_t = 1, \ldots, t$.*

0 Set $i := 1$.

*I Compare the subsequences of N and S points in the sets $C_{p_j}^j, j \in I_t$. Choose the first n points of each of these sequences to define a **core** model $M_n^{j,i}$. Extract the correspondences between $M_n^{j,i}$ and the other NS-subsequences. Choose the model $M_n^{k,i}$ from the kth sequence where the correspondence solution resulted in the largest number of skipped points in the other subsequences.*

II The rest of the $C_{p_k}^k$ sequence between the points corresponding to the skipped points in the other subsequences are added to $M_n^{k,i}$. This altered model $(M_n^{k,i})^*$ is then tested as a subsequence against the $M_n^{j,i}, j = 1, \ldots, t$. If the best correspondence solution to all of these imply that a NW, NE, SW, or SE has been matched in $(M_n^{k,i})^*$, we relabel the corresponding point in $C_{p_k}^k$ to a N (or S) point.

III Rerun step I and II until there are no points that should be added or that no points are skipped in the models.

IV Set $i := i + 1$. Move the start point of the subsequences of the C_p^js so that the first point of $M_n^{j,i+1}$ will correspond to the last point in $M_n^{j,i}, \forall j \in I_t$. To make the process more stable we can set this startpoint to the $(n-l)^{th}$ point in the $M_n^{j,i}$s so that the models overlap by l points. Redo all steps until for some r the nth point of $M_n^{j,r}$ corresponds to the last point in $C_{p_j}^j$ for some j.

3.2 The Database

The database template for an authentic signature is generated from a number of sample signatures from a training set. In this paper the size of the training set was 5. It is of course desirable for the user that this number be as low as possible. First the **core shape** of these samples was extracted using algorithm 1. The training signatures with the matched **core** is shown in figure 1.

For each of the training signatures the correspondence problem is solved like above and the vector context distances and correspondences are obtained. In this simple implementation the **core** is formed by combining all the submodels from algorithm 1 into one. For simplicity all of the features are here assumed to be normally distributed. Consequently the database will contain a normal distribution of the distances to the **core** built on the distances for the training signatures. The mean vectors of speed, pressure and pen inclination at the **core points** of the samples are also computed. Based on these mean vectors the normal distributions for these features are inferred from the training samples.
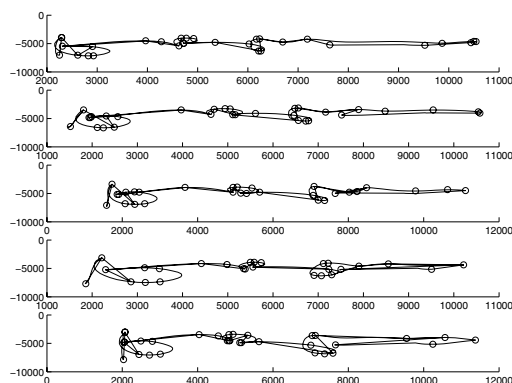
Fig. 1. A plot of the sample signatures in the training set. The **potential core points** of each sample are marked with circles. The extracted **core** is displayed with a connected line

We call the distribution of each such trait an **expert**, since it will be called upon to determine whether the profile of an unknown sample are within the acceptable statistical variations of an authentic signature.

3.3   Verification

To determine whether a signature is authentic or not we first calculate the correspondences to the **core** in the database. This procedure will also calculates the distance in vector context from the extracted **core shape** of the unknown sample to the **core**. We also evaluate the speed, pressure and pen inclination profiles at the extracted **core shape**. The experts in the database then produce an evaluation of each of the traits to evaluate whether the extracted profiles are likely to come from an authentic signature.

4   Experiments

We have tested the system on signature 1 from the sample data provided by SVC 2004 (www4.comp.polyu.edu.hk/~icba/). It consists of 20 authentic signatures and 20 skilled forgeries. Five of the authentic signatures were used in training and hence the testing set consisted of 15 authentic and 20 forged signatures. The system rejected all forgeries and admitted all authentic signatures in the test set. Figure 2 clearly displays the difference in features evaluated at the **core** between the authentic and forged samples of the test set.

4.1   Convergence of Combinatorial Optimization

For almost all samples in the training and testing set of signatures, the implementation had a stable convergence rate. For the 15 authentic signatures the
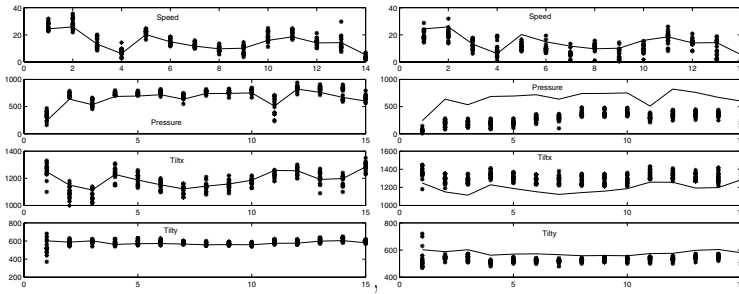
Fig. 2. A scatter plot of the features evaluated at the core points. The left plot shows the authentic signatures in the test set and the right shows the forgeries in the test set. The mean stored in the database is displayed as the connected line

median number of steps required by the branch and bound algorithm was 2934. The worst convergence for one authentic signature was 68 000 steps. Forgeries did significantly worse in terms of convergence. Two signatures required more than 300 000 steps to converge and the median for the remainder was 5698 steps.

Typically bad convergence occurs when there are potential core points in the sample that are very close and which correspond only to one significant point in the core of the database or when in general there is no good match for the core points in the database. A good remedy for these convergence problems is probably to use the vector context matrix approach like it was originally intended - to extract correspondences for submodels in a sample. These submodels are automatically obtained by algorithm 1.

5   Conclusions and Future Work

In this paper a novel method for signature verification has been presented. The method utilizes the explicit solution to the point correspondence problem, enabled by the vector context matrix approach, to obtain speed, pressure, and pen inclination profiles of corresponding points. In this paper the method has only been tested on very limited data but the experimental results are very promising.

There are obvious flaws when using the vector context distance measure to match against words since the distance measure is sensitive to irregular transformations of the individual letters in a signature. Algorithm 1 for finding a common **core shape** extracts a sequence of submodels. To really benefit from the knowledge of the explicit correspondences one should use these submodels in the matching process. This way one can discriminate harder on parts of the authentic signatures with high consistency. In the experiments of this paper no forgery was rejected because of differing shape.

In some cases there were convergence problems for the branch and bound algorithm. It is probable that bad convergence is caused by forgeries in most cases so one way to improve the convergence rate could be to fix a an upper

bound for the number of iterations. Subdividing the correspondence extraction to submodels, as proposed above, is also likely to eliminate most convergence problems.

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(24):509–522, 2002. 570

[2] M.C Fairhurst C. Allgrove. Majority voting for improved signature verification. *IEE Colloquium on Visual Biometrics*, (18), 2000. 570

[3] A. Kosmala G. Rigoll. A systematic comparison between on-line and off-line methods for signature verification with hidden markov models. In *Proceedings: 14th International Conference on Pattern Recognition*, volume 2, pages 1755–1757. International Association on Pattern Recognition, 1998. 569

[4] Y. Itoh I. Nakanishi, N. Nishiguchi and Y. Fukui. On-line signature verification method utilizing feature extraction based on dwt. In *Proceedings of the 2003 International Symposium on Circuits and Systems*, volume 4, pages 73–76. ISCAS '03, 2003. 569

[5] Ma Mingming and Sardha Wijesoma. Automatic on-line signature verification based on multiple models. In *Proceedings of the IEEE/IAFE/INFORMS 2000*, pages 30–33. Conference on Computational Intelligence for Financial Engineering, 2000. 569

[6] Christos H. Papadimitriou and Kenneth Steiglitz. *Combinatorial Optimization*. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1982. 572

[7] M Plamondon, R; Parizeau. Signature verification from position, velocity and acceleration signals: a comparative study. In *Proc. 9th International Conference on Pattern Recognition*, volume 1, pages 260–265, 14-17 Nov. 1988. 569

[8] Luc Claesen Ronny Martens. On-line signature verification by dynamic time warping. In *Proceedings: 13th International Conference on Pattern Recognition*, volume 3, pages 38–42. International Association on Pattern Recognition, 1996. 569

New Features for Authentication
by On-Line Handwritten Signatures

M. Wirotius[1,2], J. Y. Ramel[1], and N. Vincent[3]

[1] Laboratoire d'Informatique, Université de Tours
64, av. Jean Portalis 37200 Tours France
`matthieu.wirotius@etu.univ-tours.fr`
`jean-yves.ramel@univ-tours.fr`
[2] AtosOrigin
19, rue de la Vallée Maillard, BP 1311, 41013 Blois Cedex France
[3] Laboratoire SIP, Université Paris 5
45, rue des Saints Pères, 75270 Paris Cedex 6 France
`nicole.vincent@math-info.univ-paris5.fr`

Abstract. Authentication by handwritten signature is the most accepted authentication system based on biometrics partly due to the fact that it is easy to be used and part of our cultural habits. We present some improvement for one of the most important problems in such systems: features construction and selection. Concerning the study of the signature's shape, we discuss the impact of the selected features on the global system architecture. We propose to extract information linked to the writer by computing new features based on fractal dimension of the on-line signature. In order to prove the interest of these features, several performance evaluation methods are presented. Finally, the results of tests using the combination of new features with classical ones, and a method of features selection using Genetic Algorithm are discussed.

1 Introduction

The use of handwritten signature in order to authenticate someone, relies on the instinctive way the movements carried out at the time of signing are performed. Then signature is reproducible, stable and has constant characteristics, specific to the author. That is why on-line signature is considered as a comportmental biometric method.

The term authentication covers in fact two different problems: identification and verification. Here we are more concerned with verification problem than with identification. In this case, the type of errors are to be precised and two categories can be considered; FRR indicates the ratio of authentic signatures rejected and FAR the ratio of forgeries recognised as authentic ones. We note EER the error value for which FAR is equal to FRR. These rates determine the quality of systems, besides the values that can be accepted also depend on the desired level of security.

This article is focusing on the fundamental stages, extraction and evaluation of characteristics. Among the whole literature, a constant question appears: how to extract characteristics from the signature which would be at the same time specific to the individual and "hidden" within the signature, like pressure or movements carried

out when the pen is up. Within this context we have chosen to compute new features based on the fractality of handwriting [1]. Indeed, this dimension extracts information out of reach of the writer's consciousness but reproducible information on the writing. These two criteria make fractal dimension an element of growing interest in the field of identification by handwriting [2]. In this paper, we will thus present the various ways of calculating fractal dimensions and their respective interests. The second part of the paper presents some methods to evaluate the features performance. Indeed, it is necessary to determine which characteristics are at the same time stable for a single individual and varying with respect to other individuals. The use of a genetic algorithm allows evaluating the performance of the new features we propose and can be adapted to select the most suitable features out of a feature set. The purpose of this selection is to remove the characteristics that could cause errors during classification. Finally, after a presentation of our test database we will compare the results obtained with traditional characteristics with those obtained with characteristics related to the fractal aspect of the signature and with those obtained after using our selection method.

2   New Features for Signature Description

Extraction of features is one of the most significant stages because it means a change in the representation space. Then, the signature will no more be represented by a succession of points but by a vector whose components indicate each characteristic.

2.1   Classical Features

Some characteristics are used in the majority of the methods of authentication by handwritten signature [5]. Here is a non exhaustive list: angle between the first and the last point, sum of the angles between the points of the signature, sum of the absolute angles between the points of the signature, length of the signature, distances between the first and the last point, ratio between displacements towards the left and the right-hand side, ratio between displacements towards the top and the bottom. Current results, using only these features, are not sufficient for all the reasons we discussed before. So we decide to compute new features, considering the shape of the signature. Nevertheless, these 8 traditional characteristics will be used thereafter in order to show the interest of the new suggested characteristics.

2.2   Fractal Dimension

The concept of fractal dimension takes place in the family of non-integer dimensions defined by following studies of Hausdorff or Minkowski. A set for which we can measure neither length, nor area, and which admits a finished measurement when a particular exponent of the gauge is considered is a fractal set. This particular value is called the dimension of Hausdorff or fractal dimension of the studied element. The calculation of fractal dimension is based on the existence of a power law relation between a measurement and a parameter that indicates the scale of observation to which measurement is realised. We then estimate this relation by the use of an

evolution graph. Fractal dimension characterises the irregularity or fragmentation degree of a set [3]. It can thus be used to quantify the complexity of a curve or the legibility of a text. One of its most interesting properties is its invariance with respect to an affine transform. Here, the principal interest of fractal dimension is to extract elements specific to the individual – elements which are not discernible while reading - from the signature. A great number of methods exist to estimate it according to the type of input data. In this article, we will present three methods, one that considers the signature as the shape we have in an off-line process. We can reproduce the initial line by joining the various points according to the temporal information. The two other methods use only the coarse data points. Then, the distribution of the points along the line is taken into account and thus brings an indication on the order of the strokes and on the dynamic.

2.3 Off-Line Approach

This first method studies the line of the signature rebuilt using the Bresenham algorithm between temporally consecutive dots. Consequently, only the shape of the signature is taken into account as in an off-line approach. The fractal dimension value of set X is given by the following formula where ε indicates the observation scale, n the dimension of the space we are working in and A the measure associated with this space:

$$D(X) = \lim_{\varepsilon \to 0}\left(n - \frac{\log(A(X_\varepsilon))}{\log \varepsilon} \right) \tag{1}$$

For a set X in a plane with dimension 2, we calculate the area of the subsets X_1 to X_d. X_k is the transformation of X in a morphological dilation the structuring element of which is parameterised by a length k (figure 1a). The area of the signature as well as the areas of its different transformations by morphological dilation using as structuring element a 3x3 square, can be calculated. The formula then becomes:

$$D(X)=\lim_{\varepsilon \to 0}\left(1 - \frac{\log(A(X_\varepsilon)/\varepsilon)}{\log\varepsilon} \right) \tag{2}$$

(a)   (b)
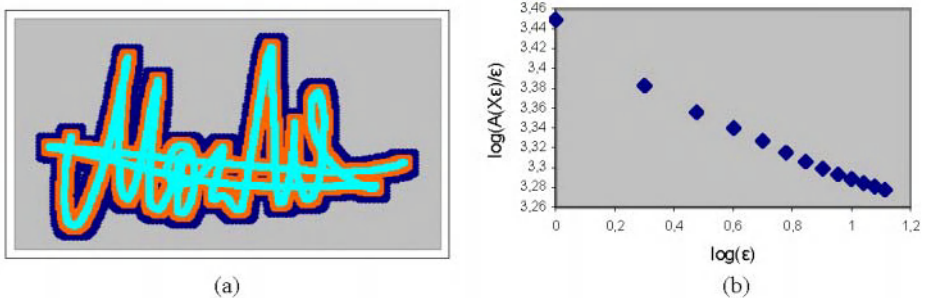
Fig. 1. (a) Computing of the fractal dimension; (b) Illustration of the evolution curve

We can represent the evolution of the logarithm of the area divided by the number of iterations carried out according to the logarithm of the number of iterations (figure 1b). With each dilation, we lose details on the signature as with a larger scale.

From the experiment, we can notice, as we see on figure 1b, that the representative curve of $\log(A(X_\varepsilon)/\varepsilon)$ with respect to the scale can be approximated by two consecutive straight line segments. These two line slopes obtained by linear regressions on the graph points can thus estimate the evolution of the area. In the case of signature images, empirically the number of iterations was fixed at 20 because beyond that, we do not get new information. The breaking point in all the 20 points is obtained by the search of the furthest point away from the segment joining the first and the last point of the curve. We will consider for fractal dimension the largest slope between both the first point and the break or between the break and the last point. Then we are defining one feature.

2.4 Vectorisation Based Approach

This second method works on the list of the original points and is based on a polygonal approximation of the signature [6]. The interest of this method lays on the use of the temporal order of the points, contrary to the first method that only considers the position of the points and where the global order of the points was lost. Off-line, this information is not available. In the polygonalisation process, the error threshold determines the number of vectors necessary to describe the whole shape of the signature for a given level of observation. The polygonal approximation can evolve gradually and the signature will become more and more rectilinear. This principle is thus related to a multi-scale representation of the signature. The formula (1) becomes then:

$$D_2(X) = \lim_{\varepsilon \to 0}\left(1 - \frac{\log(N(X_\varepsilon)/\varepsilon)}{\log\varepsilon}\right) \tag{3}$$

Where $N(X_\varepsilon)$ represents the number of vectors necessary to approximate the signature at scale ε. X_ε is an approximation of the signature obtained by a vectorisation process characterised by a threshold equal to ε. So the fractal dimension is calculated by approximating successively the signature by a set of segments while the precision is varying. An example is given on figure 2. For each scale of approximation of the signature, we calculate the number of vectors obtained to approximate the signature. Then we represent the evolution of the logarithm of the number of vectors divided by the number of iterations carried out with respect to the logarithm of the approximation threshold. At each iteration, we will decrease the precision of the approximation of the signature and thus will take into account fewer and fewer details.

As in the first fractal approach, the curve representative of $\log(N(X_\varepsilon)/\varepsilon)$ can be approximated by two consecutive straight line segments. These two line slopes obtained by linear regressions can thus estimate the evolution of the approximation. From the evolution, the same process as previously is performed to define the vector

fractal dimension, starting from the curve representing the number of vectors according to the iteration count carried out.
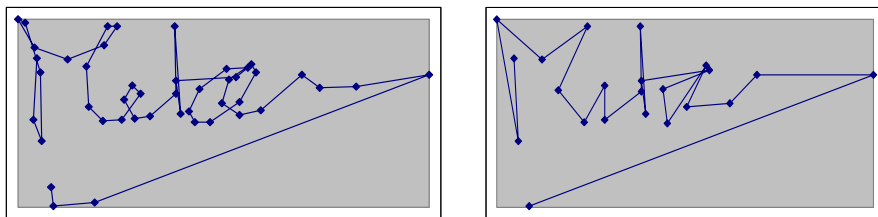
Fig. 2. Illustration of the evolution of the approximation of the signature

2.5 Local Calculation

The method is identical to the first method but, instead of considering the reconstituted signature, we use as X set the set of the data points. The mask used for dilation is the same one as in the first method. The interest of this method is that we operate directly on the signature data and thus we do not compare the signature to a succession of segments, as it is the case for the calculation of a great number of traditional characteristics as the length or the sum of angles.

It should be noticed that the curve obtained (Figure 3b) is different from the previous graphs but we can also approximate it by two half-straight-lines. The break point indicates when the dilated sets from a dot set becomes more like a curve.

Each segment can be interpreted: the first half on the left corresponds to the dilations carried out before the contact between the different dilated points. It is thus related to the speed or more exactly to the speed regularity of the signature execution. The more the drawing will be jerked, the more the slope of the first half-line will be significant. The slope of the second half corresponds to the next dilations. It is thus related to the signature shape. In this case, both slopes are considered as features.

3   Performance Evaluation of these New Features

First of all we can mention an experiment we have conducted in order to verify the discriminating power of the features introduced. We used a Fisher test to see whether a feature can differentiate between two signatures. An average value is computed on a whole database. The experiment showed fractal features are more discriminating than classical features.
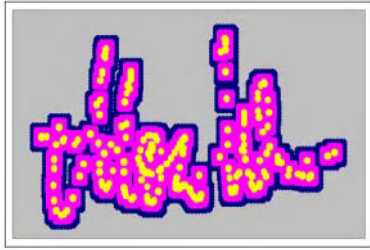
Fig. 3a. Evolution of the surface of the signature
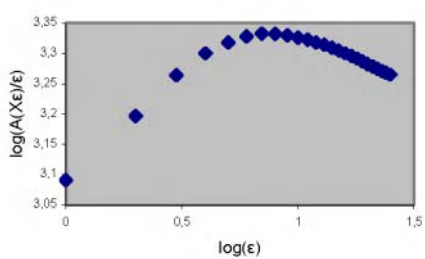
Fig. 3b. Illustration of the evolution curve
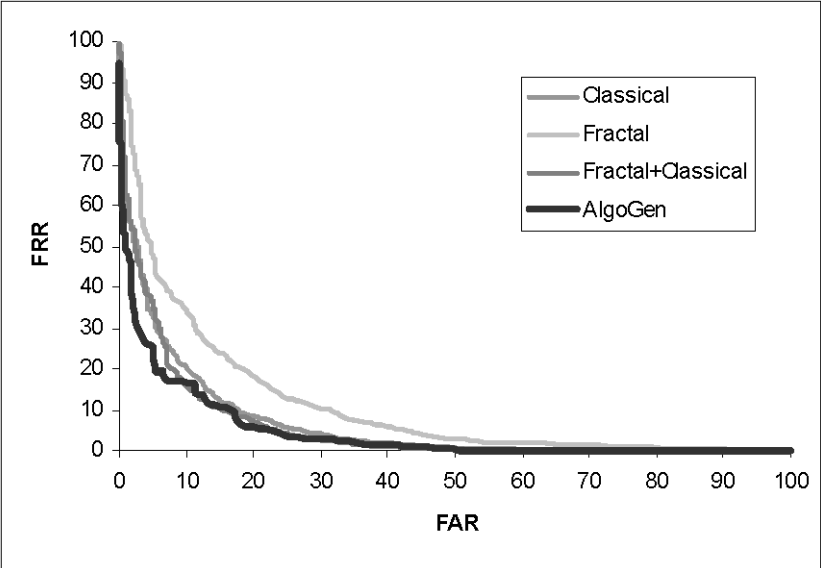
Fig. 4. FAR vs. FRR

Then we are to evaluate the performance and the complementarities of these four new fractal features with respect to the traditional ones (8). We are looking for features characterising best the signature of an individual and differentiate it from others. Indeed, some can cause error; it is thus appropriate not to take them into account. So a genetic algorithm is used.

To perform all those experiments we are using a signature base that contains 800 genuine signatures from 40 writers. Among 20 authentic signatures of the same writer, 5 are used to build the signature model and the others are for tests. Each writer involved in the database is resumed by way of an only model. The models are vectors and consist of each feature average and variance calculated from 5 signatures. Let m_i and σ_i, represent, the average and standard deviation of feature i. Let n denote the total number of features, α be a fixed threshold, and t_i the value of feature i. The decision rule is:

If $\max_{i=1..n}\left(\left|t_i - m_i\right|/\sigma_i\right) < \alpha$, then the signature is accepted, else the signature is said false. Figure 4 shows the signature verification results obtained with different feature sets. We have chosen to make α vary in order to define different systems. The quality of these systems is figured in a two dimensional space. The FAR and FRR parameters, in the verification problem are representative of the system.

The characteristics related to fractal dimension are less significant to authenticate a signature when they are used alone but the results show that they improve the signature verification significantly when they are used in addition to the classical features.

The figure also shows the improvement to the signature verification results using a multi-objective genetic algorithm, which minimizes FAR and FRR [4]. We see that increasing the number of features is not necessarily a good thing and that a selection step can be interesting. In fact the results are better with 6 or 7 characteristics than with 12 (fractal + traditional) features. According to the application, any classifier corresponding to the desired FAR or FRR values can be chosen.

4   Discussion and Further Directions

The quality of a system also involves the speed of the process, and in fact we can precise that fractal dimension has some reputation of high computation time. The number of iterations we presented in this paper however can be decreased because the slope of the left half-line can be approximated from the first 4 graph points obtained in the first iterations.

We have pointed out the original part of our work, in fact we can mention several ways of improvement. For example, the threshold for the classification process can be adapted to each characteristic and to each individual. The creation of the model can also be improved considering as a model of a writer each training signature. Besides, it is obvious, the more significant the base is, the more reliable the results will be in a real application. The principal interest of selection is to decrease the number of characteristics without reducing the performance. This selection stage is actually done only during the implementation of the system. We are to perform it in a personalised way. Indeed that would require having a consequent base of signatures at the time of enrolment. The features we have used are mostly linked to the shape of the signature so they principally allow determining random forgeries. To detect skilled forgeries, we need to introduce features more linked to the dynamic of the signature.

References

[1]   V. Bouletreau, N. Vincent, R. Sabourin, and H. Emptoz, "Synthetic Parameters for Handwriting Classification,". Proceedings of the International Conference on Document Analysis and Recognition (ICDAR'97), pp. 102-106, Ulm, Germany, August 1997.

[2]   K. Huang and H. Yan, "Signature Verification using Fractal Transformation," International Conference on Pattern Recognition (ICPR'00), pp. 2851-2854, Barcelona, Spain, September 2000.

[3]   Mandelbrot, The Fractal Geometry of Nature. W.H. Freeman and Company, New York, 1982.

[4]   L.S. Oliveira, R. Sabourin, F. Bortolozzi and C. Y. Suen, "Feature Selection for Ensembles: A Hierarchical Multi-Objective Genetic Algorithm Approach," Proceedings of the International Conference on Document Analysis and Recognition (ICDAR'03), pp. 676-680, Edimburg, Scotland, August 2003.

[5]   R. Plamondon and G. Lorette, "Automatic Signature Verification and Writer Identification – The State of the Art," Pattern Recognition, vol. 22, no.2, pp. 107-131, 1989.

[6]   K. Wall and P. Danielsson, "A fast sequential method for polygonal approximation of digitised curves," Computer Vision, Graphics and Image Processing, vol.28, pp.220-227, 1984.

On-Line Signature Verification
Based on Discrete Wavelet Domain
Adaptive Signal Processing

Isao Nakanishi[1], Naoto Nishiguchi[2], Yoshio Itoh[2], and Yutaka Fukui[2]

[1] Faculty of Regional Sciences, Tottori University, Japan
isao@fed.tottori-u.ac.jp
[2] Faculty of Engineering, Tottori University, Japan
{itoh,fukui}@ele.tottori-u.ac.jp

Abstract. This paper presents the on-line signature verification method based on the Discrete Wavelet Transform (DWT) and the adaptive signal processing. Time-varying pen-position signals of the on-line signature are decomposed into sub-band signals by using the DWT. Individual features are extracted as high frequency signals in sub-bands. Verification is achieved by utilizing the adaptive algorithm. If the adaptive weight converges on one, the sub-band signal is regarded as of the genuine signature. The total decision for verification is done by averaging verification results and then comparing with a threshold. In this paper, a nonlinear function is introduced to enhance each verification result. Experimental results show that the verification rate is 96%, while that of the time-domain method is 85%.

1 Introduction

As information services over internet such as the Electronic Commerce and the Electronic Data Interchange come to be used widely, the biometrics for the user authentication has attracted attention [1]. The on-line signature verification method classifies the signature by time-varying signals as the pen-position, the pen-pressure, the pen-inclination and so on [2, 3, 4]. In addition, the on-line signature verification method is suitable for the user authentication in the computer network service because the electronic pen-tablet which is used to digitize the on-line signature is a standard input device of the computer.

We have proposed the on-line signature verification method based on the Discrete Wavelet Transform (DWT) and the adaptive signal processing[5]. Time-varying pen-position signals of the on-line signature are decomposed into sub-band signals by using the DWT[6]. Individual features are extracted as high frequency signals in sub-bands. This makes difference between a genuine signature and its forgery remarkable. Verification is achieved by using the adaptive signal processing in each sub-band. In the adaptive signal processing, an adaptive weight is updated to reduce an error between an input signal and a desired one [7]. If the input signal is close to the desired one, the error becomes small and

then the adaptive weight is sure to converge on one. Therefore, when both the signals are of the genuine signature, the adaptive weight is expected to converge on one. By using the convergence of the adaptive weight, we can verify whether an input signature is genuine or forged. In our conventional method[5], the total decision for verification is done by merely averaging verification results and then comparing with a threshold. In this paper, we introduce a nonlinear function to enhance each verification result. Finally, we carry out verification experiments in computer simulation to confirm the performance of our verification method.

2   Enhancement of Signature Feature by DWT

The on-line signature is digitized with a pen-tablet. Especially, we use only pen-position data since it is at least provided in the portable device such as the Personal Digital Assistants (PDA). Actually, the pen-position data consists of discrete time-varying signals of x and y components which are $x^*(n^{'})$ and $y^*(n^{'})$, respectively. $n^{'}(= 0, 1, \cdots, N_{max} - 1)$ is a sampled time index. N_{max} is the number of sampled data. Moreover, to suppress the fluctuation of writing time and reduce the differences of place and size of the signature, the normalized x and y components are utilized in general.

$$x(n) = \frac{x^*(n) - x_{min}}{x_{max} - x_{min}} \cdot \alpha_x, \qquad y(n) = \frac{y^*(n) - y_{min}}{y_{max} - y_{min}} \cdot \alpha_y \qquad (1)$$

where $n(= 0 \sim 1)$ is the normalized sampled time index given by $n = n^{'}/(N_{max} - 1)$. x_{max} and y_{max} are maximum and minimum values of $x^*(n)$ and $y^*(n)$, respectively. α_x and α_y are scaling factors for avoiding under flow calculation.

In order to enhance the difference between the genuine signature and the forgery, we have proposed to verify them in the DWT domain[5]. In the following, $x(n)$ and $y(n)$ are represented as $v(n)$ for convenience. The DWT of the normalized pen-position component $v(n)$ is defined as

$$u_k(m) = \sum_n v(n)\overline{\Psi_{k,m}(n)} \qquad (2)$$

where $\Psi_{k,m}(n)$ is the wavelet function and $\overline{}$ denotes the conjugate. k is a frequency (level) index.

On the other hand, it is well known that the DWT corresponds to the octave-band filter bank[6]. Figure 1 shows a parallel structure of the DWT. ($\downarrow$ 2) and ($\uparrow$ 2) denote the down-sampling and the up-sampling, respectively. $A_k(z)$ and $S_k(z)$ where $k = 1, \cdots, M$ are the synthesis filters and the analysis filters, respectively. The synthesized signal $v_k(n)$ in each sub-band is called *Detail*. We consider the *Detail* as the enhanced feature of the pen-position signal.

3   Verification Using Adaptive Signal Processing

The procedure of our signature verification method is described in Fig.2. Before the verification, the template must be prepared. Five genuine signatures which
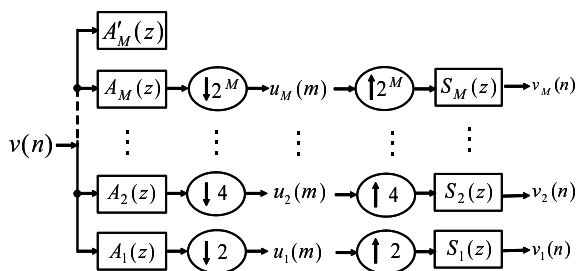
Fig. 1. Parallel structure of sub-band decomposition by DWT
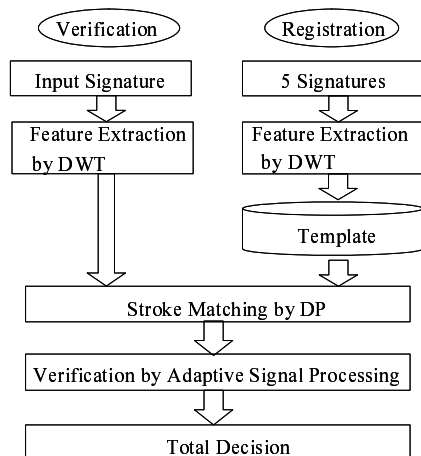
Fig. 2. Flow of proposed signature verification method

have equal number of strokes are decomposed into sub-band signals by the DWT each other. Decomposition level is decided after examinations of those genuine signatures. Extracted five *Detail*s are averaged at the same level. Incidentally, for taking the average, the number of sampled data should be equal in five signatures. However, each number of sampled data is generally different from the others even in the genuine signature. To solve this problem, five signatures are averaged every intra-stroke or inter-stroke (intra/inter-stroke) in our verification system. Concretely, the number of data in the template is determined by averaging the number of data in five intra/inter-strokes. Comparing the normalized sampling period in five intra/inter-strokes with that in the template, the nearest five *Detail* data are selected and then averaged. As a result, we obtain the template data in each intra/inter-stroke. By applying this operation to all intra/inter-strokes at all level, all template data are generated and they are registered in the database.
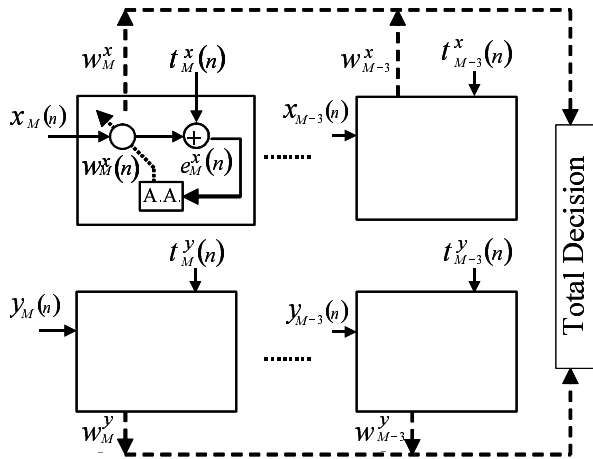
Fig. 3. Verification by adaptive signal processing

In the verification phase, an input signature is also decomposed into *Detail*s. The decomposition level M of the input signature is determined based on that in the template and given by $2^{M+1} \leq N < 2^{M+2}$ where N is total number of the template data.

By the way, not all genuine signatures have the same number of strokes. Immediately rejection of the input signature with different number of strokes causes degradation of verification performance. In addition, the proposed verification method is done every intra/inter-stroke and so the number of strokes in the input signature should be equal to that in the template. Therefore, we adopt the dynamic programming (DP) matching method to identify the number of strokes in the input signature and the template. The procedure of the stroke matching is omitted for lack of space. It is described in detail in [5].

After stroke matching, the verification is processed by using the adaptive signal processing[7]. Figure 3 shows the block diagram of the proposed verification method by using the adaptive signal processing. *Detail*s at only $k = M, \cdots, M-3$ are used in this method. *Detail*s at lower levels correspond to higher frequency elements and their variation is too large. They are not suitable for verification. Input signals $x_k(n)$ and $y_k(n)$ are respectively the *Detail*s of x and y components at level k in an input signature. The desired signals $t_k^x(n)$ and $t_k^y(n)$ are the *Details* of the template. $w_k^x(n)$ and $w_k^y(n)$ are adaptive weights and updated based on the adaptive algorithm (A.A.) to reduce the error signals $e_k^x(n)$ and $e_k^y(n)$, respectively.

The purpose of the adaptive signal processing is to reduce the error between the input signal and the desired signal sample by sample. However, if these signals have different number of sampled data, the error does not fully decrease. In general, the number of data in the input signature does not necessarily agree

with that in the template. In order to match these numbers of data, we utilize the normalized sampling time every intra/inter-stroke as used in making of the template. The nearest input data to the normalized sampled time in the template is only referred in the adaptive algorithm. Thus, the number of the input data is always guaranteed to agree with that in the template.

As the adaptive algorithm, we use a new kind of the steepest descent algorithm defined as follows. The time index based on the above mentioned normalized sampling time is represented as r. The pen-position indices x and y are also represented as v for convenience.

$$w_k^v(r+1) = w_k^v(r) + \mu E\left[e_k^v(r)v_k(r)\right] \tag{3}$$

$$e_k^v(r) = t_k^v(r) - w_k^v(r)v_k(r) \tag{4}$$

$$E\left[e_k^v(r)v_k(r)\right] = \frac{1}{N}\sum_{l=0}^{N-1} e_k^v(r-l)\ v_k(r-l) \tag{5}$$

$$\mu = \mu_0 / \left\{E\left[|v_k(n)|\right]\right\}^2 \tag{6}$$

$$E\left[|v_k(n)|\right] = \frac{1}{L}\sum_{l=0}^{L-1} v_k(n-l) \tag{7}$$

where L is the number of sampled data in the input *Detail*. μ is the step size parameter which controls the convergence in the adaptive algorithm. The step size parameter is normalized by the *Detail* power as shown in Eqs.(6) and (7), so that the convergence is always guaranteed. μ_0 is a positive constant and set to 0.0001, which is confirmed to be independent of the signature.

When the input signal is of the genuine signature, the error between the input and the template becomes small; therefore, the adaptive weight converges close on one. Inversely, if the input signature is the forgery, the weight converges far from one. In this way, the verification can be achieved by examining whether the converged value is nearly one or not.

The adaptive signal processing for verification is done in parallel at $k = M, \cdots, M-3$ levels in x and y components. After enough iterations for convergence, w_k^x and w_k^y are obtained by averaging $w_k^x(n)$ and $w_k^y(n)$ in past N samples, respectively.

Total decision of verification is achieved by considering all verification results. In our conventional method[5], the total convergence value TC is calculated by averaging eight converged values of the adaptive weight. However, the averaging tends to decrease the total convergence value, so that it degrades the verification rate. In this paper, we introduce sin function as a nonlinear function to enhance the difference between the converged values in the neighborhood of $w_k^v = 0.5$.

The total convergence value TC is calculated as follows.

$$\mathrm{TC} = \frac{1}{8} \left\{ \sum_{q=M-3}^{M} \sin\left(\frac{\pi}{2} \cdot w_k^x\right) + \sum_{q=M-3}^{M} \sin\left(\frac{\pi}{2} \cdot w_k^y\right) \right\} \tag{8}$$

Finally, TC is compared with a threshold value. If TC is larger than the threshold, the input signature is decided as the genuine signature. To consider multiple results leads robustness to our verification method.

4   Experiment and Result

In order to examine the performance of our proposed on-line signature verification method, we carried out experiments of the signature verification. Conditions of the experiment are as follows.

Before signing, the subjects were called upon to practice using the pen tablet for becoming skilled. Four subjects were requested to sign their own signatures 30 times each. When the subjects signed genuine signatures, they were not able to refer their already written signatures. After excluding unusable signatures which have only one sample data in a intra/inter stroke which causes zero division in making of the template *Detail*, we obtained 118 genuine signatures. Moreover, five genuine signatures for each subject were used as the template and the rest 98 genuine signatures were used for verification. On the other hand, five subjects were required to counterfeit the genuine signature 10 times each, so that 200 forgeries were prepared in total. The forgers were permitted to trace the genuine signature by putting the paper to which the signature was written over it. In order to obtain fully convergence of the adaptive weight, the number of iterations was set to 100 thousands. For reference, we examined the time-domain method in which the time-varying signals of x and y components in the input signature were compared with those of five genuine signatures each other by using the DP matching. The total DP distance was calculated by averaging obtained ten DP distances and the total decision of verification was achieved by whether the total DP distance was larger than a threshold value. If the total DP distance was smaller than the threshold, the input signature was decided to be the genuine signature.

Figure 4 shows the False Rejection Rate (FRR) and the False Acceptance Rate (FAR) versus a threshold value. (a) is by the proposed method and (b) is by the time-domain method. In general, verification performance is estimated by the Equal Error Rate (EER) where the FRR and the FAR are the same. The EER in the proposed method was about 4% when the threshold value was about 0.4. On the other hand, the EER was about 5% in [5] and the EER in the time-domain method was about 15%. These results mean that the verification rate 96% was achieved by using only the pen-position component even though a forger traces a genuine signature. Although the number of signature data is not enough, the comparison with the time-domain method shows that the proposed method is effective to improve verification rate in the on-line signature verification.

(a) Proposed   (b) Time-domain
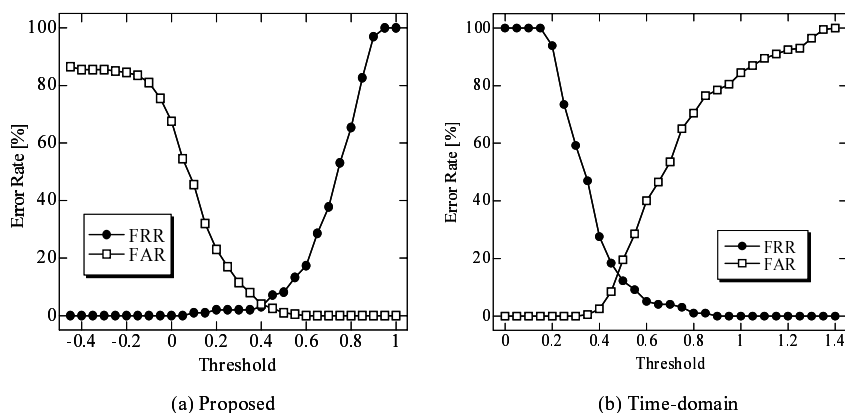
Fig. 4. Verification results

5   Conclusion

We proposed the on-line signature verification method based on the DWT and the adaptive signal processing. Especially, a nonlinear function was introduced to enhance each verification result. Experimental results showed that the verification rate of 96% was achieved by using only the pen-position component. Comparing with the results by the time-domain method, the verification rate was improved by about 11%.

For improving verification performance, we examine to adjust weighting of the converged value instead of the averaging. In addition, for reducing the FRR, it is studied in future to cope with variation in the genuine signature.

Acknowledgements

References

[1] A. Jain, R. Bolle and S. Pankanti, BIOMETRICS Personal Identification in Networked Society, Kluwer Academic Publishers, Massachusetts, 1999.   584
[2] Y. Sato and K. Kogure, "Online Signature Verification Based on Shape, Motion, and Writing Pressure," *Proc. of 6th International Conference on Pattern Recognition*, pp.823-826, 1982.   584
[3] M. Yoshimura, Y. Kato, S. Matsuda, and I. Yoshimura, "On-line Signature Verification Incorporating the Direction of Pen Movement," IEICE Trans., vol.E74, no.7, pp.2083-2092, July 1991.   584

[4] Y. Komiya, T. Ohishi and T. Matsumoto, "A Pen Input On-Line Signature Verifier Integrating Position, Pressure and Inclination Trajectories," IEICE Trans. Inf. & Syst., vol.E84-D, no.7, pp.833-838, July 2001.   584

[5] I. Nakanishi, N. Nishiguchi, Y. Itoh, and Y. Fukui, "On-Line Signature Verification Method Based on Discrete Wavelet Transoform and Adaptive Signal Processing," *Proc. of Workshop on Multimodal User Authentication*, Santa Barbara, USA, pp.207-214, Dec. 2003.   584, 585, 587, 588, 589

[6] G. Strang, T. Nguyen, Wavelet and Filter Banks, Wellesley-Cambridge Press, Massachusetts, 1997.   584, 585

[7] S. Haykin, Introduction to Adaptive Filters, Macmillan Publishing Company, New York, 1984.   584, 587

Speaker Modeling
with Various Speech Representations

Ke Chen

Department of Computation, UMIST
Manchester M60 1QD, United Kingdom
k.chen@umist.ac.uk

Abstract. Although numerous speech representations have been reported to be useful in speaker recognition, there is much less agreement on which speech representation provides a perfect representation of speaker-specific information. In this paper, we charaterize a speaker's identity through the simultaneous use of various speech representations of his/her voice. We present a parametric statistical model, *generalized Gaussian mixture model*, and develop an EM algorithm for parameter estimation. Our approach has been applied to speaker recognition and comparative results on KING corpus demonstrate its effectiveness.

1   Introduction

As one of the most important topics in biometrics, speaker recognition is a difficult problem since speech always changes over time and is affected by numerous factors ranging from environments to speaker's healthy status. An essential problem in speaker recognition is how to characterize a speaker's identity by a computational model based on his/her voice. Therefore, speaker modeling becomes a process that computationally captures a speaker's characteristics from his/her preprocessed speech.

For speech signal processing, numerous spectral representations have been developed by parameterizing the speech spectrum. These representations have been widely used in both speech and speaker recognition. The earlier studies uncover that a speech signal contains mainly three types of information; i.e., linguistic information, speaker-specific information, and information on a specific environment, and so does any spectral representation extracted from the speech signal [1]. An ideal speaker recognition system should be able to discriminate speakers regardless of linguistic information, while a generic speech recognition system would rather decode speech to distill only linguistic information without considering speaker identities. Thus the use of the same spectral representations in both speaker and speech recognition has become an obstacle hindering either of two systems from producing high performance. In most existing work on speaker modeling, as a consequence, a speaker is characterized based on a specific speech representation, which might fail to encode the salient speaker-specific information underlying raw speech under some circumstances.

Gaussian mixture model (GMM) has turned out to be an effective statistic model for encoding the speaker-specific information [2]. As pointed out by Reynolds and Rose [2], each unimodal component of a Gaussian mixture density characterizes an acoustic class underlying a phonetic event and therefore can specify a speaker dependent vocal tract configuration, while the multimodal GMM is modeling a set of speaker dependent acoustic classes. Moreover, a Gaussian mixture density also provides a smooth approximation to the underlying long-term sample distribution of observations obtained from utterances of a specific speaker [2]. Therefore, a GMM may be an adequate tool for speaker modeling.

Unlike previous work, we propose an alternative approach to speaker modeling by the simultaneous use of different speech representations in an optimal way. Inspired by our previous empirical studies, we propose a parametric statistical model, generalized Gaussian mixture model (GGMM), to characterize a speaker identity based on different speech representations. For parameter estimation in the GGMM, we develop an expectation-maximization (EM) algorithm. The proposed speaking modeling approach has been applied to text-independent speaker recognition and comparative results on the KING database demonstrate its effectiveness.

The rest of this paper is organized as follows. Sect. 2 presents the GGMM, and Sect. 3 describes its EM algorithm. Sect. 4 reports simulation results. The last section draws conclusions.

2   Generalized Gaussian Mixture Model

In our earlier work, we empirically investigated how the performance of speaker recognition is influenced by different speech representations through the use of modular neural networks [3, 4, 5]. In those investigations, there is always a common outcome; i.e., for a classifier separately trained on different representations of the same raw speech, its performance on a whole testing set is often *inconsistent* with those on subsets of this testing set when we compare the influence of different representations on speaker recognition based on the identical classifier. A typical example can be seen in [7]. Our finding further suggests that the various speaker-specific information conveyed in different parts of speech should be dynamically specified by different kinds of speech representations.

Inspired by our previous empirical studies, we propose a generalized Gaussian mixture model (GGMM). As illustrated in Fig. 1, K GMMs, where the GMM receiving the input $\boldsymbol{x}_k$, the kth representation, contains N_k component Gaussian densities, are employed to model a speaker's characteristics based on K different representations, while mixing proportions are produced by a linear combination of K proportion generators working on different representations. As a consequence, the overall output of the GGMM is the convex weighted combination of outputs of K GMMs.
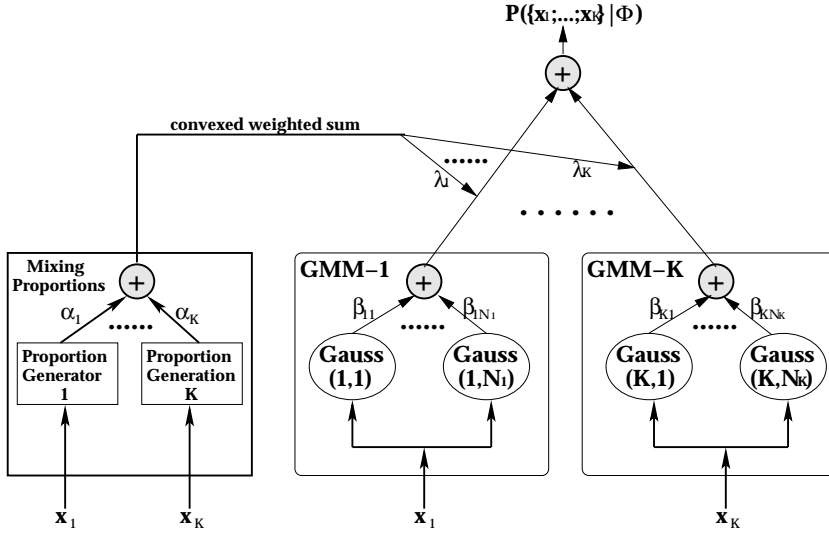
Fig. 1. The schematic diagram of GGMM

By a formal derivation [7], we specify a parametric form of our GGMM as follows:

$$p(\{\boldsymbol{x}_1; \cdots; \boldsymbol{x}_K\}|\boldsymbol{\Phi}) = \sum_{k=1}^{K} \sum_{j=1}^{N_k} \sum_{i=1}^{K} \alpha_i \beta_{kj} b(\boldsymbol{x}_i, \boldsymbol{\omega}_{ik}) G(\boldsymbol{x}_k, \boldsymbol{\mu}_{kj}, \Sigma_{kj}). \qquad (1)$$

where

$$b(\boldsymbol{x}_i, \boldsymbol{\omega}_{ik}) = \frac{\exp(\boldsymbol{\omega}_{ik}^T \boldsymbol{x}_i)}{\sum_{i=1}^{K} \exp(\boldsymbol{\omega}_{ik}^T \boldsymbol{x}_i)} \quad k = 1, \cdots, K.$$

$G(\boldsymbol{x}_k, \boldsymbol{\mu}_{kj}, \Sigma_{kj})$ is a Gaussian density function where $\boldsymbol{\mu}_{kj}$ and Σ_{kj} are the mean vector and the covariance matrix of the jth Gaussian component in the kth GMM. α_i and β_{kj} are coefficients where $0 \leq \alpha_i \leq 1$ and $0 \leq \beta_{kj} \leq 1$.

3   EM Algorithm

By the maximum likelihood principle, we develop an EM algorithm to fit our GGMM structure and the detailed derivation can be found in [7].

In the E step, we need to estimate the expectation values of random missing indicator variables based on the observable data $\mathcal{X}$ and the current parameter values of the GGMM, $\boldsymbol{\Phi}^{(s)}$. In the context of the GGMM, the following posterior probabilities, denoted by $h_i^{(t)}(\boldsymbol{\Phi}^{(s)}), h_{ik}^{(t)}(\boldsymbol{\Phi}^{(s)})$ and $h_{kj}^{(t)}(\boldsymbol{\Phi}^{(s)})$, are estimated by

$$h_i^{(t)}(\boldsymbol{\Phi}^{(s)}) = \frac{\alpha_i^{(s)} \sum_{k=1}^{K} \sum_{j=1}^{N_k} \beta_{kj}^{(s)} b(\boldsymbol{x}_i^{(t)}, \boldsymbol{\omega}_{ik}^{(s)}) G(\boldsymbol{x}_k^{(t)}, \boldsymbol{\mu}_{kj}^{(s)}, \boldsymbol{\Sigma}_{kj}^{(s)})}{\sum_{k=1}^{K} \sum_{j=1}^{N_k} \sum_{i=1}^{K} \alpha_i^{(s)} \beta_{kj}^{(s)} b(\boldsymbol{x}_i^{(t)}, \boldsymbol{\omega}_{ik}^{(s)}) G(\boldsymbol{x}_k^{(t)}, \boldsymbol{\mu}_{kj}^{(s)}, \boldsymbol{\Sigma}_{kj}^{(s)})}, \qquad (2a)$$

$$h_{ik}^{(t)}(\boldsymbol{\Phi}^{(s)}) = \frac{\alpha_i^{(s)} b(\boldsymbol{x}_i^{(t)}, \boldsymbol{\omega}_{ik}^{(s)}) \sum_{j=1}^{N_k} \beta_{kj}^{(s)} G(\boldsymbol{x}_k^{(t)}, \boldsymbol{\mu}_{kj}^{(s)}, \boldsymbol{\Sigma}_{kj}^{(s)})}{\sum_{k=1}^{K} \sum_{j=1}^{N_k} \sum_{i=1}^{K} \alpha_i^{(s)} \beta_{kj}^{(s)} b(\boldsymbol{x}_i^{(t)}, \boldsymbol{\omega}_{ik}^{(s)}) G(\boldsymbol{x}_k^{(t)}, \boldsymbol{\mu}_{kj}^{(s)}, \boldsymbol{\Sigma}_{kj}^{(s)})}, \tag{2b}$$

$$h_{kj}^{(t)}(\boldsymbol{\Phi}^{(s)}) = \frac{\beta_{kj}^{(s)} G(\boldsymbol{x}_k^{(t)}, \boldsymbol{\mu}_{kj}^{(s)}, \boldsymbol{\Sigma}_{kj}^{(s)}) \sum_{i=1}^{K} \alpha_i^{(s)} b(\boldsymbol{x}_i^{(t)}, \boldsymbol{\omega}_{ik}^{(s)})}{\sum_{k=1}^{K} \sum_{j=1}^{N_k} \sum_{i=1}^{K} \alpha_i^{(s)} \beta_{kj}^{(s)} b(\boldsymbol{x}_i^{(t)}, \boldsymbol{\omega}_{ik}^{(s)}) G(\boldsymbol{x}_k^{(t)}, \boldsymbol{\mu}_{kj}^{(s)}, \boldsymbol{\Sigma}_{kj}^{(s)})}. \tag{2c}$$

The M step decomposes the original optimization problem into the following separate maximization sub-problems based on (2). For parameters in the GMM receiving the input $\boldsymbol{x}_k^{(t)}$, two optimization sub-problems are

$$\Theta_{kj}^{(s+1)} = \arg\max_{\Theta_{kj}} \sum_{t=1}^{T} h_{kj}^{(t)}(\boldsymbol{\Phi}^{(s)}) \log G(\boldsymbol{x}_k^{(t)}, \boldsymbol{\mu}_{kj}, \Sigma_{kj}), \tag{3}$$

where $\Theta_{kj} = (\boldsymbol{\mu}_{kj}, \Sigma_{kj})$ is a collective notation of parameters of the jth Gaussian density in the GMM receiving $\boldsymbol{x}_k^{(t)}$, and

$$\beta_{kj}^{(s+1)} = \arg\max_{\beta_{kj}} \sum_{t=1}^{T} \sum_{j=1}^{N_k} h_{kj}^{(t)}(\boldsymbol{\Phi}^{(s)}) \log \beta_{kj} \quad s.t. \quad \sum_{j=1}^{N_k} \beta_{kj} = 1, \ \beta_{kj} \geq 0. \tag{4}$$

Similarly, two optimization sub-problems with respect to mixing proportions are

$$\Omega_i^{(s+1)} = \arg\max_{\Omega_i} \sum_{t=1}^{T} \sum_{k=1}^{K} h_{ik}^{(t)}(\boldsymbol{\Phi}^{(s)}) \log b(\boldsymbol{x}_i^{(t)}, \boldsymbol{\omega}_{ik}), \tag{5}$$

and

$$\alpha_i^{(s+1)} = \arg\max_{\alpha_i} \sum_{t=1}^{T} \sum_{i=1}^{K} h_i^{(t)}(\boldsymbol{\Phi}^{(s)}) \log \alpha_i \quad s.t. \quad \sum_{i=1}^{K} \alpha_i = 1, \ \alpha_i \geq 0. \tag{6}$$

Analytic solutions to optimization sub-problems in (3) and (4) are available [8] as follows:

$$\boldsymbol{\mu}_{kj}^{(s+1)} = \frac{1}{\sum_{t=1}^{T} h_{kj}^{(t)}(\boldsymbol{\Phi}^{(s)})} \sum_{t=1}^{T} h_{kj}^{(t)}(\boldsymbol{\Phi}^{(s)}) \boldsymbol{x}_k^{(t)}, \tag{7}$$

$$\Sigma_{kj}^{(s+1)} = \frac{1}{\sum_{t=1}^{T} h_{kj}^{(t)}(\boldsymbol{\Phi}^{(s)})} \sum_{t=1}^{T} h_{kj}^{(t)}(\boldsymbol{\Phi}^{(s)}) [\boldsymbol{x}_k^{(t)} - \boldsymbol{\mu}_{kj}^{(s+1)}][\boldsymbol{x}_k^{(t)} - \boldsymbol{\mu}_{kj}^{(s+1)}]^T. \tag{8}$$

When Σ_{kj} is diagonal where diagonal elements are $\sigma_{kj}^{(s+1)}(1), \cdots, \sigma_{kj}^{(s+1)}(m_k)$, the analytic solution in (8) is simplified by

$$[\boldsymbol{\sigma}_{kj}^{(s+1)}(m)]^2 = \frac{1}{\sum_{t=1}^{T} h_{kj}^{(t)}(\boldsymbol{\Phi}^{(s)})} \sum_{t=1}^{T} h_{kj}^{(t)}(\boldsymbol{\Phi}^{(s)}) \left[[\boldsymbol{x}_k^{(t)}(m)]^2 - [\boldsymbol{\mu}_{kj}^{(s+1)}(m)] \right]^2, \tag{9}$$

where $\boldsymbol{\sigma}_{kj}^{(s+1)}(m)$, $\boldsymbol{x}_k^{(t)}(m)$ and $\boldsymbol{\mu}_{kj}^{(s+1)}(m)$ are the mth element of the vectors, $\boldsymbol{\sigma}_{kj}^{(s+1)}$, $\boldsymbol{x}_k^{(t)}$ and $\boldsymbol{\mu}_{kj}^{(s+1)}$, respectively. As suggested by Reynolds and Rose [2], we use only such a diagonal covariance matrix in our simulations. Finally, the analytic solution to the optimization sub-problem in (4) is also available by

$$\beta_{kj}^{(s+1)} = \frac{1}{T}\sum_{t=1}^{T} h_{kj}^{(t)}(\boldsymbol{\Phi}^{(s)}). \tag{10}$$

For mixing proportions, the optimization sub-problem in (5) is analytically insolvable, while the analytic solution to the optimization sub-problem in (6) is

$$\alpha_i^{(s+1)} = \frac{1}{T}\sum_{t=1}^{T} h_i^{(t)}(\boldsymbol{\Phi}^{(s)}). \tag{11}$$

Due to the nature of a multinomial logit model underlying each mixing proportion generator, the optimization sub-problem in (5) is an iteratively re-weighted least squares (IRLS) problem. In our earlier work [6], we developed a Newton-Raphason algorithm for solving the IRLS problem, which is used in our simulations.

Thus the parameter values in $\boldsymbol{\Phi}$ obtained by the EM algorithm constitute a speaker model based on different speech representations.

4   Simulation Results

The KING corpus is a benchmark English acoustic corpus designed especially for text-independent speaker recognition. It consists of wide-band (WB) and narrow-band (NB) sets. Similar to those used in [2], our system adopts the following preprocessing for text-independent speaker recognition: a) pre-emphasizing with the filter response $H(z) = 1 - 0.95z^{-1}$, b) 32ms Hamming windowing without overlapping, c) removing the silence and unvoiced part of speech in terms of short-term average energy, and d) extracting different feature vectors from each short-term frame. In our simulations, three different feature extraction methods are employed. Thus, three different feature vectors can be extracted from one short-term frame; i.e. 19-order linear predictive coefficients (LPC), 16-order perceptual linear prediction (PLP) and 19-order Mel-scaled cepstral coefficient (MCC) vectors. To form a composite feature vector, we simply lump the above three feature vectors together, which leads to a 54-order composite feature vector for one short-term frame. In simulations on the NB set, the mean subtraction technique [2] is also used in preprocessing for robustness.

In our simulations, the GGMM is used to characterize a speaker identity based on different representations. For the same task, the ordinary GMM is also used for comparison (denoted by GMM-LPC, GMM-PLP, GMM-MCC) where a GMM is trained on either an individual representation or a composite representation (i.e., GMM-C). In addition, the combination of GMMs by the linear opinion pool (i.e., LOP/GMM) [7] is performed for further comparison.

Table 1. EERs (%) of different speaker modeling methods as the length of testing segments is eight seconds

Training Duration	Data Set	Speaker Model					
		GMM-LPC	GMM-PLP	GMM-MCC	GMM-C	LOP/GMM	GGMM
40 s	WB	11.6±2.5	11.8±4.1	11.1±3.1	9.6±2.8	8.7±2.5	7.6±2.7
70 s	WB	6.8±2.2	6.7±2.3	6.2±2.8	5.8±3.3	5.1±2.3	4.6±2.1
70 s	NB	17.5±3.7	16.6±3.4	16.9±4.7	15.9±4.1	14.1±3.7	12.6±3.5

Our simulations adopt multiple trials for obtaining the reliable performance. In each trial, two sessions are randomly selected from ten recording sessions for training and the remaining eight sessions are employed for testing. For the WB set, we further investigate the performance of different training speech durations. Along with a long training duration (70 seconds) from two sessions, a short training duration (40 seconds) from a single session is also employed to train each speaker model. Given that a short-time duration likely suffers from severe mismatch, we expect that such simulation results manifest how a speaker modeling method performs against mismatch. For different training durations, we use different numbers of component densities in speaker models; 32 and 64 component for GMMs as well as $12 \times 3 = 36$ and $20 \times 3 = 60$.

Equal error rate (EER), where the false rejection rate is equal to the false acceptance rate, is a common posterior measurement for testing the discriminant capability of a speaker model in speaker verification. As shown in Table 1, the statistics (mean and standard deviation) indicate the overall EERs produced by different speaker models in multiple trials. It is evident from Table 1 that the joint use of different representations results in better performance on both the WB and the NB sets and our GGMM consistently yields the best performance regardless of training durations. In particular, our approach significantly outperforms others in the presence of severe mismatch (i.e., on the NB set) and limited training data (i.e., training by a 40-second duration), which demonstrates the effectiveness of the proposed soft-competition scheme by exploiting different representations for robustness.

Unlike speaker verification, speaker identification needs to consult all speaker models registered in a system for decision-making. Table 2 lists the performance

Table 2. Identification rates (%) on different trials

Training Duration	Data Set	Speaker Model					
		GMM-LPC	GMM-PLP	GMM-MCC	GMM-C	LOP/GMM	GGMM
40 s	WB	80.6±4.2	78.9±6.1	80.3±4.7	81.4±6.8	83.1±4.7	84.9±4.3
70 s	WB	90.1±4.5	89.7±4.2	90.2±4.9	90.8±4.3	91.4±4.7	92.3±3.9
70 s	NB	63.9±6.8	64.8±7.5	65.2±7.3	66.1±6.2	67.2±6.7	73.6±6.8

of different speaker modeling methods. It is observed that higher identification rates are obtained as different speech representations are used in speaker models in comparison with those models trained on an individual representation, and our GGMM always yields the best performance, which is consistent with the performance of speaker verification.

In summary, the above results of speaker recognition demonstrate the usefulness of our speaker modeling approach via different speech representations although doing so may slightly incur higher computational loads. By the optimal use of different representations, our GGMM yields the favorite performance especially in the presence of severe mismatch and limited training data.

5   Conclusion

We have presented a novel approach to speaker modeling, which exploits different speech representations in an optimal way. Our studies here demonstrate that the favorite and robust performance can be obtained by the optimal use of different speech representations, which paves a new path for exploring and exploiting potential speaker-specific information sources for speaker recognition. In addition, the proposed GGMM architecture can be generalized by the use of other component models rather than GMMs and have been extended to supervised pattern classification tasks based on different speech representations [3]. Furthermore, the simultaneous use of different speech representations in an optimal way suggests an alternative way to handle a class of dynamic pattern recognition problems where patterns always change over time due to various factors, e.g. speaker's voice and personal signature. Since most of biometric features can be regarded as dynamic patterns, our approach would offer an alternative solution to other biometric problems as well.

References

[1] X. D. Huang, A. Acero and H. W. Hon, *Spoken Language Processing,* Wiley Inter-Science, 2000. 592

[2] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture models," *IEEE Transactions on Speech Audio Processing*, vol. 3, no. 1, pp. 72-83, 1995. 593, 596

[3] K. Chen, "A connectionist method for pattern classification with diverse features," *Pattern Recognition Letters*, vol. 19, no. 7, pp. 545-558, 1998. 593, 598

[4] K. Chen, L. Wang, and H. Chi, "Methods of combining multiple classifiers with different features and their applications to text-independent speaker identification," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 11, no. 3, pp. 417-445, 1997. 593

[5] K. Chen, D. Xie, and H. Chi, "A modified HME architecture for text-dependent speaker identification," *IEEE Transactions on Neural Networks*, vol. 7, no. 5, pp. 1309-1313, 1996. 593

[6] K. Chen, L. Xu, and H. Chi, "Improved learning algorithms for mixture of experts in multiclass classification," *Neural Networks*, vol. 12, no. 9, pp. 1225-1252, 1999. 596

[7] K. Chen, On the use of different speech representations for speaker modeling, *IEEE Transactions on Systems, Man and Cybernetics (Part C)*, vol. 34, 2004. (Special issue on Biometric Systems, in press) 593, 594, 596

[8] G. McLanchlan and D. Peel, *Finite Mixture Models,* Wiley Inter-Science, 2000. 595

Using Otoacoustic Emissions as a Biometric

Matthew A. Swabey[1], Stephen P. Beeby[1],
Andrew D. Brown[1], and John E. Chad[2]

[1] School of Electronics and Computer Science
University of Southampton, Southampton SO17 1BJ, UK
{mas01r,spb,adb}@ecs.soton.ac.uk
[2] School of Biomedical Sciences
University of Southampton, Southampton, SO17 7PX, UK
jchad@soton.ac.uk

Abstract. This paper presents research into using otoacoustic emissions (OAE) as a biometric. OAE are a group of sounds emitted by the cochlea of the ear. The main types are spontaneous OAE which are continuously emitted and the transient and distortion product types emitted after stimulation. In this paper the characteristics of OAE are discussed to identify the most suitable emission for use as a physiological biometric. The uniqueness of transient OAE are characterised from a data set of 2009 random individuals by a closed rank test. Details of a low-cost portable prototype OAE acquisition system developed at Southampton University are discussed.

1 Introduction

Otoacoustic Emissions (OAE) are a sound wave originating from an active process naturally occurring in the cochlea of the ear. OAE have been studied by the medical profession since their detection in 1977 by David Kemp [1],[2] as a tool for clinical investigation into the ear's health and for analysis of its operation. OAE are known to exhibit significant differences, for instance in the emission frequencies and numbers of spontaneous otoacoustic emissions between the genders [3] and between people of different ethnic origins [4], giving credibility to speculation that OAE are unique.

2 Otoacoustic Emissions

OAE originate from the human body's solution to improving the precision and threshold of hearing. Incoming sounds are mechanically separated into their component frequencies. As they travel up the cochlea these frequencies peak and stop at a distance proportional to their frequency. In a purely mechanical system energy loss and damping forces in the cochlea would cause quiet sounds to be beneath the detection threshold. The ears active process improves this by adding power to the sound as it travels. The OAE is a byproduct of this added power which can be reflected by structural individuality's within the ear, and

passed back through the middle ear and out of the eardrum. The generation and operation of OAE is currently not fully characterised or understood. OAE can be classified into three broad types:

Spontaneous Otoacoustic Emissions (SOAE). A spontaneous otoacoustic emission is a pure tone at a stable frequency that is continually emitted attributed to positive feedback in the active process. The exact method of generation is unknown however it is suggested that SOAE derive from structural irregularities within the cochlea which act as an energy reflector.

SOAE are currently thought to be present in approximately 75% of the population [5],[6],[7]. A SOAE is a single pure tone at a fixed frequency up to 2kHz and a fixed amplitude. The maximum observed amplitude is approximately 20dB SPL (sound pressure level) however the mean observed amplitude for adults is between -3 and 0dB SPL [8],[9]. An ear may emit more than one independent SOAE simultaneously.

Transient Evoked Otoacoustic Emissions (TEOAE). TEOAE are typically stimulated with a white noise pulse of approximately 80dB SPL (peak) with a duration of 2ms and the response emerges shortly afterwards, starting at 3ms and lasting till 20ms. The response is a stable waveform with a fixed latency to the stimulus, where the low frequencies (up to 700Hz) emerge first, then the medium frequencies and the higher frequencies (up to 6kHz) emerging last. These are now substantively proved to be a byproduct of normal hearing [10] and so are prevalent in the whole population.

Distortion Product Otoacoustic Emissions (DPOAE). These are evoked by applying two continuous sine waves at similar frequencies. A set of sine waves are then 'produced' at set frequencies defined by the following equation $nf_1 - mf_2$ where f_1 is the frequency of the lower of the two stimulus tones, f_2 the higher and n and m positive integers. These 'distortion products' magnitude is greatest for $n = 2$ and $m = 1$ and decays with increasing n and m. A DPOAE from a female adult's ear stimulated by two sine waves at 1.820kHz and 2.196kHz with magnitudes 65dB SPL and 55dB SPL had the $2f_1 - f_2$ component as the largest response, at 1.444kHz with a magnitude of 10dB SPL. The $3f_1 - 2f_2$ and $2f_2 - f_1$ components were also detected at -2.5dB SPL and -16.3dB SPL respectively. DPOAE have been recorded at frequencies as high as 7kHz however most adult DPOAE are recorded in the 1-2kHz range. They are also a characteristic of the hearing process.

3   Otoacoustic Emissions as a Biometric

In order for OAE to be used as a physiological biometric, they should be robust and relatively easy to capture. Of the three types discussed above the least suitable is the spontaneous otoacoustic emission. It is currently detected in only

75% of the population and this requires laboratory conditions and sophisticated equipment. DPOAE and TEOAE are more suitable as they are stronger responses, typically 15dB SPL at maximum compared to a typical SOAE at -3 to 0dB SPL.

When comparing hardware requirements for DPOAE vs. TEOAE the TEOAE is more straightforward. Acquiring a DPOAE requires the two frequencies to be swept across a frequency range and the magnitude of the largest component recorded. This technique requires a probe with two speakers and a microphone with accompanying sine wave generators to provide the stimulus. In order to distinguish the response from the stimuli an analogue to digital converter (ADC) with a large dynamic range (typically 16 bits) is conventionally required to provide the analysis software with suitable data. TEOAE, however, customarily use a square pulse stimulus requiring a single speaker and a single microphone to acquire the information. Since the stimulus dies away before the TEOAE emerges a large dynamic range ADC is not required.

All OAE are affected by hearing loss and are not present in the profoundly deaf. However, it is not known what level of hearing loss will prevent OAE being used as a biometric. Based on a study of hearing loss in the UK [11] the worst case, where any damage to hearing prevents the OAE's use as a biometric, 15% of the population would be excluded. The best case excludes 3.1%. As hearing loss affects the OAE having partial damage could lead to improved chances of identification due to an unusual OAE. This will be the subject of further investigation,

TEOAE were selected for the initial research due to the reduced hardware requirements, but DPOAE also offers excellent potential as a biometric.

3.1   Uniqueness

In order for an OAE to be used as a biometric a suitable level of uniqueness to an individual must be demonstrated. To estimate the suitability of TEOAE for further study an analysis was performed on subset of a large dataset collected in the UK for medical purposes (fully described Lutman *et al.* [12]). This subset consisted of the non-linear responses of 2009 individual's right ears computed from TEOAE. The non-linear response is the combination of an individual's TOAE recorded at several increasing intensities and then manipulated to attenuate any linear components, maximising the information from the OAE as opposed to characteristics of the recording process. Manual examination of a small random sample of the data indicated the power spectra showed very few similarities and would be suitable for a simple analysis.

A closed rank calculation of the euclidean distance of the frequency components of the power spectra was chosen to quantify the separation between individuals. This analysis also offers the attraction that working in the frequency domain gives immunity to phase difference and provides a single value for comparison. The separation analysis was performed on one of the two non-linear responses of the subjects right ears. The power spectra of the nonlinear component is extracted by performing a discrete fourier transform and calculating the

Table 1. The separation expressed as a percentage computed for proportions of the data set

Number of Samples	Percentage of Data Set	Separation
1005	50%	$\geq 12.57\%$
1909	95%	$\geq 5.52\%$
1989	99%	$\geq 3.68\%$

absolute value of every complex component. The non-linear responses are 256 samples long so after the fourier transform there are 128 extracted components (Nyquist's Theorem). The first component, representing the constant offset, is set to 0. Treating the 128 components as a 128 dimensional vector in euclidean space we can simply compute the separation between them using Equation 1 which gives the euclidean separation S between the two power spectra, a and b.

$$S = \sqrt{((a_1 - b_1)^2 + (a_2 - b_2)^2 + \ldots + (a_n - b_n)^2)} \tag{1}$$

The individual distance is then compared to the rest of the individuals in the dataset and the distance to the closest neighbour recorded.

The data was held in a MySQL database and the mathematical analysis was performed using a C++ program implemented with the Fastest Fourier Transform in the West (FFTW) discrete fourier transform library [13].

3.2 Results and Discussion of Uniqueness Analysis

The results of the analysis are detailed in Table 1 showing the separation of statistically significant proportions of the data set. This shows that, if the overall system variation (all sources of error) is kept below 3.68%, we can identify 1989 individuals from the 2009 in the set (i.e. 99%). In the analysis of the whole data set the smallest separation was 1.26% meaning if the system noise was further reduced to less than 1.26% we could identify the entire set of 2009 individuals.

Figure 1 shows the frequency of each percentage separation. As discussed above note that the distribution does not extend to zero. Since this analysis relies on TEOAE and we have no results very close to zero this confirms that everyone in this sample set had a TEOAE.

This exercise has successfully discriminated between 2009 individuals using a computationally non-intensive analysis. Due to the large size of the sample set we are confident that TEOAE can be used as the basis of a successful biometric system for distinguishing between individuals.

3.3 Portable Low Cost Acquisition System

TEOAE are normally collected by a probe sealed into the inner ear. This sealing process is done by medically trained personnel selecting the correct sterile foam tip and fitting the probe into the ear canal. The probe contains the necessary
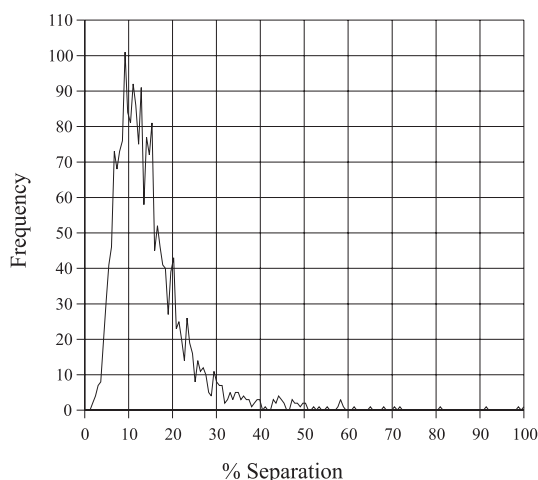
Fig. 1. Distribution of the Separation
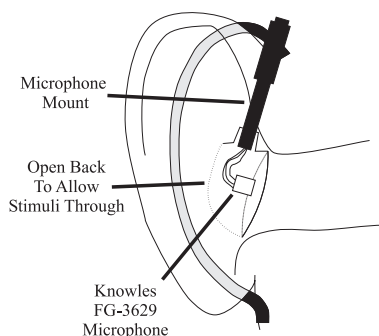
Fig. 2. Schematic of the Microphone Mount on a Left Ear

speakers and microphones to stimulate and capture the emission. A digital signal processor system or the host PC emits the stimulus then processes the data and stores the results. To detect the TEOAE a number of signal processing techniques are used to improve the signal to noise ratio as it is such a small signal (20dB SPL is equivalent to 10^{-12}W).

There has been previous work on probes not sealed to the ear canal. Work done with guinea pigs using an open system with the stimulus directed at their ears and a microphone tube fixed at the entrance of the ear canal has successfully stimulated and captured TEOAE and DPOAE [14]. Recently a paper on work done with humans [15] using a conventional OAE probe held to the ear using a wire frame with a soft tube that extended into the canal but not sealed into the canal was performed at the University of Southampton Institute of Sound and Vibration Research (ISVR).

Further work within the University of Southampton School of Electronics and Computer Science has also successfully demonstrated a superior on the ear approach. Figure 2 shows the schematic of a novel probe front end that is mounted within a set of conventional headphones that cover a persons ears and seal to the skull using a foam / rubber strip as a stimulus source this probe / stimulus source has proved effective in practise.

The DSP functionality required is provided by a Pentium III based personal computer (PC) hosting a National Instruments AT-MIO-16E-10 DAQ (data acquisition) card [16] running custom software. This functionality is fully transferable to a standard laptop system, allowing easy transport and TEOAE have been successfully captured using this system.

4   Conclusions

TEOAE have been identified as offering the most likely OAE to act as a successful biometric characteristic. They are universal in the normal hearing population, have one of the strongest detectable responses of OAE.

An analysis of a data set of 2009 random individuals based on the euclidean distance of the power spectra showed that 99% of the data set was seperated by 3.68%. This therefore would allow the correct identification of 99% of the individuals if the system error is kept less than 3.68%. A more optimised analysis will increase this separation and improve discrimination.

We have produced a portable and low cost TEOAE acquisition system using commercial of the shelf equipment and standard components which overcomes the requirements for medically trained personnel and the need for special fitting. We anticipate this offers the potential for a commercial biometric system in the future.

References

[1] Kemp, D.: Acoustic resonances originating inside the cochlea. British Society of Audiology short papers meeting. (1978) 600
[2] Kemp, D.: Stimulated acoustic Emissions from within the human auditory system. J. Acoust. Soc. Am. **64** (1978) 1386–1391 600
[3] Bilger, R., Matthies, M., Hammel, D. and Demorest, M.: Genetic-implications of gender differences in the prevalence of spontaneous otoacoustic emissions. J. Spch. Hearing Res. **33** (1990) 418–432 600
[4] Whitehead, M., Kamal, N., Lonsbury-Martin, B. and Martin, G.: Spontaneous otoacoustic emissions in different racial groups. Scand. Audiol. **22** (1993) 3–10 600
[5] Zhang, T. and Penner, M.: A new method for the automated detection of spontaneous otoacoustic emissions embedded in noisy data. Hear. Res. **117** (1998) 107–112 601
[6] Pasanen, E. and McFadden, D.: An automated procedure for identifying spontaneous otoacoustic emissions. J. Acoust. Soc. Am. **108** (2000) 1105–1116 601

[7] Talamadge, C., Long, G., Murphy, W. and Tubis, A.: New off-line method for detecting spontaneous otoacoustic emissions in human subjects. Hear. Res. **71** (1993) 170–182   601

[8] Burns, E., Arehart, K. and Campbell, S.: Prevalence of spontaneous otoacoustic emissions in neonates. J. Acoust. Soc. Am. **91** (1992) 1571–1575   601

[9] Penner, M., Glotzbach, L. and Huang, T.: Spontaneous otoacoustic emissions: measurement and data. Hear. Res. **68** (1993) 229–237   601

[10] Kapadia, S. and Lutman, M.: Are normal hearing thresholds a sufficient condition for click-evoked otoacoustic emissions? J. Acoust. Soc. Am. **101** (1997) 3566–3576   601

[11] Davis, A.: Hearing in Adults. Whurr Publishers Ltd. London, ISBN 1-897635-40-0 (1995)   602

[12] Lutman, M., Davis, A., Fortnum, H. and Wood, S.: Field Screening of Targeted Neonatal Hearing Screening by Transient-Evoked Otoacoustic Emissions. Ear. Hear. **18 No. 4** (1997) 265–276   602

[13] Frigo, M. and Johnson, S.: The Fastest Fourier Transform in the West. http://fftw.org (2003)   603

[14] Withnell, R., Kirk, D. and Yates, G.: Otoacoustic emissions measured with a physically open recording system. J. Acoust. Soc. Am. **104** (1998) 350–355   604

[15] Meritt, S. and Kapadia, S.: Otoacoustic emissions in the unoccluded human ear canal. Int. J. Audiol. **42** (2003) 373   604

[16] National Instruments: DAQ AT-MIO E Series User Manual. Part Number 320517D-01 (May 1995)   605

A Pruning Approach for GMM-Based Speaker Verification in Mobile Embedded Systems

Cheung Chi Leung[1], Yiu Sang Moon[1], and Helen Meng[2]

[1] Department of Computer Science and Engineering
The Chinese University of Hong Kong, Shatin, N.T., Hong Kong SAR, China
{ccleung,ysmoon}@cse.cuhk.edu.hk
[2] Department of Systems Engineering and Engineering Management
The Chinese University of Hong Kong, Shatin, N.T., Hong Kong SAR, China
hmmeng@se.cuhk.edu.hk

Abstract. This paper presents a pruning approach for minimizing the execution time in the pattern matching process during speaker verification. Specifically, our speaker verification system uses mel-frequency cepstral coefficient feature extraction and GMM-based pattern matching techniques. The pruning approach may make the verification decision by considering only a special portion of the testing utterance. Our experimental investigation shows that the pruning approach can reduce the execution time of the pattern matching process by as much as 50% in some verification trials. Together with other optimization strategies, this work enables speaker verification to become feasible in mobile embedded systems.

1 Introduction

Speaker verification is the task of verifying a person's claimed identity based on his/her voice. Most speaker verification systems are originally designed to run on desktop computers. However, recent technological and market trends motivate us to develop speaker verification for embedded systems. Potential applications include time-attendance systems, access control devices and security modules in electronic appliances.

To enable real-time speaker verification in a mobile embedded system, the effect of parameters involved in feature extraction [1] and the execution time optimization techniques [2], including floating-point code conversion and optimization in transcendental functions, have been studied. Here, we further optimize the execution time in the pattern matching process by a pruning approach, which is capable of making verification decision when only a portion of the testing utterance is considered.

The use of pruning in speaker identification is not entirely new. As the execution time in speaker identification becomes too lengthy when the number of registered speakers is large, pruning of unlikely speaker model candidates has been applied in [3]. The performance of speaker verification is seldom hindered by its execution time when the applications are designed to run on desktop computers. However, the

limited amount of computational resources of mobile embedded processors motivates us to explore the use of pruning for speaker verification.

In this paper, we focus on speaker verification systems based on the Mel-frequency Cepstral Coefficient (MFCC) feature extraction [4] and the Gaussian Mixture Models (GMM)-based pattern matching [5] techniques. The remainder of this paper is organized as follows: Section 2 outlines the mechanism of the conventional GMM-based pattern matching. Section 3 describes our pruning approach in speaker verification and how this approach differs the one adopted in speaker identification. Section 4 presents experiments showing how our pruning approach can run faster than the conventional approach while maintaining verification performance. Finally, Section 5 gives a brief conclusion.

2   Conventional GMM-Based Pattern Matching

The process of feature extraction segments an input utterance of a speaker into F frames by a window progressing at a certain frame rate and F mel-frequency cepstral feature vectors $\{\underline{x}_1, \underline{x}_2, \underline{x}_3 \cdots \underline{x}_F\}$ are computed from the F frames. The feature vectors are modeled by a mixture of Gaussian densities to form the speaker model. This mixture of Gaussian densities is a weighted sum of M component densities,

$$p(\underline{x}_t \mid \lambda) = \sum_{i=1}^{M} w_i \times g_i(\underline{x}_t), \quad 1 \le t \le F, \tag{1}$$

in which $\underline{x}_t$ is a feature vector at time t, λ is the speaker model, w_i's are mixture weights, and g_i's are the Gaussian densities, i.e.

$$g_i(\underline{x}) = \frac{1}{(2\pi)^{\frac{N}{2}} \left| \Sigma_i \right|^{\frac{1}{2}}} e^{-\frac{1}{2}(\underline{x} - \underline{\mu}_i)^T \Sigma_i^{-1} (\underline{x} - \underline{\mu}_i)}, \quad 1 \le i \le M, \tag{2}$$

where N is the dimension of the feature vector, $\underline{\mu}_i$ is an N-dimensional mean vector, and Σ_i is an $N \times N$ covariance matrix.

During pattern matching, the F feature vectors $\{\underline{x}_1, \underline{x}_2, \underline{x}_3 \cdots \underline{x}_F\}$ extracted from a testing utterance are compared with the claimant's model and a world model [6]. A normalized likelihood score is calculated as

$$L = \frac{1}{F} \sum_{t=1}^{F} \log[p(\underline{x}_t \mid \lambda)] - \log[p(\underline{x}_t \mid \lambda_g)], \tag{3}$$

where λ and λ_g are the claimant's model and the world model respectively. This normalized likelihood score is compared with a pre-defined threshold T to make a verification decision Θ, which is defined as

$$\Theta = \begin{cases} \text{The utterance is from the claimed speaker,} & \text{if } L > T. \\ \text{The utterance is from an impostor,} & \text{if } L \le T. \end{cases} \tag{4}$$
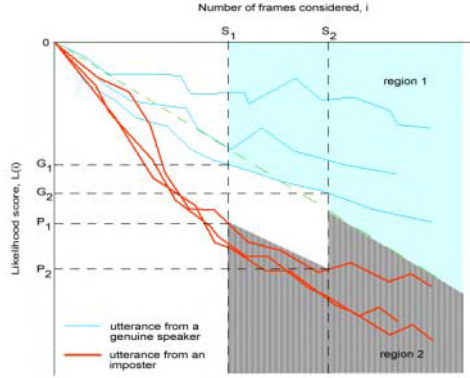
Fig. 1. Likelihood scores of different kind of utterances

3   The Pruning Approach in GMM-Based Pattern Matching

In our approach, instead of comparing all F feature vectors with the claimant's model and the world model, the F feature vectors are divided into N stages. Stage 1 consists of the first feature vector to the $S_1{}^{th}$ feature vector. Stage 2 consists of the $(S_1+1)^{th}$ feature vector to the $S_2{}^{th}$ feature vector. We can extrapolate accordingly such that the last stage N consists of the $(S_{N-1}+1)^{th}$ feature vector to the Fth feature vector. In the ith stage, only the feature vectors in this stage and previous stages are compared with the two GMM models and a normalized likelihood score is calculated as

$$L(i) = \sum_{t=1}^{S_i} \log[p(\underline{x}_t \mid \lambda)] - \log[p(\underline{x}_t \mid \lambda_g)]. \tag{5}$$

Moreover, a genuine threshold value G_i and an impostor threshold value P_i are defined in each stage. Generally, $G_i > P_i$ for $i < N$ and $G_N = P_N = T$.

The steps of the algorithm can be summarized as follows:

1) Initialize $i = 1$ and calculate $L(i)$ as described in equation (5).
2) If $L(i) > G_i$, the system decides that the utterance is from the genuine speaker (accept) and verification is deemed complete.
3) If $L(i) \leq P_i$, the system decides that the utterance is from an impostor (reject) and verification is deemed complete.
4) If $P_i < L(i) \leq G_{i,}$, increment i by 1, update the normalized score by

$$L(i) = L(i-1) + \sum_{t=S_{i-1}+1}^{S_i} \log[p(\underline{x}_t \mid \lambda)] - \log[p(\underline{x}_t \mid \lambda_g)]. \tag{6}$$

5) Repeat steps 2 to 4 until $i = N$ or the loop exits when the condition in step 2 or 3 holds and $i < N$.

For clarity, a graphical illustration showing the likelihood scores (i.e. $\sum_{t=1}^{i} \log[p(\underline{x}_t \mid \lambda)] - \log[p(\underline{x}_t \mid \lambda_g)]$ for $i=1,2,3,...,F$) against the number frames considered (i)

is shown in Fig. 1. The maximum number of stages (N) is set to be 3 in this figure as well as the experiments described in Section 4. N needs to be constrained to a small number because a large N presents excessive memory requirements for storing the predefined threshold values G_i and P_i for various i. An alternative implementation is to compute the values G_i and P_i at run-time. However, this presents additional CPU requirements and conflicts with our primary objective. In this work, the threshold values for G_i and P_i are obtained experimentally. G_i values are chosen as the maximum likelihood scores obtained among all verification tests of impostors, whereas P_i values are chosen as the minimum likelihood scores obtained among all verification tests of genuine speakers. This selection method attempts to lower false acceptance and false rejection in the algorithm.

The conventional approach without the pruning makes a verification decision only at S_3, by considering all frames extracted from the utterance. In the pruning approach, once the likelihood score enters region 1 at either S_1 or S_2, the system gives an "accept" decision. On the other hand, once the likelihood score enters region 2 at either S_1 or S_2, the system generates a "reject" decision. The underlying idea is that if the likelihood score either retains a high value (i.e. enters region 1) or falls rapidly below a certain value (i.e. enters region 2) at an early stage, the likelihood score will probably stay in the corresponding region even when all the feature vectors are considered. To reduce the computational time, the comparisons of frames after this certain stage with the GMM models are eliminated.

There is a trade-off between maximizing verification performance and minimizing execution time related to the choices of the threshold values. When the values for G_i increase or the values for P_i decrease, fewer utterances can make "early" verification decisions, but this can reduce additional false acceptance and false rejection decisions. Conversely, when the values for G_i decrease or the values for P_i increase, more utterances can make "early" verification decisions, but this may increase additional false acceptance and false rejection decisions.

The difference between our pruning approach for speaker verification and the one proposed in [3] for speaker identification is that the temporal sequence of feature vectors is kept in our pruning approach for speaker verification. In speaker identification [3], the F feature vectors $\{\underline{x}_1, \underline{x}_2, \underline{x}_3 \dots \underline{x}_F\}$ extracted are compared with all registered speakers' models in the system. The identification process then finds the speaker model λ_I which maximizes the likelihood score among all registered speakers. This is described in equation (7) that considers the case in K registered speakers:

$$\lambda_I = \arg\max_{1 \le k \le K} \sum_{t=1}^{F} \log[\, p(\underline{x}_t \mid \lambda_k)]. \tag{7}$$

Normalization with the likelihood score from the world model $\log[p(\underline{x}_t \mid \lambda_g)]$ is unnecessary in speaker identification. The term $\log[p(\underline{x}_t \mid \lambda)]$ described in equation (7) exhibits a high degree of correlation with neighboring feature vectors. To increase the efficiency of the pruning approach in [3], the likelihood score is not computed according to the temporal order of the feature vectors. Instead, feature vectors with the maximum interval distance are sampled first. This enables the pattern matching

process to rapidly consider all scores attributed from the acoustic space of the utterance.

However, reordering feature vectors has no advantage in speaker verification since the term $\log[p(\underline{x}_t \mid \lambda)] - \log[p(\underline{x}_t \mid \lambda_g)]$ described in equation (3) does not exhibit any correlation with neighboring feature vectors. There is no difference between sampling neighboring feature vectors and sampling feature vectors with the maximum interval distance.

4   Experiments

Experiments were carried out to evaluate the performance of the pruning approach in term of computation time and verification accuracy. The speaker verification system used in the experiments was developed using the Hidden Markov Toolkit (HTK) [7], which is an open source speech- and HMM-related software package primarily for speech recognition. Optimization techniques presented in [2] have also been incorporated. A PDA, iPAQ H3600 [8] was used as the mobile embedded device in this project. The PDA was equipped with a 32-bit StrongARM embedded processor, 16MB RAM and 16MB flash ROM. Embedded Linux was installed in the iPAQ. Executables were compiled for the mobile embedded system with a C-language cross compiler, Arm-Linux-GCC [9].

The YOHO corpus [10], a collection of utterances from 138 speakers over 3 month period in a real-world office environment, was used in the experiments. The corpus was separated into 3 subsets encompassing 20, 59 and 59 speakers respectively. The first subset was used to train a world model for likelihood normalization. The second subset was used to learn the threshold values G_i and P_i for various i. The third subset was used to evaluate the computation time and the verification performance based on the threshold values learnt.

In the experiments, the silence in the leading 0.8 second and in the trailing part of each utterance was removed. Each resulting utterance with the duration of around two seconds was segmented into a number of frames by a 10ms Hamming window [11]. Each speaker model consists of a 32 Gaussian mixtures trained by the 19-dimensional MFCC feature vectors computed every 10ms. Each speaker in the corpus has 4 training utterances for creating the speaker model.

For the verification test, each speaker has 40 utterances to verify against himself/herself and other speakers in the same subset act as impostors. Verification testing was done to compare the performance of the pruning approach and the conventional approach.

In the conventional approach, all 139,240 verification trials made the decision when all frames were considered. In the pruning approach, the verification trials finished in Stages 1, 2 and 3 as summarized in Table 1. S_1 and S_2 are set to be $0.5F$ and $0.8F$ respectively in the experiments. 39,328 verification trials (around 28% of all trials) made the decision at Stage 1 (only 50% of frames are considered) and another 38,162 verification trials (around another 27% of all trials) made the decision at Stage 2 (80% of frames are considered). Therefore, around 28% and 27% of the verification trials reduced their execution time in the pattern matching process by 50% and 20% respectively.

Table 1. Performance of the pruning approach

Stage number (i)	Percentages of frames considered in the verification decision	No. of utterances making the verification decision in this stage	No. of false acceptance (FA) decisions incurred by the pruning algorithm	No. of false rejection (FR) decisions incurred by the pruning algorithm
1	50%	39,328	1	1
2	80%	38,162	0	6
3	100%	61,750	---	---

Fig. 2. ROC curves

The pruning approach incurs additional false acceptances (FAs) and false rejections (FRs), as shown in Table 1. To compare the verification performance of the pruning approach and the conventional approach, two corresponding Receiver Operating Characteristic (ROC) curves are plotted, as illustrated in Figure 2. The curves show that both approaches attain the equal error rate (EER) at around 6.7%.

To further study the relationship between the performance speedup and the verification performance in the pruning approach, 3 more setups with different choices of the threshold values were compared as shown in Table 2. Setup 1 used the original threshold values learnt. Setup 2 narrowed down the "early" decision region (the combined area of regions 1 and 2 in Fig. 1.). Setup 3 and 4 widened the "early" decision region. In Setup 2, the percentage of verification trials with "earlier" decisions decreased to only 39% but the EER was the same as the EER in Setup 1. In Setup 3, the percentage of verification trials with "early" decisions increased to 74% while the EER was still kept at 6.7%. In Setup 4, the percentage of verification trials with "early" decisions increased to 88% while the EER was increased to 7.2%.

Table **2.** Performance speedup and verification performance in the pruning approach

	New threshold values	Stage number (i)	No. of utterances making the verification decision in this stage (percentage of the total no. of verification trials)	Percentage of all verification trials with "early" decisions	No. of FA(s) incurred by the pruning algorithm *in this stage*	No. of FR(s) incurred by the pruning algorithm *in this stage*	EER (%)
Setup 1	$G_i' = G_i$ $P_i' = P_i$	1	39,328 (28%)	55%	1	1	6.7
		2	38,162 (27%)		0	6	
Setup 2	$G_i' = G_i + 0.2\,|G_i|$ $P_i' = P_i - 0.2\,|P_i|$	1	22,594 (16%)	39%	0	0	6.7
		2	31,675 (23%)		0	0	
Setup 3	$G_i' = G_i - 0.2\,|G_i|$ $P_i' = P_i + 0.2\,|P_i|$	1	63,505 (46%)	74%	1	11	6.7
		2	38,494 (28%)		0	16	
Setup 4	$G_i' = G_i - 0.4\,|G_i|$ $P_i' = P_i + 0.4\,|P_i|$	1	92,713 (67%)	88%	2	51	7.2
		2	29,506 (21%)		0	80	

5   Conclusions

This paper presents a preliminary study of a pruning approach in GMM-based pattern matching used for speaker verification. With appropriate threshold values learnt from a set of training data, 28% of the verification trials reduced their execution time in the pattern matching process by 50%, another 27% of the verification trials reduced their execution time by 20%, and the remaining 45% of the verification trials kept unchanged in their execution time. Overall, the pruning approach reduces the execution time of pattern matching on the entire testing set by 20% while maintaining the verification performance. The tradeoff between the performance speedup and the verification accuracy was studied in the pruning approach. With the wider "early" decision region (Setup 4 in the experiments), the percentage of verification trials with "earlier" decisions increased from 55% to 88%, but the EER was increased slightly from 6.7% to 7.2%. Possible ways to choose the threshold values will be studied in the future. Future work will also include studying whether vowel/consonant discrimination can enhance the performance of the verification process.

References

[1]   C. C. Leung, Y. S. Moon, "Effect of Window Size and Shift Period in Mel-warped Cepstral Feature Extraction on GMM-based Speaker Verification," *Proceedings of the 4th International Conference on Audio- and Video-based Biometric Person Authentication (AVBPA)*, pp. 438-445, 2003.

[2]   Y. S. Moon, C. C. Leung, K. H. Pun, "Fixed-point GMM-based Speaker Verification over Mobile Embedded System," *Proceedings of ACM Workshop on Biometrics: Method and Applications*, pp. 53-57, 2003.

[3]   B.L. Pellom and J. H. L. Hansen, "An efficient scoring algorithm for Gaussian mixture model based speaker identification," *IEEE Signal Processing Letters*, vol. 5, issue 11, pp. 281-284, 1998.

[4]   J. R. Deller, J. G. Proakis, J. H. L. Hansen, *Discrete-Time Processing of Speech Signals*, Macmillan, 1993.

[5]   D. A. Reynolds, "Speaker identification and verification using Guassian mixture speaker models," *Speech Communications*, vol. 17, pp. 91-108, 1995.

[6]   M. Carey, E. Parris, and J. Bridle, "A speaker verification system using alphanets," *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, pp. 397-400, 1991.

[7]   Hidden Markov Model Toolkit – speech recognition research toolkit, http://htk.eng.cam.ac.uk/.

[8]   Compaq iPAQ H3600 Hardware Documents, http://www.handhelds.org/Compaq/iPAQH3600/.

[9]   arm-linux-gcc Cross Compiler, ftp://ftp.handhelds.org/pub/linux/arm/toolchain/.

[10]   J. P. Campbell, Jr., "Testing with the YOHO CD-ROM voice verification corpus," *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 341-344, 1995.

[11]   L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*, Englewood Cliffs, N.J.: Prentice Hall, 1993.

Speaker Identification Using Higher Order Spectral Phase Features and their Effectiveness vis–a–vis Mel-Cepstral Features

Vinod Chandran, Daryl Ning, and Sridha Sridharan

Queensland University of Technology
Brisbane 4001, Australia
{v.chandran,d.ning,s.sridharan}@qut.edu.au

Abstract. The effectiveness of higher-order spectral (HOS) phase features in speaker recognition is investigated by comparison with Mel Cepstral features on the same speech data. HOS phase features retain phase information from the Fourier spectrum unlike Mel–frequency Cepstral coefficients (MFCC). Gaussian mixture models are constructed from Mel–Cepstral features and HOS features, respectively, for the same data from various speakers in the Switchboard telephone Speech Corpus. Feature clusters, model parameters and classification performance are analyzed. HOS phase features on their own provide a correct identification rate of about 97% on the chosen subset of the corpus. This is the same level of accuracy as provided by MFCCs. Cluster plots and model parameters are compared to show that HOS phase features can provide complementary information to better discriminate between speakers.

1   Introduction

Most speaker recognition (SR) systems make use of features derived from second order statistics, e.g., linear prediction and the power spectrum. One such feature that has shown good results in SR is the group of Mel–frequency cepstral coefficients (MFCCs) [1, 2, 3, 4]. These features, however, ignore spectral phase information. While most of the perceptual information about speech resides in the amplitude, phase information has also been shown to be important [5, 6]. At the very least, there is loss of the ability to discriminate between a time-reversed version of the input and itself if all Fourier phase information is discarded. Whether this loss results in a loss of intelligibility or a loss of discrimination between speakers has not been studied previously in the manner described here.

2   Higher Order Spectral Features

Unlike the power spectrum, higher order spectra (HOS) retain both the phase and amplitude information from the Fourier transform. The bispectrum, $B(f_1, f_2)$, of a one–dimensional, deterministic, discrete–time signal, $x(n)$, is defined by

$$B(f_1, f_2) = X(f_1)X(f_2)X^*(f_1 + f_2) \ , \tag{1}$$

where $X(f)$ is the discrete time Fourier transform of $x(n)$ at normalised frequency, f. Note, that this is a deterministic framework and there is no expectation operation on the right hand side. If the one–dimensional signal is divided into blocks, the triple products above can be averaged to yield the more conventional estimate of the bispectrum used in higher-order statistics which is the Fourier transform of the third-order correlation of the signal. References to seminal and review papers in the field of higher-order spectra can be found in the reference lists of [7] and [8].

The normalised magnitude of $E[B(f_1, f_2)]$ gives an indication of the relative degree of phase coupling between triads of Fourier components [7]. However, phase coupling is not the information extracted here because no averaging is performed. A more direct relationship between the shape of a deterministic signal and the phase of its deterministic bispectrum is exploited. This shape contains information about speech and speaker, as do MFCCs. A statistical model of features, such as a Gaussian Mixture, that is trained over many speech blocks from the speaker will tend to become speech independent and can be used for speaker identification (SI). For good discriminability between speakers, the feature set must be sensitive to small changes in the shape of the signal between speakers for the same speech. At the same time, the features must be invariant or robust to changes in amplitude (decibel-level) and time-shifts caused by changes in sampling or segmentation. If features are robust to such transformations, there is less intra-class variance and the probability density will be more dependent on changes that discriminate between speakers.

A set of features based on bispectral phases that is translation, scaling, amplification, and mean-shift invariant was derived by Chandran and Elgar [8] and is described briefly here. Assuming there is no bispectral aliasing, the bispectrum of a real signal is uniquely defined within the triangle $0 \leq f_2 \leq f_1 \leq f_1 + f_2 \leq 1$. Parameters are obtained by integrating along straight lines passing through the origin in bifrequency space. The region of computation and line of integration are depicted in Fig. 1. The bispectral invariant, $P(a)$, is the phase of the integrated bispectrum along the radial line with slope equal to a. This is defined by

$$P(a) = \arctan \left(\frac{I_i(a)}{I_r(a)} \right) \ , \tag{2}$$

where

$$I(a) = I_r(a) + jI_i(a) = \int_{f_1=0^+}^{1/(1+a)} B(f_1, af_1)df_1 \ , \tag{3}$$
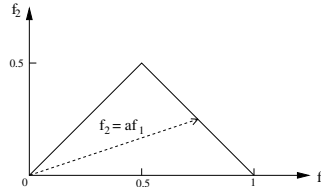
Fig. 1. Region of computation of the bispectrum for real signals. Features are calculated by integrating the bispectrum along the dashed line with slope=a

for $0 < a \leq 1$, and $j = \sqrt{-1}$. The variables I_r and I_i refer to the real and imaginary parts of the integrated bispectrum respectively.

3   Speaker Identification Using HOS

The HOS phase parameters, $P(a)$, are used as features in a SI system using Gaussian mixture speaker models. The feature extraction, speaker modelling, and performance evaluation of this system are described briefly in this section.

In our experiments, only the voiced speech frames are utilised for HOS feature extraction. Voiced segments contain the appropriate harmonic structure that give rise to significant bispectral values [9, 10]. Each frame of speech consists of 256 samples with a frame advance (hop) of 80 samples. The bispectrum is calculated from each frame and the integrated phase parameters, $P(a_i)$, are determined, where $a_i = i/D$, $i = 1 \ldots D$, and $D = 16$. These phase parameters are not unwrapped such that $-\pi < P(a_i) \leq \pi$.

The distribution of integrated phase parameters from each speaker is modelled using a GMM. A GMM is simply a weighted sum of M Gaussian densities, and in this work, the densities are multivariate. An explanation of GMM's and procedures for estimation of their mixture weights and densities is given in [1]. In this work, diagonal covariance matrices are used, and $M = 32$.

The task in SI is to correctly identify a speaker (from a group of known speakers) given some of his/her speech. We attempt this by finding which, out of a group of S speaker models, is most likely to produce the observation sequence, $X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T\}$. X is simply a sequence of T feature vectors extracted from the given speech. Assuming equally likely speakers, independence between observations, and noting that $p(X)$ is the same for all speaker models, we classify the speaker based on the following:

$$\hat{S} = \arg \max_{1 \leq k \leq S} \sum_{t=1}^{T} \log p(\boldsymbol{x}_t | \lambda_k) \ , \tag{4}$$

where λ_k is the model for the kth speaker. To evaluate the SI system, the test speech is divided into overlapping segments of observation sequences. Each sequence consists of $T = 200$ feature vectors, corresponding to 2 second utterances.

An observation advance of 10 ms is used, hence each sequence differs from the previous sequence by only one feature vector. For each segment, (4) is used to determine the speaker which gives the maximum probability for the observation sequence. This is repeated for all the possible segments in the length of the test speech and across all the speakers in the database. The correct identification rate is then calculated as

$$\% \text{ Correct Identification} = \frac{\text{Total \# of correctly identified segments}}{\text{Total \# of segments}} \times 100 \ .$$

(5)

4 Experiments and Discussion

4.1 Telephone Speech

The first set of experiments used speech data obtained from the Switchboard–2 Phase I telephone speech corpus. Since our intention is to investigate variability of the features between speakers and not owing to channel, handset or recording condition variations, we selected speech files keeping the channel quality ratings as similar as possible. A total of 19 speakers (10 male and 9 female) were selected from this database, and from each speaker, one conversation was selected. This conversation was then split into two non-overlapping parts. One part was used for training, and the other was used for testing.

Using the 16 integrated phase parameters as features, a correct identification rate of 97.46% was achieved. This simple SI experiment shows that the integrated bispectral phase parameters hold information capable of discerning known speakers within a database. A set of 12 MFCC parameters were used in the comparison experiment, (instead of the 16 integrated phase parameters), and all other aspects of the system remained the same. The MFCC based system achieved a correct identification rate of 97.95%. From this result we can see that the HOS phase parameters can produce comparable results to the more widely used MFCC parameters.

Since the MFCC parameters can already deliver similar identification rates, there is no real need for using HOS phase parameters as an alternative feature set. It may be useful, however, to use both sets of parameters in a fused classification system system for improved performance. The main diagonal of the confusion matrix for each of the two tests is given in Table 1. This indicates the percentage of correct identifications for each individual speaker in the database. This table suggests that the two feature sets may be used to complement each other. For speakers 3 and 8, the MFCC features obtain correct identification rates of 90% and 80.7% respectively, whereas the HOS phase features obtain higher rates of 100% and 92.1% respectively. Likewise, the MFCC features outperform the HOS phase features for speakers 7, 10 and 12.

In order to understand the differences between the feature sets that give rise to the small differences in recognition accuracy, we used scatter plots to model parameter distributions in two arbitrary dimensions. This is a reduced representation but it allows us to visualize the feature spaces and compare models. We

Table 1. Percent correct ID for tests using MFCC and HOS phase parameters

Speaker	MFCC Feature Set	HOS Phase Feature Set	Speaker	MFCC Feature Set	HOS Phase Feature Set
1	100	100	11	100	100
2	100	100	12	100	78.1
3	90.0	100	13	100	100
4	100	100	14	100	100
5	100	97.6	15	99.6	99.8
6	100	100	16	100	100
7	100	91.8	17	100	100
8	80.7	92.1	18	100	100
9	100	100	19	100	100
10	100	90.5			

selected speakers that were significantly misclassified as each other as observed from the confusion matrix. Speakers 3 and 8 caused problems for the MFCC features, with misclassifications of 10% and 19.7% from 3 to 8 and vice-versa. Speakers 12 and 16 caused problems for the HOS phase feature set with misclassification of 6.8% from speaker 12 to speaker 16. Feature distributions for both feature sets were plotted for these sets of speakers.

Figure 2(a) shows scatter plots of the two feature sets for speakers 3 and 8. The HOS phase features are on the left and MFCC features on the right. The centres of the Gaussian mixtures and elliptical contours corresponding to unit standard deviations along the major and minor axes are overlaid on the cluster plots to give some indication of the models in each case. The principal axes of these ellipses are along the x and y dimensions because the covariance matrices of the models are diagonal by choice. The weights (or priors) of each mode cannot be inferred from these plots. Differences in feature distributions between speakers are evident in both sets of plots, and there is also a marked difference between the clusters for the two feature sets.

However, it is still difficult to understand why one feature set may perform better than the other or why there would be more misclassifications between the two particular speakers from one of the feature sets. We chose to compare the feature distributions indirectly using the distribution of the centres (or mean values) of the 32 modes in the GMM. The distribution of these mode centres is approximated by a unimodal Gaussian and a Bhattacharya distance [11] given by

$$\frac{1}{8}(M_2 - M_1)^T \left[\frac{\Sigma_1 + \Sigma_2}{2}\right]^{-1} (M_2 - M_1) + \frac{1}{2}\ln\frac{|\frac{\Sigma_1 + \Sigma_2}{2}|}{\sqrt{|\Sigma_1||\Sigma_2|}} \quad , \tag{6}$$

where M_i is the mean of the set of mean vectors for speaker i, and Σ_i is the covariance of the set of mean vectors for speaker i, is computed. This distance gives a quantitative measure of the difference between two probability distributions.

Figure 3(a) shows the centres and the unit standard deviation ellipses of the mode centre distributions (in the reduced 2 dimensional case) for speakers 3 and 8 – for HOS phase features on the left and MFCC features on the right. The Bhattacharya distances for the HOS phase set and the MFCC set for speakers 3

(a)

(b)
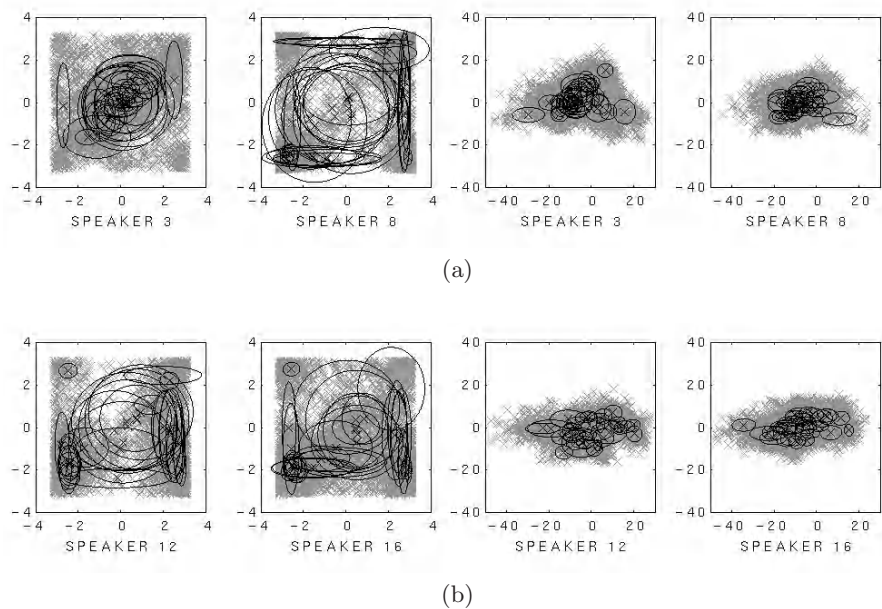
Fig. 2. Scatter plot of 2 dimensions for (a) HOS phase parameters and their corresponding GMMs for speakers 3 and 8 (left) and MFCC parameters and their corresponding GMMs for speakers 3 and 8 (right); and (b) HOS phase parameters and their corresponding GMMs for speakers 12 and 16 (left) and MFCC parameters and their corresponding GMMs for speakers 12 and 16 (right)

and 8 are given in Table 2. The greater distances, indicating better separation of speaker models, corresponds to the better performance of HOS features for these two speakers. Note that distances are given when calculating (6) using the reduced two–dimensional case (corresponding to Figure 3(a)), as well as using the full 16 dimensional feature space.

Scatter plots for features from speakers 12 and 16 are given in Fig. 2(b), and the GMM modal centre distributions represented by unimodal Gaussians are depicted in Fig. 3(b). Figure 3(b) shows that the overlap between the centre distributions is less for the MFCC than for HOS parameters in the reduced two–dimensional case. The Bhattacharya distances between the mode centre distributions for the models of these two speakers are given in Table 2.

Table 2. Bhattacharya distances between mode centre distributions of the GMMs

Dimensions	Speakers 3 and 8		Speakers 12 and 16	
	MFCC Feature Set	HOS Phase Feature Set	MFCC Feature Set	HOS Phase Feature Set
All	1.46	7.62	1.73	3.12
2	0.04	0.4	0.12	0.01
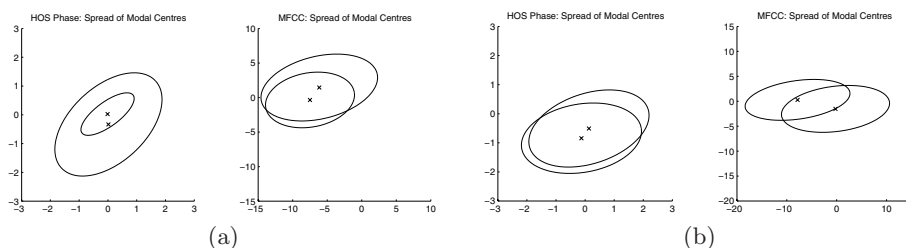
Fig. 3. A representation of the distribution of centres of the modes in the GMM models for (a) Speakers 3 and 8, and (b) Speakers 12 and 16. The × marks are the mean values and the ellipses correspond to unit standard deviations along the principal axes. The plot on the left is for HOS phase features, on the right for MFCC features

The distance measures for the two–dimensional case, corresponds to Fig. 3(b) and with the fact that MFCC features performed better. However, for the full feature set in each case, HOS features show a greater distance, and this was surprising. Although these distances are only a rough indication, they seem to suggest that the HOS parameter set may contain more information than the MFCC set for discrimination between the speakers. This is not surprising, if we recall that they do contain phase as well as magnitude spectral information. This does not necessarily translate to identification accuracy, however, because the HOS phase distributions may be more difficult to capture in a statistical model such as the GMM.

4.2   Microphone Speech

It is possible that the original system was classifying speakers based on individual channel differences, rather than speaker differences. In order to remove the influence of the channel on SI performance, another experiment was performed using speech obtained from a tabletop microphone (as opposed to through a telephone channel). The speech for this experiment was obtained from the multi-modal task evaluation data used in the 2002 National Institute of Standards and Technology (NIST) SR evaluation. The speech from this database is recorded at 16 kHz, so we first filter and down-sample each of the speech files to 8 kHz before processing. We follow the same procedure for training and testing as in the first experiment. A total of 20 male speakers was used in this experiment. The HOS phase features give a correct identification rate of 98.5%. Since the same microphone is used for each of the speakers in the database, we can conclude that the HOS phase features are successful in identifying the speakers based on their speech as opposed to channel variations. The MFCC features achieved a correct identification rate of 99.4% for the same database. When the test segment is increased from 2 seconds to 4 seconds, the correct identification rate improves to 100% for both feature sets.

4.3   Computational Cost

Lastly, the computational price paid for incorporating phase information is not exorbitant. A 29 second segment of speech can be processed for MFCC feature extraction in about 1 second and for HOS feature extraction in roughly 12 seconds, using the procedures described in this work. These results are for Matlab programs without any optimization and averaged over 100 runs on the same platform. Roughly, the price for computing bispectral phase features is a factor of 12 when processing 32 millisecond blocks. It does not limit practical implementation in real-time systems.

5   Conclusions

HOS phase parameters contain information that can be used to recognize speakers. They are shown to perform at the same level as MFCCs on the same data under identical conditions. They contain information that can complement MFCCs in a fused classifier. The widely accepted notion that Fourier phase information is unimportant in speech processing is questioned by experimental evidence.

Acknowledgments

References

[1] Reynolds, D. A., Rose, R. C.: Robust Text-Independent Speaker Identification using Gaussian Mixture Speaker Models. IEEE Transactions on Speech and Audio Processing **3:1** (1995) 72–83   614, 616

[2] Reynolds, D. A.: Large Population Speaker Identification using Clean and Telephone Speech. IEEE Signal Processing Letters **2:3** (1995) 46–48   614

[3] Liu, L., He, J., Palm, G.: Signal Modeling for Speaker Identification. International Conference on Acoustics, Speech, and Signal Processing **2** (1996) 665–668   614

[4] Cordella, L. P., Foggia, P., Sansone, C., Vento, M.: A Real–time Text Independent Speaker Identification System. Proceedings of the 12th International Conference on Image Analysis and Processing (2003) 632–637   614

[5] Patterson, R. D.: A Pulse Ribbon Model of Monaural Phase Perception. Journal of the Acoustical Society of America **82:5** (1987) 1560–1586   614

[6] Pobloth, H., Kleijn, W. B.: On Phase Perception in Speech. International Conference on Acoustics, Speech, and Signal Processing **1** (1999) 29–32   614

[7] Elgar, S., Chandran, V.: Higher Order Spectral Analysis to Detect Nonlinear Interactions in Measured Time Series and an Application to Chua's Circuit. International Journal of Bifurcation and Chaos **3:1** (1993) 19–34   615

[8] Chandran, V., Elgar, S. L.: Pattern Recognition using Invariants Defined from Higher Order Spectra–One Dimensional Inputs. IEEE Transactions on Signal Processing **41:1** (1993) 205–212   615

[9] Wells, B. B.: Voiced/Unvoiced Decision based on the Bispectrum. International Conference on Acoustics, Speech and Signal Processing **10** (1985) 1589–1592   616

[10] Fackrell, J. W. A., McLaughlin, S.: The Higher–Order Statistics of Speech Signals. IEE Colloquium on Techniques for Speech Processing and their Applications (1994) 7/1–7/6   616

[11] Fukunaga, K.: Introduction to Statistical Pattern Recognition. Academic Press, Boston (1990)   618

High Performance Speaker Verification System Based on Multilayer Perceptrons and Real-Time Enrollment

Tae-Seung Lee[1] and Ho-Jin Choi[2]

[1] School of Electronics
Telecommunication and Computer Engineering, Hankuk Aviation University
200-1, Hwajeon-dong, Deokyang-gu, Koyang-city, Kyonggi-do, 412-791, Korea
`thestaff@hitel.net`
[2] Software Technology Institute, Information and Communications University
517-10, Dogok-dong, Kangnam-gu, Seoul, 135-854, Korea
`hjchoi@icu.ac.kr`

Abstract. Speaker verification systems based on multilayer perceptrons (MLPs) have good prospects in reliability and flexibility as required for a successful authentication system. However, poor learning speed of error backpropagation (EBP), the representative method of learning for MLPs, has been the major problem which must be resolved to achieve real-time user enrollment. In this paper, we implement an MLP-based speaker verification system by applying methods of omitting patterns in instant learning (OIL) and discriminative cohort speakers (DCS) to approach the real-time enrollment. We evaluate the system on a Korean speech database and demonstrate the feasibility of it as a speaker verification system of high performance.

1 Introduction

An influential speaker verification system requires to have reliability of achieving high verification rate and flexibility of easily accessing the system. Reliability is the most important property for any authentication system. A reliable speaker verification system gives verification scores as high as possible in any working conditions. A reliable verification system, however, may not be acceptable if accessibility to the system is not so flexible that users feel some difficulty. A flexible speaker verification system presents fast accessibility in verifying and enrolling users.

Among various pattern recognition engines, parametric or nonparametric, multiplayer perceptrons (MLPs) seem to satisfy the above two properties (viz. reliability and flexibility) most efficiently. MLPs are an assembly of simple computational nodes which are commonly trained by an error backpropagation (EBP) algorithm for classifying learning models. MLPs have superior recognition performance and faster operation speed than other methods, in that they utilize inter-model information and that learning models of an MLP share working capabilities of the neural network [1]. For speaker verification applications, MLPs show low error rate and fast speed of verification.

However, MLPs have a serious drawback of slow speed in learning. A standard EBP algorithm for training MLPs is notoriously slow due to its dependency on local gradient [1]. For pattern recognition in general, as many learning patterns are required as possible in order to achieve high recognition rate. For speaker verification in particular, abundant background speakers should be reserved for verifying a claimant with strict criterion [2]. This requirement of plenty background speakers inevitably causes longer learning time of EBP, long waiting for completion of enrollment, and ultimately poor flexibility of MLP-based speaker verification systems.

To complement the slowness of EBP algorithm and relieve the burden of plentiful background speakers, Lee et al. suggested two different methods, called omitting patterns in instant learning (OIL) [3] and discriminative cohort speakers (DCS) [4], [5]. The OIL method exploits redundancy of pattern recognition data, and achieves a substantial improvement in learning speed without losing any recognition rate. The DCS method selects the very background speakers related to the enrolling speaker for utilizing discriminant learning property of MLPs, and obtains a rather effective result in enrolling speed. Since OIL works inside the EBP algorithm and DCS works on the outside learning data set, they can be considered as local and global, respectively, optimization of speaker enrollment speed in MLP-based speaker verification systems.

In this paper, we present an implementation of MLP-based speaker verification system and evaluate the performance. The implemented system features low error rate of verification and fast verifying speed although it has some tedious process of enrollment. By combining the two methods of enrollment speed-up, viz. OIL and DCS, we obtain better flexibility for the implemented system as well as superior performance.

2   Implementation of MLP-Based Speaker Verification System

The implemented speaker verification system isolates words from input utterance, classifies the isolated words into nine streams of Korean continuants (/a/, /e/, /ə/, /o/, /u/, /ɨ/, /i/, /l/, nasals), and learns an enrolling speaker for each continuant using MLPs. The system then calculates identity scores for customer speakers. Because the system is based on continuants, which consist of the little phoneme set, it can adapt itself easily to any of text-dependent, text-independent or text-prompt modes. In our system, text-dependent mode is adopted for easy implementation, that is, enrolling text and verifying text are the same.

The steps to process the speech of enrolling and verifying speakers are (1) to analyze and extract features from given speech, and detect isolated words and continuants on the features, (2) to train MLPs with an enrolling speaker, and (3) to evaluate identity scores of claimants and determine acceptance or rejection.

Input utterance, sampled in 16-bit and 16-kHz, is divided into 30ms-frames each overlapped by 10 ms. 16 Mel-scaled filter bank coefficients are extracted from each frame and used to detect isolated words and continuants. To remove the effect of utterance loudness from entire spectrum envelope, an average of the coefficients from 0-Hz to 1-kHz is subtracted from every coefficient and the resulting coefficients are adjusted such that the average of all the coefficients becomes zero. 50 Mel-scaled filter bank coefficients that are linear scaled from 0-Hz to 3-kHz are extracted from

each frame and used for speaker verification. We have adopted this scaling from the study in [6] that information about speakers is more concentrated on the second formant rather than the first. To remove the effect of utterance loudness, the same process is applied as used in detecting isolated words and continuants.

Since the system uses continuants as units for speech recognition, the underlying densities exhibit mono-modal distribution [7]. Thus, it is good enough for each MLP to have a two-layered structure that includes one hidden layer [8], [9]. Since the MLPs need to learn only two models, i.e., one for the enrolling speaker and the other for the background speakers, they can learn the models using one output node and two hidden nodes. In total, nine MLPs are provided for the nine continuants.

3   Fast Methods for Speaker Enrollment

MLPs learn the representation of models by establishing a decision boundary to discriminate geometrically the model areas. If patterns of all models are fully presented in an iterative manner and the internal learnable weights of an MLP are adjusted so that all patterns of each model are classified into their own model area, the decision boundary can be finally settled in an optimal position.

The online mode EBP algorithm updates the weights of an MLP using the information related to a given pattern and the current status of weight vector [1]. The usefulness of the given pattern in current epoch can be determined on the criterion of error energy objective. In the online mode EBP, achievement of learning in current epoch is measured using the error energy averaged for all N patterns like this:

$$e_{avg}(t) = \frac{1}{N}\sum_{p=1}^{N}e_p(t) = \frac{1}{2N}\sum_{p=1}^{N}\sum_{k=1}^{M}e_k^2(t) \cdot \tag{1}$$

where e_p represents the summed error energy from all M output nodes for given pattern, e_k the error between network value of output node k and learning objective, and t the epoch count. Learning continues until the average error energy $e_{avg}(t)$ is less than the learning objective e_{obj}. The relationship between average error energy and error energies of individual patterns can be described as follows:

$$e_{avg}(t) \le e_{obj}, \quad if \ e_C^2(n) \le 2\lambda e_{obj} \ for \ all \ N \ patterns, \quad 0 < \lambda \le 1 \cdot \tag{2}$$

where $e_C^2(n)$ represents the error energy of the output node C associated with given pattern and n the update count of weight vector. This expression means that if $e_C^2(n)$s for all learning patterns are less than or equal to $2e_{obj}$, then the learning is complete, assuming that learning has been progressed sufficiently to be able to ignore the other output values beside C. As a result, it is possible to learn only the patterns of $e_C^2(n) > 2e_{obj}$ to complete learning. In Eqn. 2 the coefficient λ is inserted to determine the depth of patterns whose weight vectors are to be updated. When λ is near 1, the number of omitted patterns increases, but the count of learning epochs increases as well. Hence it is necessary to search for a proper λ to achieve the

minimum count of learning epochs and the maximum number of omitted patterns so that the shortest duration of learning is obtained. Omitting weight updates of useless patterns on the criterion in Eqn. 2 is the OIL method.

The prospect to reduce background speakers in MLP-based speaker verifications arises from the geometric contiguity of learning models. That is, in MLP learning, learning of a model is cooperated only with the models geometrically contiguous to the model. When an enrolling speaker is given in the crowd of background speakers, the decision boundary of an MLP to learn the difference between the enrolling speaker and background speakers is affected only by the background speakers adjacent to the enrolling speaker. If a large number of background speakers are reserved in the system to obtain very low verification error, the percentage of such adjacent background speakers does increase so that the actual number of background speakers needed to establish final decision boundary can be decreased.

The process of the DCS method to select the background speakers similar to an enrolling speaker is implemented like this:

$$S_{Cohort} = Sel_{M_{MLP} \geq \theta, I} (Sort_{Dec} (M_{MLP} (S_{BG} \mid \mathbf{X}))), \quad S_{BG} = \{S_i \mid 1 \leq i \leq I\}. \quad (3)$$

where $\mathbf{X}$ represents the speech of enrolling speaker, S_{BG} the background speakers set which population is I, M_{MLP} the MLP function which evaluates likelihoods of the given $\mathbf{X}$ to the background speakers. $Sort_{Dec}$ stands for the function to sort given values in descending manner, and $Sel_{M_{MLP} \geq \theta, I}$ for the function to select relevant background speakers whose M_{MLP}s exceed the preset threshold θ.

4   Performance Evaluation and Discussion

Performance of the implemented system is evaluated with respect to reliability and flexibility. Here, reliability of a speaker verification system is related to verification error rate and flexibility to working and enrolling speed. In this section we describe the measurements of the implemented system for these two aspects, and discuss the improvement achieved by applying to the system the two methods described in Section 3.

The speech data used in this evaluation are the recorded voice of connected four digits, spoken by 40 Korean male and female speakers. The digits are ten Arabic numerals pronounced in Korean as /goN/, /il/, /i/, /sam/, /sa/, /o/, /yug/, /cil/, /pal/, /gu/, each corresponding to a digit from 0 to 9. The average duration of each 4-digit string is about 1 to 1.5 second. Each speaker utters 35 words of different 4-digit strings four times, when the utterance is recorded in 16-bit resolution and 16-kHz sampling. Three of the four utterance samples are used to enroll the speaker, and the last utterance is used for verification. In order to learn the enrolling speakers discriminatively, additional 29 male and female speakers are participated as background speakers for MLPs other than the above 40 speakers.

Each of the 40 speakers can be treated as both the enrolling speaker and the test speaker. When one of them is picked as the test speaker, then the other 39 speakers are used as imposters. As a result, 35 tests using the 35 words are performed for a true

speaker and 1,365 (35 * 39) tests for the imposters. In total, we performed 1,400 (35 * 40) trials of test for true speaker and 54,600 (35 * 40 * 39) trials for imposters.

In our evaluation, we use equal error rate (EER) and the number of learning epochs, which stands for average number of epochs used to enroll a speaker for an isolated word. These values are calculated by taking the average of values obtained from three trials of learning, each trial being set to the same MLP conditions. The evaluation has been conducted on a 1-GHz personal computer.

Table 1 summarizes the best performance of the implemented system (using the online EBP algorithm) with learning rate of 0.5 and learning objective error energy of 0.005 being set. In the table, the enrolling frames and verifying frames designate the extracted average frames of continuants from enrolling three 4-digit strings and verifying one, respectively, and the durations mean processing times to enroll and verify a speaker. It shows that EER is good when we consider the length of enrolling (about 3 seconds) and verifying (about 1 second) utterances, compared with existing parametric speaker verifications [2], [10]. Especially, the verifying duration (measured in milli-second) seems excellent because most parametric speaker verifications tend to take long verifying durations because of computing matrices [10], [11]. However, the enrolling duration is slower than a single Gaussian model, though faster than the Gaussian mixture model [11]. The MLPs used in this paper correspond to the single Gaussian model since mono-modal distribution of speech generation probability is assumed.

To improve the performance for the enrolling duration, the OIL and DCS methods are applied to the implemented system. We first apply OIL only then apply DCS combined with OIL, and the performance results are evaluated and compared with those of the online EBP algorithm. The results are presented in Fig. 1. In the figure, OnEBP designates the online EBP algorithm and the numbers on the bottom the preset threshold θ s in DCS. The figures for OIL performance are measured with learning rate of 1, learning objective error energy of 0.005, and λ of 0.3. In the measurements of the DCS combined with OIL, the optimal result can be taken at $\theta = -0.999$ because the numbers beyond this point produce higher verification errors. To compare with the online EBP algorithm, OIL achieves a substantial improvement in enrolling duration without worsening verification error. With OIL applied, DCS keeps the learning duration decreased as the threshold increases. From the results, it can be inferred that the combination of the two methods is effective to shorten the enrolling duration over each individual method.

While maintaining the same level of verification error as with the online EBP algorithm, DCS marks the improvement of 14.6 % and OIL 55.6 % over the online EBP algorithm. The combination of the two methods further improves enrolling duration by 75.6% over the online EBP algorithm. These results demonstrate that the two methods are operating on different optimization principles which make a synergy when combined together. As a result, the enrolling duration of the implemented system has been reduced from 2.7 to 1.5 seconds.

Table 1. The best performance of the implemented system with the online EBP algorithm

| EER (%)	Number of Enrolling Frames	Number of Verifying Frames	Enrolling Duration (sec)	Verifying Duration
1.59	164.2	53.5	2.7	0.86
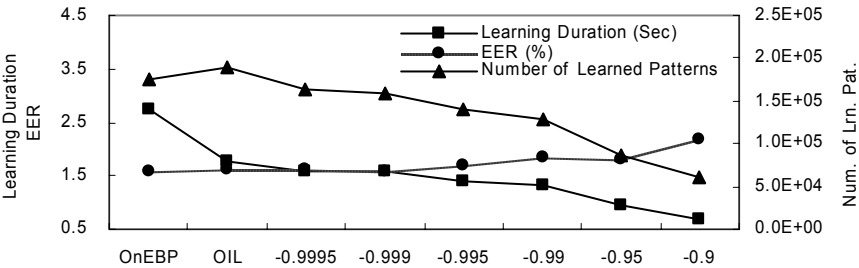
Fig. 1. Evaluation results of online EBP, OIL, and DCS with OIL

The reliability of the implemented system can be further improved by taking into account the tendency of verification errors with respect to the numbers of background speakers and continuants. In Fig. (a) of Fig. 2., the declining rate of EERs is nearly linear as the number of background speakers increases from 6 through 10, 16, 20, 26 to 29. Therefore, if more background speakers are given, EERs are expected to be lowered further. In figure (b) of Fig. 2, EERs are plotted with respect to the numbers of continuants and extracted frames of the continuants included in enrolling 4-digit strings. The figure shows that if more than seven continuants are included in all enrolling utterances (i.e. if the enrolling utterances contain continuants evenly), an EER lower than 1.59 % can be achieved. It is noted that the amount of enrolling frames hardly affects EERs when more than 220 enrolling frames are given. The EERs to the number of continuants are recorded in the first three columns of Table 2.

It is also worth evaluating EERs with respect to gender, i.e., when the genders of verifying speaker and enrolling speaker are the same or different. It has been reported that in parametric speaker verification, large part of verification error occurs when the genders are different [12]. It is inferred that the amount of training data is insufficient in such case, hence parameters are inaccurately estimated. In our system implemented using MLPs, the opposite phenomenon is observed in the same situation as seen in the latest two columns of Table 2. That is, superior EER is obtained for the case with different genders. This phenomenon can be understood from the characteristics of learning with MLPs to establish decision boundary to discriminate adjacent models.

Table 2. Verification errors to various conditions

	> 6 Continuants	> 7 Continuants	> 8 Continuants	Same Gender	Different Gender
EER (%)	1.15	0.89	0.81	1.84	1.01
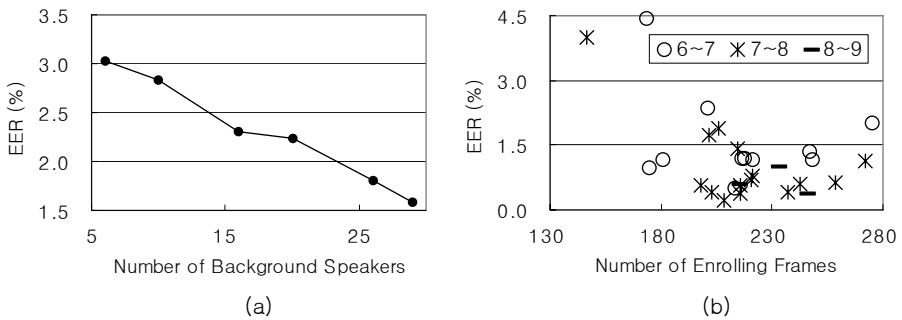
Fig. 2. Tendency of verification error to the numbers of (a) background speakers and (b) continuants

To conclude, MLPs have good properties for the application of speaker verification in reliability and flexibility as required for a successful authentication system. Although MLPs have the weak point of slow learning speed, the drawback can be improved by adopting the combination of the OIL and DCS methods suggested in this paper.

Acknowledgement

References

[1]   Bengio, Y.: Neural Networks for Speech and Sequence Recognition. International Thomson Computer Press, London Boston (1995)
[2]   Rosenberg, A. E., Parthasarathy, S.: Speaker Background Models for Connected Digit Password Speaker Verification. ICASSP **1** (1996) 81-84
[3]   Lee, T., Choi, H., Kwag, Y., Hwang, B.: A Method on Improvement of the Online Mode Error Backpropagation Algorithm for Pattern Recognition. LNAI **2417** (2002) 275-284
[4]   Lee, T., Choi, S., Choi, W., Park, H., Lim, S., Hwang, B.: Faster Speaker Enrollment for Speaker Verification Systems Based on MLPs by Using Discriminative Cohort Speakers Method. LNAI **2718** (2003) 734-743
[5]   Lee, T., Choi, S., Choi, W., Park, H., Lim, S., Hwang, B.: A Qualitative Discriminative Cohort Speakers Method to Reduce Learning Data for MLP-Based Speaker Verification Systems. LNCS **2690** (2003) 1082-1086
[6]   Cristea, P., Valsan, Z.: New Cepstrum Frequency Scale for Neural Network Speaker Verification. ICECS **3** (1999) 1573-1576
[7]   Savic, M., Sorensen, J.: Phoneme Based Speaker Verification. ICASSP **2** (1992) 165-168
[8]   Delacretaz, D. P., Hennebert, J.: Text-Prompted Speaker Verification Experiments with Phoneme Specific MLPs. ICASSP **2** (1998) 777-780
[9]   Lippmann, R. P.: An Introduction to Computing with Neural Nets. IEEE ASSP Magazine **4** (1987) 4-22

[10] Zhang, Y., Zhu, X., Zhang, D.: Speaker Verification by Removing Common Information. Electronics Letters **35** (1999) 2009-2011
[11] Zilca, R. D.: Text-independent Speaker Verification Using Utterance Level Scoring and Covariance Modeling. IEEE Trans. on Speech and Audio Processing **10** (2002) 363-370
[12] Parris, E. S., Carey, M. J.: Language Independent Gender Identification. ICASSP **2** (1996) 685-588

Spectral Subband Centroids as Complementary Features for Speaker Authentication

Norman Poh Hoon Thian, Conrad Sanderson, and Samy Bengio

IDIAP
Rue du Simplon 4, CH-1920 Martigny, Switzerland
{norman,bengio}@idiap.ch
conradsand@ieee.org

Abstract. Most conventional features used in speaker authentication are based on estimation of spectral envelopes in one way or another, e.g., Mel-scale Filterbank Cepstrum Coefficients (MFCCs), Linear-scale Filterbank Cepstrum Coefficients (LFCCs) and Relative Spectral Perceptual Linear Prediction (RASTA-PLP). In this study, Spectral Subband Centroids (SSCs) are examined. These features are the centroid frequency in each subband. They have properties similar to formant frequencies but are limited to a given subband. Empirical experiments carried out on the NIST2001 database using SSCs, MFCCs, LFCCs and their combinations by concatenation suggest that SSCs are somewhat more robust compared to conventional MFCC and LFCC features as well as being partially complementary.

1 Introduction

Speech recognition is the task of determining the linguistic contents of a speech signal, while speaker authentication is the task of verifying whether a person really is who he or she claims to be. Even though both tasks are very different, the front-end processing of speech signals is often common. Although there is some literature on designing new and effective speech features for speaker authentication [8] (i.e., Line Spectrum Pairs, Time-Frequency Principal Component and Discriminant Components of the Spectrum), Mel-scale Frequency Cepstral Coefficients (MFCCs), which are commonly used in speech recognition, remain the state-of-the-art features, as far as speaker authentication is concerned. Empirical studies in [12] showed that Linear-scale Frequency Cepstral Coefficients (LFCCs) [11] achieve comparable performance to that of MFCCs [12, 14]. According to the same study, Perceptual Linear Prediction (PLP) cepstral coefficients, which are widely used in speech recognition, did not perform significantly better than MFCCs. Furthermore, in the same experiment setting, the performance of PLP with RASTA-preprocessing (RASTA-PLP) [6] was slightly worse than PLP alone. Hence, features that work better in speech recognition *may not* always work better in speaker authentication.

The aim of this study is double-fold: to provide complementary features that describe information not captured by the conventional state-of-the-art MFCC

features for speaker authentication tasks; and to examine how these features perform alone, as compared to MFCC features. In [2, Sec. 3.3], frequency and amplitude information are extracted from "spectral lines" [5]. Spectral lines are extracted from the spectrogram of a signal by using thinning and skeletonisation algorithms that are often used in image-processing. Low frequency spectral lines in this case actually correspond to the fundamental frequency or pitch. The pair (frequency, amplitude) hence represents a point in 2D space. With quantisation on frequency and amplitude, this frequency/amplitude encoded data was classified using a feed-forward network and was shown to achieve a lower generalisation error as compared to the encoding scheme which uses fixed frequency intervals with their corresponding amplitude values. The study suggests that frequency information, when encoded properly, can increase the robustness of a *speech recognition* system.

Contrary to the first approach, in the context of *speaker authentication*, Sönmez *et al* directly estimated the (long-term) pitch information using parametric models called log-normal tied mixture models [16]. Follow-up work [15] used the (local variation of) pitch dynamics which contain speaker's intonation (speaking style). In both works, the resultant pitch system was combined with the cepstral feature-based system by summation of (log-)likelihood scores over the same utterance. They all showed improvement over the baseline system.

In the context of *speech recognition*, frequency information can be represented in the form of Spectral Subband Centroids (SSCs) [9], which represent the centroid frequency in each subband. In conventional MFCC features, the power spectrum in a given subband is often smoothed out, so that only the (weighted) amplitude of the power spectrum is kept. Therefore, SSCs provide different information to conventional MFCCs. It has been demonstrated [9] that SSCs, when used in conjunction with MFCCs, result in better speech recognition accuracy than that of the baseline MFCCs; when used alone, SSCs achieve performance that is comparable (but with slight degradation) to that of MFCCs.

Would frequency information enhance the performance of a speaker authentication system? According to [15, 16], the answer is yes. How should this information be incorporated into an existing system based on MFCC features? In this work, SSCs are used as a preliminary study since they can be incorporated at the frame-level (and of course at the classifier-score level) while this is not possible in [15, 16]. Furthermore, in these works, spectral information other than pitch (e.g. higher frequency band) is not used at all. Secondly, SSCs have not been applied to speaker authentication, constituting an interesting research question.

The rest of this paper is organised as follows: Section 2 briefly presents SSCs. Section 3 explains the experiment setting. This is followed by empirical results in Section 4 and conclusions in Section 5.

2   Spectral Subband Centroids

Let the frequency band $[0, F_s/2]$ be divided into M subbands, where F_s is the sampling frequency. For the m-th subband, let its lower and higher edges be l_m
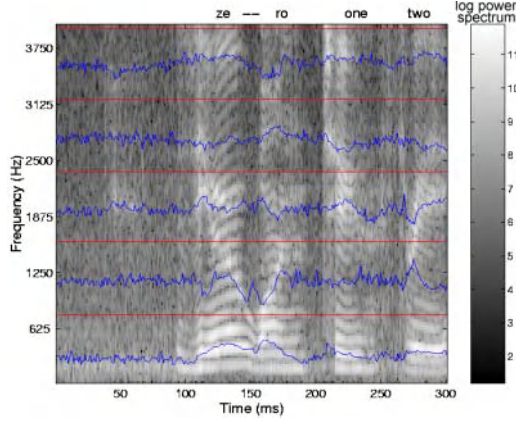
Fig. 1. SSC features across time

and h_m, respectively. Furthermore, let the filter shape be $w_m(f)$ and $P^\gamma(f)$ be the power spectrum at location f raised to the power of γ. The m-th subband centroid, according to [9], is defined as:

$$C_m = \frac{\int_{l_m}^{h_m} f w_m(f) P^\gamma(f) df}{\int_{l_m}^{h_m} w_m(f) P^\gamma(f) df} \tag{1}$$

Note that the term $w_m(f)P^\gamma(f)$ can be viewed as a bias which influences where the centroid should be. A peak in this term leads to a higher weight in the corresponding f.

Typically, $w_m(f)$ takes on the shape of either a square window (ones over the m-th subband and zeros everywhere else) or a triangular window. In the case of MFCCs, w_m is a triangular window. The same window is used here. The use of γ parameter in this function is a design parameter and is not motivated by any psychological aspect of hearing. The γ parameter has been used elsewhere in the literature [4] as part of feature extraction (which is called a two-dimensional root spectrum) for speech recognition. According to that study, γ is a design parameter which can be optimised on a given data set and task at hand.

Figure 1 shows a conventional spectrogram overlaid with the SSC features with five equally-spaced bands, calculated using square windows. The utterance contains three digits: "zero", "one" and "two". It can be observed that, firstly, when there is no speech, SSCs in a given frequency subband tend to be the center of the band. On the other hand, with the presence of speech, SSCs show some regular trends: the trajectory of SSCs in a given subband actually locates the peaks of the power spectrum in that subband. This coincides with the idea of spectral lines [5] discussed earlier. However, in this context, the representation is limited to one value per subband. Secondly, if there is not enough centroids, then SSCs will not adequately represent a given speech signal.

Prior to testing SSCs using a real-life noisy database, we carried out several preliminary studies on SSCs using Linear Discriminant Analysis (LDA)

under the Analysis of Variance (ANOVA) framework [10]. A subset of XM2VTS database and the female development set of NIST2001 (same as the one described in Section 3) were used for this test. The LDA analysis was used because it can separate useful sources of variance (e.g. physical articulatory features) from harmful sources of variance (e.g. handset differences, environmental and channel noise) [7]. We outline several conclusions of the preliminary studies reported in [10]:

- Based on LDA, we showed that about 12 to 16 centroids cover 99% of variance that is speaker discriminative. If less than 12 centroids are used, the speech utterance will be under-represented.
- Additional experiments based on LDA suggest that class labels (speaker's identities) not separable in SSC feature space are separable in MFCC (Mel-scale Frequency Cepsstrum Coefficient) feature space. This suggests that SSCs are potentially complementary to MFCCs.
- The Fisher-ratio test showed that the feature space induced by MFCCs is more separable than that induced by SSCs, thus predicting that the performance due to MFCCs under matched conditions is probably better than that due to SSCs.
- Preliminary empirical experiments on the female development subset of NIST2001 showed that about 16 to 18 centroids are optimal for speaker authentication.
- A theoretical study showed that mean-subtracted SSCs can somewhat reduce the effects of additive noise. The mean subtraction is done as follows:

$$C_m - E\{C_m\} \tag{2}$$

where $E\{C_m\}$ is the expectation of C_m over the whole utterance in a single access claim. The demonstration began with the assumption that a signal is composed of additive noise and the original clean signal. Deriving SSCs and mean-subtracted SSCs using this formulation, we showed that the additive component is partially cancelled during the mean subtraction. Empirical studies on NIST2001 also strongly supported this observation.
- Lastly, we showed empirically that first temporal derivates (deltas) of SSCs can also be used to further improve the performance.

The above studies were limited to studying the characteristics of SSCs compared to MFCCs under clean conditions. In this paper, the aspect of noise-robustness is evaluated.

3   Experiment Setup

In this study, a subset of NIST2001 was used to evaluate how well these features perform on telephone data with and without additive environmental noise, on speaker authentication tasks. It was obtained from the Switchboard-2 Phase 3 Corpus collected by the Linguistic Data Consortium. In this paper, only the female subset (which is known to be slightly more difficult than the male subset)

was used for evaluation. In the original database, data for two different handsets are present (i.e., carbon and electret). However, only data from electret handsets were used (5 speakers who used the carbon handsets were removed) so that any variation of performance, if any, will not be attributed to this factor. This database was separated into three subsets: a training set for the world model, a development set and an evaluation set. The female world model was trained on 218 speakers for a total of 3 hours of speech. For both development and evaluation (female) clients, there was about 2 minutes of telephone speech used to train the models and each test access was less than 1 minute long. The development population consisted of 45 females while there were 506 females in the evaluation set. There are 2694 accesses for the development population and 32029 accesses for the evaluation population, with a proportion of 10% of true claimant accesses. Four types of noise (`white`, `oproom` (for operational room), `factory` and `lynx`), taken from the NOISEX-92 database [17], were used to contaminate the NIST2001 dataset.

The classifier used in this paper is based on Gaussian Mixture Models (GMMs), similar to the one used in [13]. It models the statistical distribution of training feature vectors for each client. Briefly, a common impostor GMM model (also called a world model) is first obtained from the said 218 speakers using the Expectation-Maximization algorithm [3]. The world model is then adapted to each client's speech features using Maximum *a Posteriori* (MAP) estimation [13]. To make a decision, an average log-likelihood ratio between the client-adapted model and the world model (over all feature frames) is compared to a threshold chosen on development data.

The commonly used Half Total Error Rate (HTER) is used as evaluation criterion[1]. It is defined as $(FAR + FRR)/2$, where FAR is False Acceptance Rate and FRR is False Rejection Rate. Here, we assume that the costs of false acceptance and false rejection are equal and that the prior (class) distribution of clients and impostors are equal as well. The HTER is calculated based on a threshold which itself is estimated *from a development set*. This threshold is estimated such that $|FAR(\theta) - FRR(\theta)|$ is minimised with respect to θ. It is then used to make decisions on an evaluation set. Hence, the HTER is *unbiased* with respect to the evaluation set since its associated threshold is estimated *a priori* on the development set. We call the resultant measure an *a priori* HTER and is used whenever an evaluation set is used. The smaller the HTER is, the better the performance.

4   Empirical Results in Mismatched Conditions

Preliminary studies in [10] showed that the following configuration of SSCs was optimal for the speaker authentication task: 16 centroids, sampled using triangular windows and spaced linearly on the Mel-scale, with delta information and

---

[1] It should be noted the popular Equal Error Rate (EER) *was not used* here because this criterion does not reflect real applications where a threshold must be fixed in advance. Moreover, the use of DET or ROC curves to compare two systems has recently been shown to be erroneous and misleading [1], despite the fact that they are widely accepted in the literature.
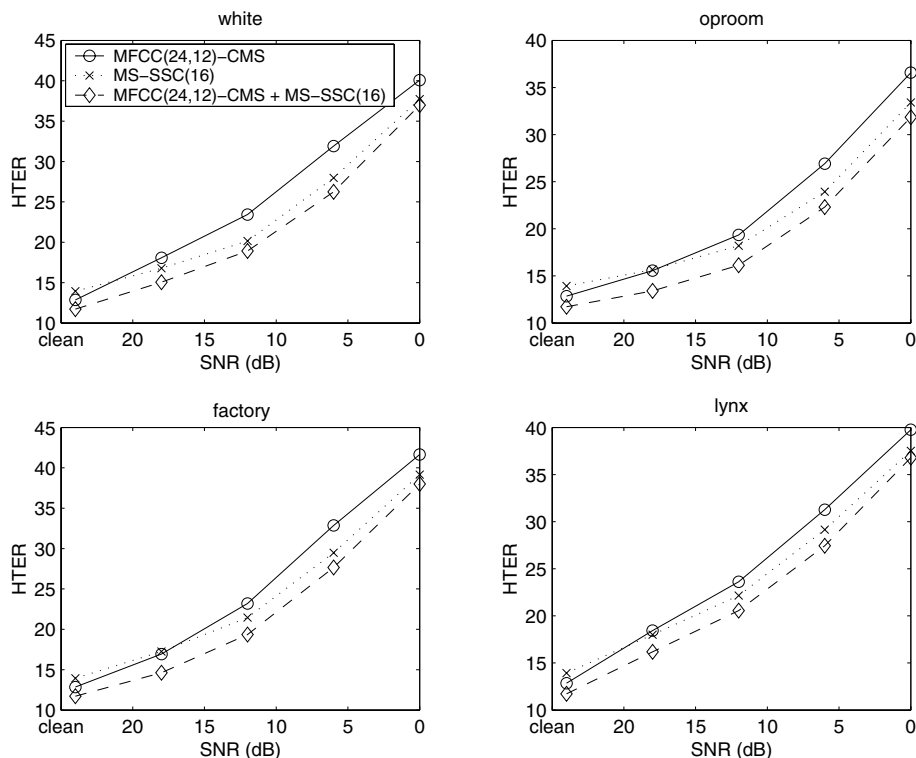
Fig. 2. *A priori* HTERs (in %) of SSCs, MFCCs and MFCC+SSC feature sets on the female evaluation subset of NIST2001 database, under mismatched conditions, using thresholds estimated on clean development data

mean-subtraction. This configuration was used on the female *evaluation* subset (contrary to the *development* subset used in [10]). Furthermore, only bands in the 300-3400 Hz frequency range are used. The log of delta energy is also used. To accomplish energy normalisation, the absolute log energy is not used.

There are two goals: to investigate how resistant SSCs are to mismatched noisy conditions; and to see if concatenation of SSCs with conventional features will improve performance. Two conventional features are used here: LFCCs and MFCCs. The LFCCs are extracted using 24 filterbanks with 16 cepstrum coefficients. MFCCs are extracted using 24 filterbanks with 12 cepstrum coefficients. Several noise types are artificially added to the database at the following Signal-to-Noise Ratios (SNRs): 18, 12, 6 and 0 decibels. Two sets of experiments are conducted: in the first set, MFCCs, SSCs and their combined features are trained in clean conditions and tested in noisy conditions. Hence the combined MFCC-SSC features have $12 + 16 = 28$ dimensions. With delta information, which also has 28 dimensions and log energy, the resultant features have 57 $(28 \times 2 + 1)$ dimensions. Using the same configuration, the second set of exper-
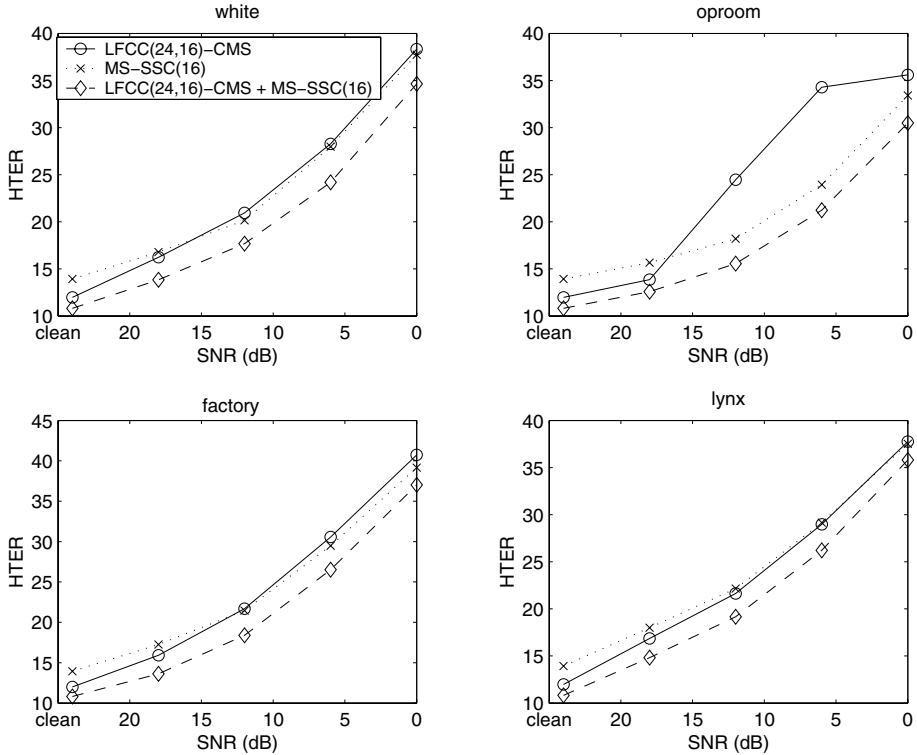
Fig. 3. *A priori* HTERs (in %) of SSCs, LFCCs and LFCC+SSC feature sets on the female evaluation subset of NIST2001 database, under mismatched conditions, using thresholds estimated on clean development data

iments used LFCCs instead. The resultant LFCC-SSC combined features have $65\,((16+16)\times 2+1)$ dimensions. GMMs with 128 Gaussians were used as classifiers for all experiments. The number of Gaussians was found by cross-validation based on the LFCCs features.

The results are shown in Figures 2 and 3 for these two sets of experiments.

For both sets of experiments, it can be observed that MFCCs (respectively LFCCs) perform better than SSCs under clean conditions but are not as good as SSCs under noisy conditions. When MFCCs (respectively LFCCs) are combined with SSCs, the resultant feature sets perform better than any of the features when used alone, in both clean and noisy conditions. Hence, SSCs are potentially useful as complementary features for speaker authentication.

5   Conclusions

Spectral Subband Centroids (SSCs) are relatively new features that exploit the dominant frequency in each subband. The use of SSCs in recent literature has

shown some successes in speech recognition. In this study, the potential use of SSCs in *text-independent speaker authentication* task was studied. Preliminary findings in [10] based on ANOVA and LDA showed that SSCs are potential complementary features to conventional features such as MFCCs. In this paper, we valided these findings using the female development subset of the NIST2001 SwitchBoard database. Based on the results, it is concluded that that SSCs perform somewhat better than MFCCs in noisy conditions; and that combining SSCs with MFCCs (and respectively LFCCs) improves the accuracy of the system in both clean and noisy conditions compared to using any of the feature sets alone. Hence, dominant frequencies represented by SSCs contain speaker discriminative information, somewhat different from what MFCCs (respectively LFCCs) provide. One potential future direction to study the usefulness of the medium to long-term time-trajectory of SSCs. This is motivated by [15], where it is shown that speaker's pitch dynamics (speaker's intonation) are useful for speaker authentication. The advantage of using the time-trajectory of SSCs as compared to pitch dynamics is that not only that the (low frequency) pitch is included, the whole frequency band is actually taken into account.

Acknowledgement

References

[1] S. Bengio, M. Keller, and J. Mariéthoz. The Expected Performance Curve. IDIAP Research Report 03-85, Martigny, Switzerland, 2003. 635

[2] Y. Bengio. *Neural Networks for Speech and Sequence Recognition*. Thompson Computer Press, 1995. 632

[3] C. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1999. 635

[4] E. Chilton and H. Marvi. Two-Dimensional Root Cepstrum as Feature Extraction Method for Speech Recognition. *Electronics Letters*, 3(10):815–816, 2003. 633

[5] R. de Mori and M. Palakal. On the Use of a Taxonomy of Time-Frequency Morphologies for Automatic Speech Recognition. In *Int'l Joint Conf. Artificial Intelligence*, pages 877–879, 1985. 632, 633

[6] H. Hermansky, N. Morgan, Aruna Bayya, and Phil Kohn. Rasta-PLP speech analysis. In *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Processing*, volume 1, pages 121–124, San Francisco, 1992. 631

[7] S. S. Kajarekar and H. Hermansky. Analysis of Information in Speech and its Application in Speech Recognition. In *3rd Int'l Workshop Text, Speech and Dialogue (TSD'2000)*, pages 283–288, Brno, Czech Republic, September 2000. 634

[8] I. Magrin-Chagnolleau, G. Gravier, M. Seck, O. Boeffard, R. Blouet, and F. Bimbot. A Further Investigation on Speech Features for Speaker Characterization. In *Proc. Int'l Conf. Spoken Language Processing*, volume 3, pages 1029–1032, Beijing, October 2000. 631

[9] K. K. Paliwal. Spectral Subband Centroids Features for Speech Recognition. In *Proc. Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, volume 2, pages 617–620, Seattle, 1998. 632, 633

[10] N. Poh, C. Sanderson, and S. Bengio. An Investigation of Spectral Subband Centroids For Speaker Authentication. IDIAP Research Report 03-62, Martigny, Switzerland, 2003. to appear in Int'l Conf. on Biometric Authentication, Hong Kong, 2004. 634, 635, 636, 638

[11] L. Rabiner and B-H Juang. *Fundamentals of Speech Recognition*. Oxford University Press, 1993. 631

[12] D. A. Reynolds. Experimental Evaluation of Features for Robust Speaker Identification. *IEEE Trans. Speech and Audio Processing*, 2(4):639–643, 1994. 631

[13] D. A. Reynolds, T. Quatieri, and R. Dunn. Speaker Verification Using Adapted Gaussian Mixture Models. 10(1–3):19–41, 2000. 635

[14] C. Sanderson. Speech Processing & Text-Independent Automatic Person Verification. IDIAP Communication 02-08, Martigny, Switzerland, 2002. 631

[15] M. K. Sönmez, E. Shriberg, L. Heck, and M. Weintraub. Modeling Dynamic Prosodic Variation for Speaker Verification. In *Proc. Int'l Conf. Spoken Language Processing*, volume 7, pages 3189–3192, Sydney, 1998. 632, 638

[16] M. Kemal Sönmez, Larry Heck, Mitchel Weintraub, and Elizabeth Shriberg. A Lognormal Tied Mixture Model of Pitch for Prosody-Based Speaker Recognition. In *Proc. Eurospeech*, volume 3, pages 1291–1394, Rhodes, 1997. Greece. 632

[17] A. Varga and H. Steeneken. Assessment for Automatic Speech Recognition: NOISEX-92: A Database and an Experiment to Study the Effect of Additive Noise on Speech Recognition Systems. *Speech Communication*, 12(3):247–251, 1993. 635

Maximum Likelihood and Maximum a Posteriori Adaptation for Distributed Speaker Recognition Systems

Chin-Hung Sit[1], Man-Wai Mak[1], and Sun-Yuan Kung[2]

[1] Center for Multimedia Signal Processing
Dept. of Electronic and Information Engineering
The Hong Kong Polytechnic University, China
{tenchsit,enmwmak}@polyu.edu.hk
[2] Dept. of Electrical Engineering
Princeton University, USA
kung@princeton.edu

Abstract. We apply the ETSI's DSR standard to speaker verification over telephone networks and investigate the effect of extracting spectral features from different stages of the ETSI's front-end on speaker verification performance. We also evaluate two approaches to creating speaker models, namely maximum likelihood (ML) and maximum a posteriori (MAP), in the context of distributed speaker verification. In the former, random vectors with variances depending on the distance between un-quantized training vectors and their closest code vector are added to the vector-quantized feature vectors extracted from client speech. The resulting vectors are then used for creating speaker-dependent GMMs based on ML techniques. For the latter, vector quantized vectors extracted from client speech are used for adapting a universal background model to speaker-dependent GMMs. Experimental results based on 145 speakers from the SPIDRE corpus show that quantized feature vectors extracted from the server side can be directly used for MAP adaptation. Results also show that the best performing system is based on the ML approach. However, the ML approach is sensitive to the number of input dimensions of the training data.

1 Introduction

The use of mobile and hand-held devices has become increasingly popular in recent years. However, inputting text and data to these devices is very time consuming and difficult. While speech input is an ideal alternative for this task, mobile phone users tend to use their phones in noisy environment, making robust speech and speaker recognition a challenging task.

Traditionally, speech signals are encoded at the client-side and coded speech is transmitted to the server. Recognition is then performed at the server-side after the reconstruction and parameterization of the decoded speech. However, it has been found that channel- and codec- distortion can degrade recognition

performance significantly [1] [2]. To address this problem, the European Telecom-munications Standard Institute (ETSI) has recently published a front-end pro-cessing standard in which feature vectors are extracted at the client-side and the vector quantised features are transmitted to the server-side for recognition [3] [4]. Since data in the data channel contain the recognition parameters only, codec distortion can be avoided. The technology is commonly referred to as distributed speech recognition (DSR) in the literature.

Research has shown that systems based on the DSR front-end achieve a sig-nificantly better performance than those based on recognizing the transcoded speech [5]. In this paper, we investigate the performance of the maximum likeli-hood (ML) and the maximum a postoriori (MAP) approaches to creating speaker models in the context of distributed speaker verification. We also compare the performance of using 12 MFCCs per speech frame against that of using 12 MFCCs plus 12 delta MFCCs per speech frame.

2   Perturbation of Quantized Vectors

Fig. 1 illustrates the feature-extraction and feature-processing stages of an ETSI-compliance DSR system. Since the feature vectors at the server-side are vector quantized, the distribution of the quantized vectors is discrete. As a result, it is difficult to use the maximum-likelihood approach (based on the EM algo-rithm [6]) to train a Gaussian mixture model whose output represents a contin-uous density function to fit the quantized data. To overcome this problem, we propose to add zero-mean, random vectors to the quantized MFCCs to produce the training vectors. Specifically, the training feature vectors $\mathbf{u}_t$'s are obtained by $\mathbf{u}_t = Q(\mathbf{x}_t) + \boldsymbol{\eta}_t$ where $Q(\cdot)$ and $\mathbf{x}_t$ represent the quantization operation and the unquantized vectors respectively, $\boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma})$ are zero mean Gaussian vectors, $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \ldots, \sigma_D]^T$ represents the standard deviation of $\boldsymbol{\eta}_t$, and D is the dimension of the feature vectors.

The values of σ_j's are estimated as follows. An index table $\{m_{kt}\}$ is built using the unquantized feature vectors $\mathbf{x}_t^{(b)}$ extracted from background speakers during the training phase. More specification, we have

$$m_{kt} = \arg\min_{i \in \{1,2,\ldots,S\}} \|\mathbf{x}_t^{(b)} - \mathbf{v}_i\|, \quad k = 1, 2, \ldots, N \text{ and } t = 1, 2, \ldots, T_k \quad (1)$$

where S is the codebook size ($= 64$ in the ETSI standard) with code vectors $\{\mathbf{v}_i; i = 1, \ldots, S\}$, T_k is the total number of training vectors from background speaker k, and N is the total number of background speakers. Let us denote $\boldsymbol{\sigma}'_k = [\sigma'_{k,1}, \sigma'_{k,2}, \ldots, \sigma'_{k,D}]^T$ as the standard deviation vector corresponding to speaker k. The components of $\boldsymbol{\sigma}'_k$ are found by:

$$\sigma'_{k,j} = \sqrt{\frac{1}{T_k} \sum_{t=1}^{T_k} \left(v_{m_{kt},j} - x_{t,j}^{(b)}\right)^2}, \quad j = 1, 2, \ldots, D \text{ and } k = 1, 2, \ldots, N \quad (2)$$

where $v_{m_{kt},j}$ and $x_{t,j}^{(b)}$ are respectively the j-th component of $\mathbf{v}_{m_{kt}}$ and $\mathbf{x}_t^{(b)}$ and D is the dimension of the feature vectors $\mathbf{x}_t$. Finally, $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \ldots, \sigma_D]^T$ is calculated by:

$$\boldsymbol{\sigma} = \frac{\alpha}{N} \sum_{k=1}^{N} \boldsymbol{\sigma}'_k \quad k = 1, 2, \ldots, N \tag{3}$$

where α is a scaling factor which is determined using enrollment data. Note that we can get access to the unquantized feature vectors derived from the background speakers but not from the client speakers. This is because for client speakers, the server can only extract quantized vectors $Q(\mathbf{x}_t)$ from the DSR bit-stream. As background speakers' speech can be obtained from pre-recorded speech copora, we are always able to implement the DSR front-end in software and obtain the unquantized spectral vectors.

3   Experiments

The SPIDRE corpus, which is a subset of the Switchboard corpus, was used to evaluate the speaker features extracted from different stages of the DSR front-end. SPIDRE consists of 45 target speakers (23 males and 22 females) and 100 non-target speakers. Each utterance contains about 5 minutes speech. There are 4 sessions (conversations) for each target speaker with a total of 3 different telephone numbers (handsets). Each Speaker uses the same handsets in Session 1 and Session 2 while they use a different handset in Session 3 and Session 4. This arrangement allows us to investigate the speaker verification performance under handset matched and handset mismatch conditions.

A collection of all target speakers in the speaker set was used to train a 128-center GMM background model. For each speaker, we trained a personalized GMM to model the characteristics of his/her own voice using the utterances in Session 1. Instead of using the silence detection facility in the ETSI standard, silence was removed by applying the word transcription files provided by SPIDRE.

Two sets of feature vectors, an *unquantized* set and a *quantized* set, were extracted from different stages of the front-end processor of the ETSI standard (see Fig. 1). More specifically, the unquantized set was extracted before feature compression in the terminal front-end while the quantized set was extracted just after server feature processing. To have a better comparison, speaker recognition was performed under three conditions shown in Table 1.

The speaker models were created by two different approaches. In one set of experiments, a maximum likelihood (ML) approach based on the EM algorithm [6] was applied to train the speaker models, while in another set, maximum a posteriori (MAP) adaptation [7] was applied. We have also investigated the performance of using 12 MFCCs (c_1 to c_{12}) and 12 MFCCs plus their first-order derivatives as features. Note that the ETSI standard specifies a total of 39 coefficients ($c_1, c_2, \ldots, c_{12}, \{\ln E \& c_0\}$, and their first- and second-derivatives) per
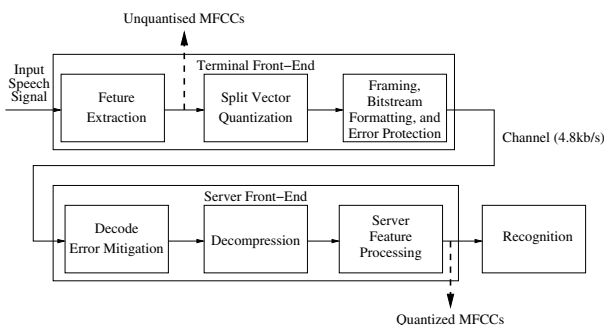
Fig. 1. Feature extraction and processing in an ETSI-compliance DSR system

speech frame. As energy is not speaker-dependent in text-independent speaker recognition, we did not use the energy and its derivatives as features.

For the ML-based speaker models, a personalized 128-center GMM speaker model was trained for each speaker using 12-dimensional MFCCs feature vectors. Due to the curse of dimensionality, however, a 64-center GMM model was trained when 24-dimensional MFCCs feature vectors (MFCCs + delta MFCCs) were used. For the training phase of Condition C shown in Table 1, since the feature vectors were vector quantized, random noise were added to the quantized MFCCs in order to train the Gaussian mixture speaker models. These noise vectors were estimated using the method described in Section 2. The value of α was determined empirically using training data. It was found that $\alpha = 0.26$ gives the lowest EER on the training data.

Fig. 2 shows the projection of the unquantized, quantized, and noise-added feature vectors on the c1-c2 plane. Evidently, the noise vectors are able to make the distribution of the quantized vectors similar to that of the unquantized vectors, which facilitates the ML training of speaker models using the EM algorithm.

For the MAP-adapted speaker models, we adapt a 128-center GMM background model to form a personalized, 128-center GMM speaker model for each speaker in order to fit the quantized feature vectors of that speaker. As each conversation in SPIDRE contains about 5 minutes of speech (2.5 minutes when silence was excluded), we divided the conversation into non-overlapping seg-

Table 1. Feature sets for training the speaker models and for verification. "Unquantized" denotes the MFCCs before vector quantization in the client, and "quantized" denotes the vector quantized MFCCs extracted from the bit-stream in the server-side (see Fig. 1)

	Training	Verification
Condition A	Unquantized	Unquantized
Condition B	Unquantized	Quantised
Condition C	Quantised	Quantised
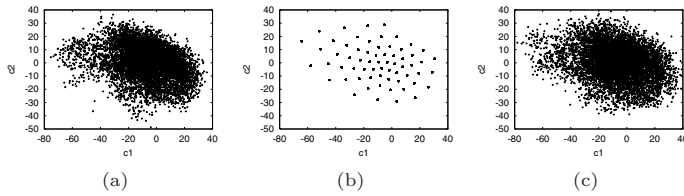
Fig. 2. (a) Projection of unquantized feature vectors on the c1-c2 plane under Conditions A and B. (b) Projection of the quantized feature vectors on the c1-c2 plane before random vectors were added. (c) Projection of quantized training vectors on the c1-c2 plane after adding the random vectors to the quantized vectors in (a)

ments, with each segment contains 200 consecutive feature vectors $\mathbf{Y}$'s. During verification, a vector sequence $\mathbf{Y}$ derived from a claimant's segment was fed to a GMM speaker model ($\mathcal{M}_s$) to obtain a score ($\log p(\mathbf{Y}|\mathcal{M}_s)$), which was then normalized according to

$$S(\mathbf{Y}) = \log p(\mathbf{Y}|\mathcal{M}_s) - \log p(\mathbf{Y}|\mathcal{M}_b) \qquad (4)$$

where $\mathcal{M}_b$ represents the 128-center background model. The normalized score $S(\mathbf{Y})$ was compared with a threshold to make a verification decision. Therefore, each verification decision is based on 200 feature vectors, i.e., 2.8 seconds of speech. For ease of comparison, we collect the scores of 45 speakers, each being impersonated by 100 impostors, to compute the speaker-independent equal error rate (EER) and to produce a speaker detection performance curve [8]. Therefore, speaker-independent decision thresholds were used, and for each session in an experimental setting, there were roughly 3,375 client speaker trials (45 client speakers × around 75 segments per conversation) and 337,500 impostor attempts (100 impostors per client speaker × 45 client speakers × 75 segments per conversation).

4   Results

The experimental results are summarized in Table 2 and the DET curves corresponding to Condition C are plotted in Fig 4. All error rates are based on the scores of 45 genuine speakers and 100 impostors. The conversations in Session 2 of SPIDRE were used to produce the results for the handset matched cases, whereas those in Sessions 3 and 4 were used to produce the results for the handset mismatched cases.

For 12-dimensional MFCCs, it is evident from Table 2 that using the technique of maximum likelihood for enrollment generally gives better performance as compared to using MAP adaptation. This is especially obvious in Condition C, as shown in the DET curve (Fig. 4), where both training and verification parameters were extracted from the server side. Compared to MAP adaptation,

Table 2. Equal error rates (in %) under 3 different training and verification conditions using 2 training approaches with 12-dimensional or 12-D MFCCs plus 12-D ΔMFCCs as features. Refer to Table 1 for the definition of Conditions A, B and C

| | 12 MFCCs | | | | 12 MFCCs + 12 ΔMFCCs | | | |
| | ML | | MAP | | ML | | MAP | |
Condition	handset matched	handset mismatched	handset matched	handset mismatched	handset matched	handset mismatched	handset matched	handset mismatched
A	15.63	20.95	15.96	22.83	22.15	20.87	15.47	18.98
B	15.42	21.10	15.39	22.73	22.41	20.98	15.78	19.29
C	**12.15**	18.61	15.20	22.43	34.07	32.78	16.15	19.09

the maximum-likelihood approach achieves a 20.07% improvement in the handset matched case and 17.03% improvement in the handset mismatched case.

When both 12-dimensional MFCCs and 12-dimensional delta MFCCs were used as features, 64 centers were found to be just right for maximum likelihood training. This means that setting the number of centers to 64 gives the minimum EER with trainable speaker models. However, as shown in Table 2, when compared to the MAP approach with 12-dimensional MFCCs plus its 12 first-order derivatives, the overall performance of the maximum likelihood approach is poorer. The EER is even the worst in Condition C. This may be due to the curse of dimensionality problem. As a result, the MAP approach is better than the maximum likelihood approach when 12-dimensional MFCCs plus 12-dimensional delta MFCCs were used.

It is of interest to see whether the delta MFCCs contain speaker information and help improve the performance of speaker verification. We may observe this by comparing the performance achieved by the MAP approach using 12-dimensional MFCCs plus 12-dimensional delta MFCCs against the one achieved by the same approach but using 12-dimensional MFCCs only. Table 2 (Columns 4-5 and Columns 8-9) clearly shows that the extra information provided by the
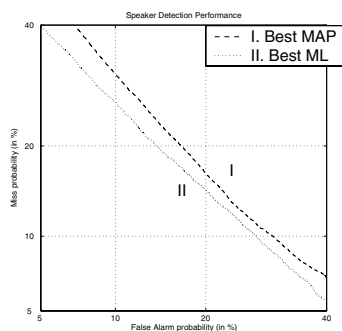
Fig. 3. DET curves for maximum likelihood (ML) and maximum a posteriori (MAP) adaptation under Condition C. Curve I represents results using MAP with 12-D MFCCs plus 12-D Delta MFCCs for all sessions (the best performing MAP) and Curve II represents results using ML with 12-D MFCCs for all sessions (the best performing ML)

delta MFCCs can lower the EER under handset mismatched conditions and maintain the error rates under handset matched conditions. However, it is surprising to see that the maximum likelihood approach cannot fully utilize the speaker information provided by the delta MFCCs, as evident by Column 7 of Table 2. This may be due to the curse of dimensionality problem as the EM algorithm fails to find a solution in the maximum likelihood approach when the number of Gaussian is larger than 64.

Readers should bear in mind that only Condition C reflects the practical situation in DSR systems. Fig. 4 shows the detection error tradeoff curves under this practical situation. In the figure, the scores of all sessions (Sessions 2 to 4) were aggregated. Therefore, this DET plot shows the average performance under both handset matched and handset mismatched conditions. Evidently, the maximum likelihood approach using 12-dimensional MFCCs is better than the MAP approach using 12-dimensional MFCCs plus 12-dimensional MFCCs.

5   Conclusions

Features extracted from different stages of the ETSI-DSR front-end in the context of distributed speaker verification have been evaluated. Both maximum likelihood and maximum a posteriori adaptation have been applied to create speaker-dependent models. Results show that maximum a posteriori adaptation using 12-dimensional MFCCs plus their first-order derivatives can generally reduce the error rates. However, the maximum likelihood approach favors using 12 MFCC only because of the curse of dimensionality problem.

Acknowledgement

References

[1] S. Euler and J. Zinke, "The influence of speech coding algorithms on automatic speech recognition," in *Proc. ICASSP*, 621-624, p. 1994. 641
[2] B. T. Lilly and K. K. Paliwal, "Effect of speech coders on speech recognition performance," in *Proc. ICSLP*, Oct 1996, vol. 4, pp. 2344–2347. 641
[3] D. Pearce, "Enabling new speech driven services for mobile devices: An overview of the ETSI standards activities for distributed speech recognition front-ends," in *AVIOS 2000: The Speech Application Conference*, 2000. 641
[4] ETSI ES 202 050 V1.1.1 (2002-10), *Speech Processing, Transmission and Quality Aspects (STQ); Distributed Speech Recognition; Advanced Front-end Feature Extraction Algorithm; Compression Algorithms*, Oct 2002. 641
[5] H. Kelleher, D. Pearce, D. Ealey, and L. Mauuary, "Speech recognition performance comparison between DSR and AMR transcoded speech," in *Proc. ICSLP'02*, 2002, pp. 1873–1876. 641

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. of Royal Statistical Soc., Ser. B.*, vol. 39, no. 1, pp. 1–38, 1977. 641, 642

[7] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000. 642

[8] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The DET curve in assessment of detection task performance," in *Proc. Eurospeech '97*, Rhodes, Greece, 1997, pp. 1895–1898. 644

Fuzzy Normalisation Methods
for Pattern Verification

Dat Tran

University of Canberra
School of Information Sciences and Engineering, ACT 2601, Australia
`dat.tran@canberra.edu.au`

Abstract. A fuzzy approach to normalisation methods for pattern verification is presented in this paper. For an input token and a claimed identity, a score is calculated and compared with a given threshold to accept or reject the claimed person. The score is regarded as a fuzzy memebership function in this approach. The paper shows how to find effective scores to reduce both false rejection and false acceptance errors. Speaker verification experiments performed on the ANDOSL database and utterance verification experiments performed on the TI46 database showed better results for fuzzy methods.

1   Introduction

In pattern verification, an identity claim is made by an unknown person, and a token of this unknown person is compared with the model for the person whose identity is claimed. If the match is above a given threshold, the identity claim is accepted. In all verification paradigms, there are two types of errors: false rejections and false acceptances. An equal error rate (EER) condition is often used to adjust system parameters so that the two types of errors are equal [12].

Speaker verification (SV) and utterance verification (UV) are considered in this paper. In SV, an identity claim is made by an unknown speaker, and an utterance of this speaker is used to compute a similarity score compared with the speaker model for the speaker whose identity is claimed. In UV, the claim identity is now a linguistic unit such as a word or a phrase. The claimed identity is verified by computing a similarity score to compare the content of spoken utterance with the word or phrase model whose identity is claimed [5].

Normalisation methods have been used in SV to compute the similarity score as the ratio of the claimed speaker's and the impostors' likelihood functions [10]. In this paper, a fuzzy approach to normalisation methods is presented and applied to SV and UV. Speaker verification experiments performed on the AN-DOSL database including 108 speakers and utterance verification experiments performed on the TI46 database containing 26 letters, 10 digits and 10 commands showed better results for fuzzy methods.
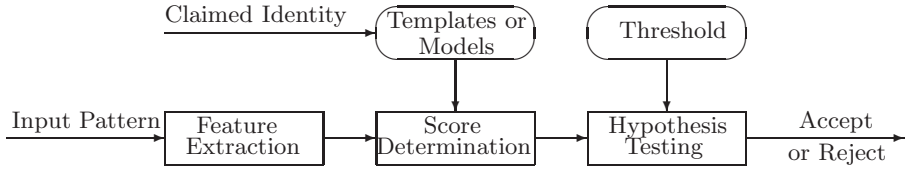
Fig. 1. A Typical Speaker Verification System

2   A Typical Pattern Verification System

In statistical approach, the verification problem is formulated as a problem of statistical hypothesis testing [3]. For a given input token X and a claimed identity, the problem formulation is to test the null hypothesis H_0: X is from the claimed identity, against the alternative hypothesis H: X is not from the claimed identity. According to Neyman-Pearson's Lemma, the optimum test to decide between these two hypotheses is a likelihood ratio test given by

$$S(X) = \frac{P(X|\lambda_0)}{P(X|\lambda)} \begin{cases} > \theta & accept\ H_0 \\ \leq \theta & reject\ H_0 \end{cases} \tag{1}$$

where $P(X|\lambda_0)$ and $P(X|\lambda)$ are the likelihood functions of the claimed person and of all other possible persons, i.e. impostors, respectively. The denominator is called the normalisation term and requires calculation of all impostors' likelihood functions. When the size of the population increases, a subset of B "background" impostor models close to the claimed person is used [4]. Figure 1 presents a typical pattern verification system.

3   Current Normalisation Methods

Depending on the approximation of $P(X|\lambda)$ in (1) by background models' likelihood functions $P(X|\lambda_i)$, $i = 1, \ldots, B$, we obtain different normalisation scores (see [9], [7] and [6]) $S_1(X)$, $S_2(X)$ and $S_3(X)$ as follows

$$S_1(X) = \frac{P(X|\lambda_0)}{\frac{1}{B}\sum_{i=1}^{B} P(X|\lambda_i)} \tag{2}$$

$$S_2(X) = \frac{P(X|\lambda_0)}{\sum_{i=0}^{B} P(X|\lambda_i)} \tag{3}$$

$$S_3(X) = \frac{P(X|\lambda_0)}{\frac{1}{B}\prod_{i=1}^{B} P(X|\lambda_i)} \tag{4}$$

4   Fuzzy Normalisation Methods

4.1   Fuzzy Membership Scores

The space of input tokens can be considered as consisting of two fuzzy subsets for the claimed person and impostors. The similarity score means the fuzzy membership function, which denotes the degree of belonging of an input token to the claimed person. Accepting (or rejecting) the claimed person is viewed as a defuzzification process. According to this viewpoint, fuzzy membership functions can be used as similarity scores. The next task is to find effective fuzzy scores which can reduce both false rejection and false acceptance errors.

4.2   A Solution for the False Rejection Problem

A false rejection of the claimed person can arise because of the use of the background person set. The likelihood values of the background persons are assumed to be equally weighted. However, this assumption is often not true. We propose to use transformed likelihood values to reduce the false rejection error as follows

$$S_f(X) = \frac{f\big[P(X|\lambda_0)\big]}{f\big[P(X|\lambda)\big]} \tag{5}$$

where $f(P)$ is a function of P. In general, there may exist a function $f(P)$ for reducing the EER. For convenience in calculating products of probabilities, the function $f(P)$ should be related to the logarithm function. Moreover, the function should be a quickly increasing function to obtain the EER reduction.

   With $f(P) = (-\log P)^{1/(1-m)}$ and $f(P) = P^{1/n}$, we obtain the fuzzy c-means (FCM) membership score $S_4(X)$ [1] and fuzzy entropy (FE) score $S_5(X)$ [11] as follows

$$S_4(X) = \frac{\big[-\log P(X|\lambda_0)\big]^{1/(1-m)}}{\frac{1}{B}\sum_{i=0}^{B}\big[-\log P(X|\lambda_i)\big]^{1/(1-m)}} \tag{6}$$

$$S_5(X) = \frac{\big[P(X|\lambda_0)\big]^{1/n}}{\frac{1}{B}\sum_{i=0}^{B}\big[P(X|\lambda_i)\big]^{1/n}} \tag{7}$$

where $m > 1$ and $n > 0$ control degree of fuzziness and degree of fuzzy entropy, respectively.

   Applying the function $f(P) = 1/\log P$ to current likelihood scores in (2), (3) and (4), we obtain the following fuzzy scores

$$S_6(X) = \frac{1/\log P(X|\lambda_0)}{1/\log\big[\frac{1}{B}\sum_{i=1}^{B}P(X|\lambda_i)\big]} \tag{8}$$

$$S_7(X) = \frac{1/\log P(X|\lambda_0)}{1/\log\sum_{i=0}^{B}P(X|\lambda_i)} \tag{9}$$

$$S_8(X) = \frac{1/\log P(X|\lambda_0)}{1/\left[\frac{1}{B}\sum_{i=1}^{B}\log P(X|\lambda_i)\right]} \tag{10}$$

4.3 A Solution for the False Acceptance Problem

The false acceptances of impostors can be caused by the relativity of the ratio-based values. For example, the two ratios of $(0.08/0.04)$ and $(0.00008/0.00004)$ have the same value of 2. The first ratio can lead to a correct decision whereas the second one is unlikely since both likelihood values are very low. This problem can be overcome by applying the idea of the noise clustering method [2] in fuzzy clustering. This is implemented by simply adding to the normalisation term a constant membership value $\epsilon > 0$, which denotes the belonging of all input tokens to impostors' fuzzy subset. The general form of the proposed scores after considering the false acceptances and the false rejections is proposed as follows

$$S_{fe}(X) = \frac{f\left[P(X|\lambda_0)\right]}{f\left[P(X|\lambda) + \epsilon\right]} \tag{11}$$

Applying the general form to all proposed scores, we obtain the following scores

$$S_{4\epsilon}(X) = \frac{\left[-\log P(X|\lambda_0)\right]^{1/(1-m)}}{\frac{1}{B}\sum_{i=0}^{B}\left[-\log P(X|\lambda_i)\right]^{1/(1-m)} + \left(-\log \epsilon\right)^{1/(1-m)}} \tag{12}$$

$$S_{5\epsilon}(X) = \frac{\left[P(X|\lambda_0)\right]^{1/n}}{\frac{1}{B}\sum_{i=0}^{B}\left[P(X|\lambda_i)\right]^{1/n} + \epsilon^{1/n}} \tag{13}$$

$$S_{6\epsilon}(X) = \frac{1/\log P(X|\lambda_0)}{1/\log\left[\frac{1}{B}\sum_{i=1}^{B}P(X|\lambda_i) + \epsilon\right]} \tag{14}$$

$$S_{7\epsilon}(X) = \frac{1/\log P(X|\lambda_0)}{1/\log\left[\sum_{i=0}^{B}P(X|\lambda_i) + \epsilon\right]} \tag{15}$$

$$S_{8\epsilon}(X) = \frac{1/\log P(X|\lambda_0)}{1/\left[\frac{1}{B}\sum_{i=1}^{B}\log P(X|\lambda_i) + \log \epsilon\right]} \tag{16}$$

5 Experimental Results

5.1 Speaker Verification

A subset of the Australian National Database of Spoken Language (ANDOSL) corpus [8] consisting of 108 native speakers was used for speaker recognition. Each speaker contributed 200 phonetically rich sentences. All waveforms were sampled at 20 kHz and 16 bits per sample. The data were processed in 32 ms

Table 1. Equal Error Rate (EER) results (%) for the ANDOSL corpus using GMMs with 5 closest background speaker and 5 same-group background speaker

Similarity Score	Speaker Verification Equal Error Rate (%)			
	Five closest background speakers		Five same group background speakers	
	16-mixture GMM	32-mixture GMM	16-mixture GMM	32-mixture GMM
$S_1(X)$	2.2	1.8	2.7	2.1
$S_2(X)$	2.2	1.8	2.7	2.1
$S_3(X)$	3.2	3.0	3.5	3.3
$S_4(X)$	1.9	1.5	2.3	1.7
$S_5(X)$	2.1	1.7	2.3	1.7
$S_6(X)$	1.9	1.5	2.2	1.6
$S_7(X)$	1.9	1.5	2.2	1.6
$S_8(X)$	2.6	2.2	2.8	2.1
$S_{4\epsilon}(X)$	1.7	1.3	1.9	1.3
$S_{5\epsilon}(X)$	1.9	1.5	2.2	1.6
$S_{6\epsilon}(X)$	1.8	1.4	1.9	1.3
$S_{7\epsilon}(X)$	1.8	1.4	1.9	1.3
$S_{8\epsilon}(X)$	2.0	1.7	2.3	1.7

frames at a frame rate of 10 ms. The basic feature set consisted of $12th$-order mel-frequency cepstrum coefficients (MFCCs) and the normalized short-time energy, augmented by the corresponding delta MFCCs to form a final set of feature vector with a dimension of 26 for individual frames. The 16 and 32-mixture Gaussian mixture models (GMMs) were trained for each speaker using the first 10 sentences numbered from 001 to 010. Sentences numbered from 011 to 200 were used for verification.

Experiments were performed on 108 speakers using each speaker as a claimed speaker with 5 closest background speakers or 5 same-group background speakers and 102 mixed-gender impostors (excluding the 5 background speakers) and rotating through all speakers. The total number of claimed test utterances and impostor test utterances are 20,520 (108 claimed speakers x 190 test utterances) and 2,093,040 ((108 x 102) impostors x 190 test utterances), respectively. Equal error rate (EER) results are shown in Table 1. The proposed method $S_{4\epsilon}(X)$ produced the lowest EERs in all cases.

5.2   Utterance Verification

The TI46 database was designed and collected at Texas Instruments (TI). The speech was produced by 16 speakers, 8 females and 8 males, labelled f1-f8 and m1-m8 respectively. The vocabulary contains the ten digits from 0 to 9, ten command words (enter, error, go, help, no, rubout, repeat, speak, stop, yes), and the 26 letters from a to z. For each vocabulary item, each speaker produced 10 tokens in a single training session and another two tokens in each of 8 testing sessions. The speech database was sampled as the ANDOSL database.

In the training phase, 80 training tokens (8 same gender speakers x 1 training session x 10 repetitions) of each letter, digit or command word were used to train a 3-state 2-mixture hidden Markov model (HMM) for that letter, digit or command word by using the HTK software [13]. These HMMs are speaker-independent but gender-dependent, therefore 46 HMMs (26 letters + 10 digits + 10 command words) were trained for each of the two gender groups.

In the verification phase, each letter, digit or command words was used as a claimed utterance with 5 closest background utterances and 40 utterances (excluding 5 background utterances) acting as impostors and rotating through all 46 utterances. So the total number of claimed test utterances and impostor test utterances are 5888 (46 claimed utterances x 8 same-gender speakers x 8 testing sessions x 2 repetitions) and 235520 ((46 x 40) impostors x 8 same-gender speakers x 8 sessions x 2 repetitions), respectively. We also used the HTK to compute the Viterbi scores and regarded them as the likelihood functions in the normalisation methods.

Table 2 presents the EER results for verifying 26 letters, 10 digits and 10 command words. Fuzzy parameters were set as follows: $m = 1.05$, $n = 0.5$ and $\log \epsilon = -31.0$. Most of results obtained from the proposed normalisation methods are lower than the corresponding results obtained from the current normalisation methods. The lowest EER is 5.8% obtained from the normalisation methods $S_{6\epsilon}(X)$ and $S_{7\epsilon}(X)$ using the arithmetic mean for background models and the logarithm function for the transformation.

Table 2. Equal Error Rate (EER) results (%) for the TI46 corpus using 3-state 2-mixture HMMs

Similarity Score	Utterance Verification Equal Error Rate (%)		
	26 letters, 10 digits and 10 commands		
	8 Female speakers	8 male speakers	16 speakers
$S_1(X)$	6.5	6.2	6.3
$S_2(X)$	6.5	6.2	6.3
$S_3(X)$	7.5	7.5	7.5
$S_4(X)$	6.4	5.8	6.1
$S_5(X)$	6.7	6.3	6.5
$S_6(X)$	5.8	5.9	5.9
$S_7(X)$	5.8	5.9	5.9
$S_8(X)$	7.1	7.1	7.1
$S_{4\epsilon}(X)$	6.2	5.8	6.0
$S_{5\epsilon}(X)$	6.5	6.2	6.4
$S_{6\epsilon}(X)$	6.1	5.5	5.8
$S_{7\epsilon}(X)$	6.1	5.5	5.8
$S_{8\epsilon}(X)$	7.0	7.0	7.0

6   Conclusion

Fuzzy normalisation methods for speaker verification and utterance verification have been presented and evaluated in this paper. Based on fuzzy set theory, fuzzy membership functions and a likelihood transformation have been used to reduce both false acceptance and false rejection errors. The noise clustering-based normalisation methods have produced the good results.

Acknowledgement

References

[1] Bezdek, J. C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York and London (1981).   650
[2] Davé, R. N.: Characterization and detection of noise in clustering. Pattern Recognition Letters, **12**(11), (1991) 657-664.   651
[3] Fukunaga, K.: Introduction to Statistical Pattern Recognition. New York: Academic (1972).   649
[4] Higgins, A. L., Bahler, L. and Porter, J.: Speaker Verification using Randomized Phrase Prompting. Digital Signal Processing, **1** (1991) 89-106.   649
[5] Li, Q., Juang, B. H., Zhou, Q., and Lee, C. H.: Automatic Verbal Information Verification for User Authentication. IEEE Transactions on Speech and Audio Processing, **8** (2000) 585-596.   648
[6] Liu, C. S., Wang, H. C. and Lee, C.-H.: Speaker Verification Using Normalisation Log-Likelihood Score. IEEE Trans. Speech and Audio Processing, **4** (1980) 56-60.   649
[7] Matsui, T. and Furui S.: Concatenated Phoneme Models for Text Variable Speaker Recognition. Proceedings of ICASSP'93, (1993) 391-394.   649
[8] Millar, J. B., Vonwiller, J. P., Harrington, J. M. and Dermody, P. J.: The Australian National Database of Spoken Language. Proceedings of the ICASSP'94, **1**(1994) 97-100.   651
[9] Reynolds, D. A.: Speaker identification and verification using Gaussian mixture speaker models. Speech Communication, **17** (1995) 91-108.   649
[10] Rosenberg, A. E., Delong, J., Lee, C. -H., Juang, B. -H. and Soong, F. K.: The use of cohort normalised scores for speaker verification. Proceedings of the ICSLP'92, (1992) 599-602.   648
[11] Tran, D. and Wagner, M.: A Fuzzy Approach to Speaker Verification. International Journal of Pattern Recognition and Artificial Intelligence, World Scientific Publishing Company, **16**(7) (2002) 913-925.   650
[12] Tran, D. and Wagner, M.: Fuzzy C-Means Clustering-Based Speaker Verification. Lecture Notes in Computer Science: Advances in Soft Computing - AFSS 2002, Pal, N. R., Sugeno, M. (Eds.), Springer-Verlag, (2002) 318-324.   648
[13] Woodland, P. C., Hain, T., Johnson, S. E., Niesler, T. R., Tuerk, A., Whittaker, E. W. D. and Young, S. J.: The 1997 HTK Broadcast News Transcription System. Proceedings of the 1998 DARPA Broadcast News Transcription and Understanding Workshop, Lansdowne (1998) 41-48.   653

A Hyperbolic Function Model
for Multiple Biometrics Decision Fusion

Kar-Ann Toh, Xudong Jiang, and Wei-Yun Yau

Institute for Infocomm Research
21 Heng Mui Keng Terrace, Singapore 119613
{katoh,xdjiang,wyyau}@i2r.a-star.edu.sg

Abstract. In this paper, we treat the problem of combining fingerprint and speech biometric decisions as a classifier fusion problem. The Feedforward Neural Network provides a natural choice for such data fusion as it has been shown to be a universal approximator. However, the training process remains much to be a trial-and-error effort since no learning algorithm can guarantee convergence to optimal solution within finite iterations. In this work, we propose a network model to generate different combinations of the hyperbolic functions to achieve some approximation and classification properties. This is to circumvent the iterative training problem as seen in neural networks learning. The proposed hyperbolic functions network model is applied to combine the fingerprint and speaker verification decisions which show either better or comparable results with respect to several commonly used methods.

1 Introduction

The biometric verification problem can be considered as a classification problem wherein a decision is made upon whether or not a claimed identity is genuine with inference to some matching criteria. We thus treat the problem of combining multi-modal biometrics as a classifier decisions combination problem in this paper.

Generally, the approaches for classifiers combination differ in terms of assumptions about classifier dependencies, type of classifier outputs, combining strategies and combining procedures. Two main types of combination can be identified: *classifier selection* and *classifier fusion*. The difference between these two types lies on whether the classifiers are assumed to be complementary or competitive. Classifier selection assumes that each classifier is a "local expert" while classifier fusion assumes that all classifiers are trained over the entire feature space. In this paper, our focus will be on classifier fusion and main effort will be on arriving at a fusion methodology that optimizes the accuracy of the combined decision.

According to the information adopted, three levels of combination can be identified: (i) abstract level, (ii) rank level, and (iii) measurement level. At abstract level, the output information taken from each classifier is only a possible label for each pattern class, whereas at rank level, the output information taken

from each classifier is a set of ordered possible labels which is ranked by decreasing confidence measure. At measurement level, the output information taken from each classifier is a set of possible labels with associated confidence measure. In this way, with the measurement outputs taken from each individual system, the decision is brought forward to the final output of the combined system. We shall work at the measurement level to combine two biometric classifiers.

2   Hyperbolic Functions and Some Properties

2.1   Motivation

The sigmoidal, hyperbolic and Gaussian functions have been widely used in neural network structures as nonlinear discriminant or activation functions. The approximation capabilities of some of these functions have been proven when they are combined in numbers and in layered form. The good approximation and classification capabilities of these networks are usually impaired by the tedious iterative training procedure which does not guarantee convergence to desired optimal solution. In this work, we investigate into possible linear combinations of several hyperbolic functions for biometric decision fusion, particularly for those fusion problems that do not involve a large number of decision outputs to be combined.

2.2   Properties Related to the Powers and Products
     of Hyperbolic Functions

We shall begin with observations on some basic properties of product and power terms of the following hyperbolic functions: $sinh(x)$, $cosh(x)$, $tanh(x)$, $coth(x)$, $sech(x)$ and $csch(x)$ (denoted as $\phi(x)$ as a general basis function).

First, we take a look at the power terms. From (1)-(3) as shown below, we see that high power hyperbolic functional terms can be expressed in terms of low power terms with different signal width or period. Next, we check on the product terms of (4)-(5) where similar effect of dilation or contraction can be interpreted from the summation or subtraction of the variables x and y even if we leave aside the usual phase shift perspective.

The above observations show that *the phase and width parameters within the nonlinear activation functions could be approximated using linear combinations of power and product terms.* We shall explore these observations to construct a network model that provides an effective linear combination of the product and power terms for necessary decision data fusion capability.

$$sinh^2(x) = \frac{1}{2}cosh(2x) - \frac{1}{2}, \;\; cosh^2(x) = \frac{1}{2}cosh(2x) + \frac{1}{2}, \tag{1}$$

$$sinh^3(x) = \frac{1}{4}sinh(3x) - \frac{3}{4}sinh(x), \;\; cosh^3(x) = \frac{1}{4}cosh(3x) + \frac{3}{4}cosh(x), \tag{2}$$

$$sinh^4(x) = \frac{3}{8} - \frac{1}{2}cosh(2x) + \frac{1}{8}cosh(4x),$$

$$cosh^4(x) = \frac{3}{8} + \frac{1}{2}cosh(2x) + \frac{1}{8}cosh(4x), \tag{3}$$

$$sinh(x)sinh(y) = \frac{1}{2}[cosh(x+y) - cosh(x-y)],$$

$$cosh(x)cosh(y) = \frac{1}{2}[cosh(x+y) + cosh(x-y)], \tag{4}$$

$$tanh(x)tanh(y) = \frac{cosh(x+y) - cosh(x-y)}{cosh(x+y) + cosh(x-y)},$$

$$sinh(x)cosh(y) = \frac{1}{2}[sinh(x+y) + sinh(x-y)]. \tag{5}$$

2.3   Hyperbolic Basis Functions Selection

On top of the properties observed so far, there are certain activation charac-
teristics which deserve some attention before the function can be chosen as the
basis function for the combination. Essentially, the output of each basis function
should not be infinitely large at the origin as it gives rise to unstable zero inputs.
Also, the output range is preferably free from any value offset which results in
possible biased approximation. A plot on these functions show that the *cosh*
function needs to be offset by -1 in order to have a zero origin. Since *coth* and
csch functions have functional values at infinity at the origin, we do not include
them in the study. We also exclude the *sech* function for this study since very fre-
quently, it gives rise to matrices which are close to singular or badly scaled. The
remaining hyperbolic function networks under consideration are then labelled as
follows:

$$(\text{SinhNet}): \quad \hat{f}_{sinh} = g(\boldsymbol{\alpha}, sinh(\hat{f}_j)_{j=1,2,...,l}), \tag{6}$$

$$(\text{CoshNet}): \quad \hat{f}_{cosh} = g(\boldsymbol{\alpha}, cosh(\hat{f}_j)_{j=1,2,...,l}), \tag{7}$$

$$(\text{TanhNet}): \quad \hat{f}_{tanh} = g(\boldsymbol{\alpha}, tanh(\hat{f}_j)_{j=1,2,...,l}), \tag{8}$$

where $g(\boldsymbol{\alpha}, \phi)$ denotes the combination function for ϕ with coefficient vector
$\boldsymbol{\alpha} \in \mathcal{R}^K$, $K \geqslant l$. $\hat{f}_j$, $j = 1, 2, ..., l$ denotes the l number of individual bio-
metric measurement outputs to be combined. In next section, we shall derive
a combination function that provides necessary approximation and classification
properties.

3   Function Combinations

Polynomial power series expansion in multivariate form provides a natural platform for spanning a wide variety of combination of product and power terms. However, for a full interaction multivariate polynomials model, the number of terms or parameters becomes very huge for high dimensional and high order problems. Multivariate polynomials expansion becomes impractical due to this prohibitive number of product terms. In view of this problem, we resort to possible reduced models whose number of parameters do not increase exponentially and yet preserving the necessary variety of combinations for the desired approximation or classification capabilities.

3.1   A Reduced Multivariate Expansion Model

In the following, to simplify the expression as well as to avoid possible confusion, the notation of individual unit basis functions on each classifiers $\phi(\hat{f}_j(\boldsymbol{r}, s))$, $j = 1, ..., l$ to be combined will be replaced by x_j, $j = 1, ..., l$ as polynomial inputs.

To significantly reduce the huge number of terms in multivariate polynomials, we use the following reduced polynomial model [1, 2] for function combination:

$$\text{(GRM):} \quad g_{GRM}(\boldsymbol{\alpha}, \boldsymbol{x}) = \alpha_0 + \sum_{j=1}^{r} \alpha_{j+1}(x_1 + x_2 + \cdots + x_l)^j$$

$$+ \sum_{w=1}^{r} \sum_{j=w}^{r} (\boldsymbol{\alpha}_j^T \cdot \boldsymbol{x}^w)(x_1 + x_2 + \cdots + x_l)^{j-w} \quad (9)$$

where $\boldsymbol{x}^w \triangleq [x_1^w, x_2^w, \cdots, x_l^w]$. The total number of terms in (9) can be expressed as: $K = 1 + r(l+1) + \sum_{j=1}^{w}(r-j)l$ with $w \leqslant r$.

When the total number of terms in (9) is plotted against the model order (r) with different w values, for GRM we see that the increase in number of terms is more rapid at low order models, especially when w is large. The increase in number of terms becomes linear when r is large.

The above reduced model can be used in a regularized least squares error objective function to obtain a stable solution. For simplicity and without loss of generality, consider the following second-order model with $w = 1$ and $r = 2$ taking $x_1 = \phi(\hat{f}_1)$ from output of biometric-A and $x_2 = \phi(\hat{f}_2)$ from output of biometric-B:

$$\hat{f}_{GRM2}(\boldsymbol{\alpha}) = g_{GRM2}(\boldsymbol{\alpha}, \boldsymbol{x}) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_1^2 + \alpha_4 x_2^2$$

$$+ \alpha_5(x_1 + x_2) + \alpha_6(x_1 + x_2)^2$$

$$+ \alpha_7 x_1(x_1 + x_2) + \alpha_8 x_2(x_1 + x_2) \quad (10)$$

The matrix $\hat{\mathbf{F}}$ (Jacobian of $\hat{f}_{GRM2}(\boldsymbol{\alpha})$) can then be obtained as

$$\hat{\mathbf{F}} = \begin{bmatrix} 1 & x_{1,1} & x_{2,1} & x_{1,1}^2 & x_{2,1}^2 & (x_{1,1}+x_{2,1}) & (x_{1,1}+x_{2,1})^2 & x_{1,1}(x_{1,1}+x_{2,1}) & x_{2,1}(x_{1,1}+x_{2,1}) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{1,m} & x_{2,m} & x_{1,m}^2 & x_{2,m}^2 & (x_{1,m}+x_{2,m}) & (x_{1,m}+x_{2,m})^2 & x_{1,m}(x_{1,m}+x_{2,m}) & x_{2,m}(x_{1,m}+x_{2,m}) \end{bmatrix}$$
$$(11)$$

and $\mathbf{f} = [f_1, ..., f_m]^T$ is the known desired output vector from training data. In (11), subscripts j and k of the matrix elements $x_{j,k}$ ($j = 1, 2, k = 1, ..., m$) indicate the number of biometric systems and the the number of training samples respectively. The model coefficients $\alpha_0, ..., \alpha_8$ can be obtained using (12) and they can be used to predict future decision outcomes given instances of outputs from biometric-A and biometric-B ($\hat{f}_{1,i}$ and $\hat{f}_{2,i}$, $i = 1, ..., m$ given m training samples).

$$\boldsymbol{\alpha} = (\hat{\mathbf{F}}^T\hat{\mathbf{F}} + b\mathbf{I})^{-1}\hat{\mathbf{F}}^T\mathbf{f}, \tag{12}$$

4   Combinations Search

With the combination model given by (9), our next task is to determine the parameters w and r which are related to model order. As it is known that good training accuracy does not necessary imply good test accuracy, we adopt the cross-validation approach to find the values of w and r that produce good generalization property. In this network learning framework, we partition the entire given data set into three parts: training data $\mathcal{S}_{tr}$, validation data $\mathcal{S}_v$ and test data $\mathcal{S}_{test}$. Here we note that $\mathcal{S}_{train} = \{\mathcal{S}_{tr}, \mathcal{S}_v\}$. The training and the validation data $\{\mathcal{S}_{tr}, \mathcal{S}_v\}$ are used to obtain the best w and r for training. The remaining data $\mathcal{S}_{test}$, which are not used in both training and validation, will then be used to test the performance of the classifier.

Notice that the set $\mathcal{S}_{train}$ can be partitioned into $\mathcal{S}_{tr}$ and $\mathcal{S}_v$ in various ways, typically $N(\mathcal{S}_{tr})/N(\mathcal{S}_{train}) \geqslant 0.5$ with $N(\mathcal{S}_v) = N(\mathcal{S}_{train}) - N(\mathcal{S}_{tr})$ where $N(\bullet)$ denotes the number of elements within the set $\bullet$. In this application we use $N(\mathcal{S}_{tr})/N(\mathcal{S}_{train}) = 0.9$ and perform a corresponding 10-fold cross-validation using the data sets $\{\mathcal{S}_{tr}, \mathcal{S}_v\}_i$, i=1,...,10.

A search for w and r across various combinations can be performed within a certain integer range to locate the optimal 10-fold validation error for the given training set $\mathcal{S}_{train}$ and integer range. The values of w and r corresponding to the optimal 10-fold validation error is then selected for the final training using the set $\mathcal{S}_{train}$.

5   Combining Fingerprint and Speaker Classifiers

In this section, we perform experiments on the above combinations of hyperbolic functions using physical data from two biometrics: fingerprint and speech data. Since SinhNet and CoshNet give rise to matrices which are singular to working
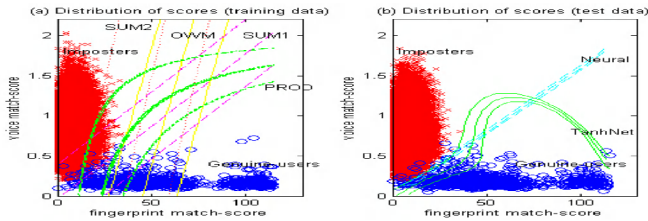
Fig. 1. Distribution of genuine and imposter scores for all users

precision, we shall compare only the performance of the TanhNet with that of OWM and several conventional methods, in the form of Receiver Operating Characteristic curves (ROC).

5.1   Combining Fingerprint and Speaker Verification Decisions

In this experiment, both the databases for fingerprint verification and speaker verification consists of 16 different identities, each comes with 6 different fingers or 6 different words with each fingerprint or word contains 10 samples. A total of 960 samples were thus used for each fingerprint and speech verification system. The fingerprint images were collected using Veridicom's iTouch sensor and the speech data were taken from TIDIGIT database. An arbitrary one-to-one correspondences was taken between the two biometric databases. Both databases are partitioned into two equal sets for training and testing, i.e. set-1 with 480 ($16 \times 6 \times 5$) samples for training and, set-2 with 480 ($16 \times 6 \times 5$) samples for testing.

Depending on individual implementation, the matching output ranges for different modalities may differ significantly. For such cases, numerical sensitivity may be affected and hence a score normalization should be performed between the outputs of different modalities. Otherwise, for reasonably small differences between the scores like in our case, the weighting parameters can be adapted automatically. The distributions of genuine (960 samples) and imposter (45000 samples) scores obtained from matching pairs of biometric samples in each training and test data sets used are shown in Fig. 1. It can be seen from this figure that the separability of the two classes of scores has been improved in the 2-dimensional decision plane as compared to that of the one-dimensional decision based on single biometric. Our problem here is to find the best decision hyperplane from the data given by Fig. 1 and observe how much improvement can be achieved for both training and test sets.

The network structure is selected according to the search procedure listed in section 4 using a 10-fold validation process. Since training of SinhNet and CoshNet resulted in matrices being singular to working precision in many cases, we include only the results for TanhNet. With the regularization parameter empirically set at $b = 10^{-8}$, the search covers the model orders $w \in [1,3] \subset \mathcal{Z}$ and $r \in [1,6] \subset \mathcal{Z}$. This results in a TanhNet network with $(w,r) = (2,6)$ being

chosen for the final training and test since it has the minimum MSE for the 10-fold validation.

The performance of TanhNet is compared to several decision fusion methods: SUM1, SUM2, PROD, OWM, Parzen, RBF, NN and TanhNet. SUM1 and SUM2 denote decision fusion using the simple SUM-rule as seen in [3] with different scalings for each biometric output: $\bar{f}_1 = \hat{f}_1/a$ where $\hat{f}_1 \in [0, 115]$ (fingerprint: high score denotes genuine user) and $\bar{f}_2 = d(c - \hat{f}_2)$ where $\hat{f}_2 \in [0, 2.05]$ (speech: low score denotes genuine user). SUM1 chooses $a = 1.15$, $d = 48$ and $c = 2.05$ by trial-and-error whereas SUM2 uses an automatic nonlinear least-squares search which results in $a = 1.1069$, $d = 11.3778$ and $c = 0.0311$. SUM1 and SUM2 represent somewhat 'optimal' weightings/scalings between the two biometrics and they provide a range of possible performances from adjusting the scaling parameters. PROD denotes the PRODUCT-rule [4] using scalings similar to SUM1. OWM is the linear optimal weighting method and Parzen is the decision based on Parzen window density estimate as seen in [5]. NN denotes the neural network approach and RBF denotes the SVM learning using the Radial Basis Function kernel as seen in [6]. It was observed that the OWM, NN, RBF and TanhNet are not significantly affected by the above scalings and hence scalings similar to that in SUM1 were used.

Fig. 2 shows the training and test ROC results for the biometric decision fusion using those above mentioned methods. It is observed that the performance of the SUM-rule varies between SUM1 and SUM2 from adjusting the scaling parameters. The ROC curves for the original fingerprint and speaker verifications are also included in the same figure for comparison purpose. It can be seen from these plots that the proposed TanhNet improves significantly over the entire operating range of individual fingerprint and speaker verification systems as compared to that of the PROD, SUM2, OWM and Parzen. SUM2 and OWM have very similar ROC performances because they are both least-squares optimized with respect to data distributions. The performance of TanhNet is seen to be comparable to that of NN and only marginally over that of SUM1 whose parameters had been obtained by trial-and-error. It is noted here that even though the Parzen decision strategy is 'optimal' according to Neyman-Pearson lemma, the empirical selection of the window size h affects much the ROC performance.

It is noted here that if good normalization and weighting can be found for SUM1, it can be implemented with less memory than most of the above methods. However, tuning of these weighting parameters remains a trial-and-error effort even though it may be based on a small training data set. The TanhNet reliefs the tuning effort by having a larger parameter memory requirement.

6   Conclusion

In this paper, we proposed a reduced multivariate polynomial network to combine hyperbolic functions where its number of parameters increases almost linearly with model order and number of inputs. Main advantage of this network
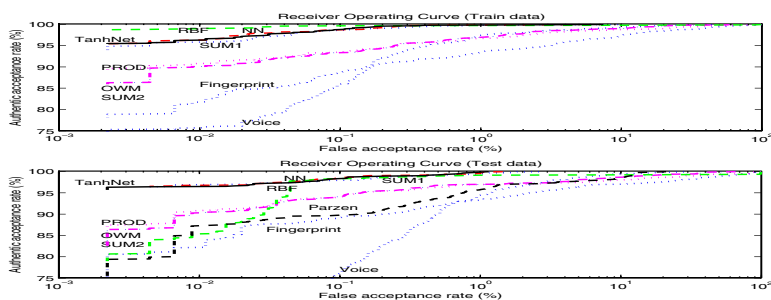
Fig. 2.   ROC   curves   for   SUM1(dotted),   SUM2(dotted),   PROD(dotted), Parzen(dashed), OWM(dashed), RBF(dashed), NN(dashed) and TanhNet(solid)

model over the more complex neural network models is its straight-forward model parameter computation. The network is applied to a biometric decision fusion problem combining fingerprint and speech data. The receiver operating characteristic performance of the hyperbolic function network is found to be better than most compared methods except for the neural network and an empirically tuned SUM-rule which have comparable performances.

References

[1] K.-A. Toh, "Fingerprint and speaker verification decisions fusion," in *International Conference on Image Analysis and Processing (ICIAP)*, (Mantova, Italy), pp. 626–631, September 2003. 658

[2] K.-A. Toh, Q.-L. Tran, and D. Srinivasan, "Benchmarking a reduced multivariate polynomial pattern classifier," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, 2004. (to appear). 658

[3] A. Ross, A. Jain, and J.-Z. Qian, "Information fusion in biometrics," in *Proc. 3rd International Conference on Audio- and Video-Based Person Authentication (AVBPA)*, (Sweden), pp. 354–359, June 2001. 661

[4] L. Hong, A. Jain, and S. Pankanti, "Can multibiometrics improve performance?," in *Proc. Auto ID*, (Summit, NJ), pp. 59–64, 1999. 661

[5] S. Prabhakar and A.K. Jain, "Decision-level fusion in fingerprint verification," *Pattern Recognition*, vol. 35, no. 4, pp. 861–874, 2002. 661

[6] J. Ma, Y. Zhao, and S. Ahalt, "OSU SVM classifier matlab toolbox (ver 3.00)," in *http://eewww.eng.ohio-state.edu/∼maj/osu_svm/*, 2002. The Ohio State University. 661

Combining Fingerprint and Voiceprint Biometrics for Identity Verification: an Experimental Comparison

Yuan Wang, Yunhong Wang, and Tieniu Tan

Center for Biometrics Authentication and Testing
National Laboratory of Pattern Recognition, Institute of automation,
Chinese Academy of Sciences, P.O. Box 2728, Beijing, P.R.China, 100080
{ywang,wangyh,tnt} @nlpr.ia.ac.cn

Abstract. Combining multiple biometrics may enhance the performance of personal authentication system in accuracy and reliability. In this paper, we compare 13 combination methods in the context of combining the voiceprint and fingerprint recognition system in two different modes: verification and identification. The experimental results show that Support Vector Machine and the Dempster-Shafer method are superior to other schemes.

1   Introduction

The emergency of biometrics helps to solve the problems that the traditional methods such as password and IC cards have faced. But there are many problems such as noisy data, non-universality, which may affect the performance of the biometrics system when using a single biometric feature. Multiple biometrics can help to solve several practical problems.

There has been much work on combining multiple biometrics and multi-classifier combination. Roberto [15] presented a person identification system on acoustic and visual features. Ross [7] combined three biometric features: face, fingerprint and hand geometry and Hong [9] integrated face and fingerprints to meet the accuracy and response time requirements of system. Wang [6] used three methods to combine the iris and face recognition system. Except the research of multi-biometrics system, a comprehensive list of classifier combination strategies can be found [5,12]. And a lot of traditional methods such as fuzzy integral [3], naive Bayes [4], Dempster-Shafer fusion rule [2], neural network [6], Fisher discriminant function [6] and logistic regression [4] have been used in multiple classifier system.

Although there has been a substantial amount of work done on combining different biometrics for a variety of purposes, however, not much work has focused on the combination of fingerprint and voiceprint. With the development of mobile communication, we need verify one's identity more frequently. The fingerprint and voiceprint system can be easily applied to the mobile applications to overcome a number of inherent difficulties of the standalone classifier system without much cost increase. In this paper we will try to construct a multi-biometrics authentication system using 13 different combination methods in the context of combining the

voiceprint and fingerprint recognition system and give some theoretical analysis of fusion methods.

The rest of this paper is organized as follows: Section 2 gives a brief overview of multi-biometrics systems as well as the fingerprint and voiceprint recognition system used in our experiment. Section 3 introduces 13 different combination methods and Section 4 presents the experimental results. Finally, Section 5 concludes the paper.

2   Multi-Biometrics

2.1 Biometrics and Multi-Biometrics

A biometrics system can be deemed as a pattern recognition system, which may work in two different modes: identification and verification. An identification system can be measured by Correct Recognition Rate (CRR) and the performance of a verification system measured by Receiver Operation Characteristic Curve (ROC Curve).

For the reason of noise, non-universality of single biometric feature, multi-biometrics system is proposed. Suppose N biometrics features $[F_1, F_2, \cdots\cdots, F_N]$ are used to verify the claimed identity. Let $\{1,2,\cdots\cdots,c\}$ be the label set of c classes (c persons in this paper). For each person x, using F_k feature we will get c matching scores from the matching with other $c-1$ persons and himself. So we can get a matrix $H(x)$ for the person x [1]:

$$H(x) = \begin{bmatrix} \mu_{11} & \mu_{12} & \cdots & \mu_{1N} \\ \mu_{21} & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \mu_{(c-1)N} \\ \cdots & \cdots & \mu_{c(N-1)} & \mu_{cN} \end{bmatrix} \tag{1}$$

where $\mu_{i,j}$ denotes the matching score of x and the *ith* person given by the *jth* classifier.

2.2 Fingerprint and Voiceprint Authentication

A critical step in fingerprint verification system is to automatically and reliably extract minutiae from the input fingerprint images. We use the minutia extraction algorithm presented in [14]. With these minutia features, an alignment-based elastic matching algorithm is used. For the voiceprint recognition system, at first we divide the input speech into several small segments with a fixed length and for each segment a 39-dimensional MFCC (Mel Frequency Cepstral coefficients) feature vector is extracted. With these MFCCs, we train the Gaussian Mixture Model (GMM) for each speaker [13].

3   Algorithms for Multi-biometrics Combination

In this part, we will outline the methods employed for fusion in two modes: verification and identification.

3.1 Verification Mode

In this mode, the incomer X will be only matched with the template of the person he claims. We treat the fingerprint and voiceprint matchers output x_1 and x_2 as a feature vector $X = (x_1, x_2)$. Then we can use any known classifiers to determine the separation bound between imposter and client. The fusion methods we employed to combine fingerprint and voiceprint are introduced as follows:

T1: Logistic regression (LOG)

This method depends on the assumption that the fingerprint and voiceprint recognition system is independent and that the conditional density of the client and imposter class can be written as the logistic distribution function [4].

T2: Fisher discriminant classifier

Fisher rule is a well-known linear discriminant, which has been widely used in the field of pattern recognition. It designs the discriminant bound by performing dimensionality reduction using linear projection and still preserve linear separability [6].

T3: User-specific weighted sum rule

This method is proposed by Jain [9]. The so-called 'user-specific weight' means that the weights of the matching scores of different classifier when summing are selected specifically for each person. In our experiment, we estimate the user-specific weights by exhaustive search in the space $\{(W_1, W_2) \mid W_1 + W_2 = 1\}$ to find the weights corresponding to the minimum error rate on the train set.

T4: ENN (Nearest-Neighbor with Class Exemplars)

Suppose the norm of x, $y = \sqrt{x_1^2 + x_2^2}$, and we find the within class scatter of y is 0.3372 while that of X is 0.665. So we use y as the classification feature. We deem the mean value of y of imposter and client as their exemplar value and compare the norm of an unclassified x with the two-exemplar values and make decision.

T5: Support vector machine

The standard SVM produces a non-linear classification boundary in the original input space by constructing a linear boundary in a transformed version of the original input space. The classification result is showed in Figure 1.

3.2 Identification Mode

T6: Minimum, maximum, average and product Rule

According to Section 2, for a given input pattern x we will get a matrix $H(x)$; find the minimum (or maximum, or average, or product) values of every row, which are the similarity measure between x and pre-stored templates. The output of these rules simply consists in assigning x to class i if the ith value is the maximum of them. A tie will be broken by choosing one randomly.

T7: Naive Bayes rule

This method can be found in many published papers. In this method, the independence of fingerprint and voiceprint recognition system must be satisfied [4].

T8: Fuzzy integral rule

Fuzzy integral has been used for classifier fusion in several applications. Each classifier produces a confidence value for every class, which represents the worth of the corresponding classifier for the class. The overall confidence for the class is the fuzzy integral value. The input will be assigned to the class with the largest integral value [3].

T9: Decision templates

This method is proposed by Kuncheva [1]. Suppose $Z = \{z_1, \cdots\cdots, z_n\}, z_j \in \Re^N$ be the training set. In this method, at first the decision template (DT_i) is trained for each class, which is in fact the average of the matrix $H(x)$ of the elements whose label is i in the training set Z. For an unclassified input x, we deem DT_i and $H(x)$ as a fuzzy set, compute the similarity between them. Here we use four kinds of similarity measures in our experiment. The first measure is the Euclide distance of the two sets and the other three is denoted as following:

$$S_1(A, B) = \frac{\|A \cap B\|}{\|A \cup B\|} \qquad S_2(A, B) = 1 - \|A \nabla B\| \qquad S_3(A, B) = 1 - \|A \Delta B\|$$

T10: Dempster-Shafer rule

In this frame of the evidence theory, the best representation of support is a belief function rather than a Bayesian mass distribution. The theory embraces the familiar idea of assigning numbers between 0 and 1 to indicate the degree of support but, instead of focusing on how this numbers are determined, it concerns the combination of degrees of belief. Here, we use the algorithm proposed by [1].

4   Experimental Results and Discussions

4.1 Experimental Results

The experiment is conducted on the database that contains 44 persons and for each person 24 fingerprint samples and 24 voiceprint samples have been collected. The collection course is divided into two sessions, which have an interval of one month. For each person, 12 samples are captured in each session. We can get the matching scores of each sample with the 44 pre-stored templates by the designed fingerprint and voiceprint recognition system. For the fusion methods that need training, we use the first four samples of each person to constitute the training set and the rest to constitute the testing set.

All of the results of these methods including those introduced in verification mode are presented in Table 1. Here we did not give the CRR of SVM because it is computing costly for SVM to achieve a multi-classification.

Table 1. The Correct Recognition Rate of different methods.

| Methods	CRR	Methods	CRR	Methods	CRR
Fingerprint only	98.2%	Voiceprint only	93.7%	Max Rule	99.6%
				Min Rule	98.6%
Naïve Bayes	98.2%	Fuzzy Integral	98.2%	Average Rule	99.1%
Fisher	98.8%			Product Rule	98.9%
Dempster-Shafer	99.9%	Decision template	S1 99.4%	User-specific	99.8%
			S2 99.1%	Logistic	99.1%
ENN	99.3%		S3 97.1%		
			Eu 99.1%		

Next, we present the ROC curve of different techniques in Figure 2. In the mode of verification, we use the leave-one-out strategy to compute the ROC curve.

4.2 Discussions

Based on the above results and analysis, we can draw some conclusions and find some issues that need further investigating. The goal of our experiment is to compare the performance of different combination methods in the context of combining fingerprint and voiceprint recognition system. As Table 1 shows, most of the methods are superior to the fingerprint recognition system, which is the best single system. Especially the Dempster-Shafer method and user-specific weighted sum rule both give a very good result with only one and two error classified samples in total 880 samples respectively.

And according to Figure 2, we can find that in the verification mode the SVM method is better than user-specific weighted sum rule and these two methods are the best two methods for verification in our fusion experiment. As shown in Figure 1, the

distribution of the imposter and client data cannot be separated by a liner function. And SVM is based on the Structural Risk Minimization (SRM) principle. So we can see in Figure 2 the decision boundary computed by SVM method is fitted well to the data and SVM give a better performance comparing with Fisher rule that adopts linear discriminant function.
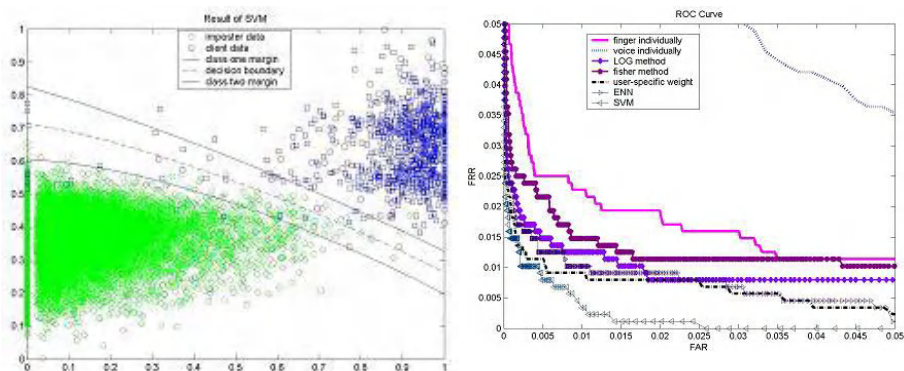
Fig. 1. The distribution of client and imposter. The line in the figure denotes the classification boundary computed by SVM method

Fig. 2. The ROC curve from different methods

For the simple aggregation methods (Min, Max, Average and Product Rule) that need not train beforehand, they all give a better CRR compare with single biometrics systems. Max and Average rule outperform the other two methods. According to theoretical and experimental results reported in other papers [12], researchers agree that fixed rules usually perform well for the combination of classifiers exhibiting similar performance and methods that need training give a better performance for the combination of classifiers exhibiting different accuracy. In our experiment, the difference of accuracy between fingerprint and voiceprint recognition system is not very large, but it is not very clear that fixed rules outperform those trained rules.

The decision template method has been proved in [1] to get a good performance with both data sets using 10 different similarity measures. But in our experiment, the choice of similarity measure affects the performance of system significantly. We adopted four kinds of measures that give the best performance within the 10 proposals in [1]. However, when using S_3 measure, the system fails to get improvement.

From the experiment results based on combining fingerprint and voiceprint recognition scores, we can find that user-specific weighted sum and SVM get better performance among all the fusion manners. Since the test data are not always as good as the training data, the feature of a specific user may not typical. Such data cannot be divided by a linear function. So the nonlinear classification method: User-specific, DS rule and SVM can give a good solution.

As discussed above, we compared 13 different methods in our experiment. But in our experiment, the test data set is too small and the samples in the data set are mostly "good" samples, which means the samples are collected in the normal condition, so the result of comparison is not very convincing. In the future work, we will investigate these methods in a larger data set and include some "bad" samples. We will also try to add some other biometrics such as gait and face to find out the best method in a more universal context. And with combining more than two biometric features, we can investigate the relationship of the system performance and the characteristics of the combination classifiers.

5   Conclusion

Fusion of multiple biometrics has recently gained more interests with an increasing emphasis on security. In this paper, we have compared 13 different classifier-combination methods based on the fingerprint and voiceprint matching scores in two different modes: identification and verification. Dempster-Shafer combination and SVM method are proved to get the best performance among all the fusion methods we employed in this paper.

Acknowledgements

References

[1]   L.I.Kuncheva, James C.Bezdek, Robert P.W. Duin, Decision templates for multiple classifier fusion: an experimental comparison, Pattern Recognition 34(2001) 299-314.
[2]   L. Xu, A. Krzyzak, C.Y. Suen, Methods of combining multiple classifiers and their application to handwriting recognition, IEEE Trans. Systems Man Cybernet. 22 (1992) 418-435.
[3]   P.D. Gader, M.A. Mohamed, J.M. Keller, Fusion of handwritten word classifiers, Pattern Recognition Lett. 17 (1996) 577-584.
[4]   P. Verlinde, A Contribution to Multi-Modal Identity Verification Using Decision Fusion, PhD Thesis, Department of Signal and Image Processing, Telecom Paris, France, 1999.
[5]   D.M.Tax, M.Breukelen, P.W.duin, J.Kittler   Combining multiple classifiers by averaging or by multiplying?, Pattern Recognition 33(2000) 1475-1485
[6]   Y.H.Wang, Tieniu Tan, A.K.Jain, Combining Face and Iris Biometrics For Identity Verification, AVBPA'2003
[7]   A.K.Jain, Arun Ross, Learning user-specific parameters in a multibiometric system, Proc. International Conference on Image Processing (ICIP), Rochester, New York, September 22-25, 2002
[8]   Kumar, David C.M.Wong, Helen C.Shen, A.K.Jain, Personal Verification using Palmprint and Hand Geometry Biometric, Pattern Recognition , vol. 36, pp. 371-381, 2003

[9]   L.Hong, A.K.Jain. Integration Faces and Fingerprints for Personal Identification. IEEE Transaction Pattern Analysis and Machine Intelligence, 1998,20(12), 1295~1300.

[10]   Steve R.Gunn, Support Vector Machines for Classification and Regression, Technical Report

[11]   Ji Zhu, Trevor Hastie, Support Vector Machines, Kernel Logistic Regression and Boosting, 3rd Int. Workshop on Multiple Classifier Systems (MCS 2002), Cagliari, Italy, June 2002, Springer-Verlag, LNCS.

[12]   Roli, F., Raudys, S., Marcialis, G.L: An experimental comparison of fixed and trained fusion rules for crisp classifier outputs. 3rd Int. Workshop on Multiple Classifier Systems (MCS 2002), Cagliari, Italy, June 2002, Springer-Verlag, LNCS.

[13]   Peng Ding, Yang Liu, Bo Xu, Factor Analyzed Gaussian Mixture Models for Speaker Identification, In Proc. ICSLP 2002

[14]   L. Hong, A.K. Jain, R. Bolle and S. Pankanti, "Identity Authentication Using Fingerprints", Proc. of First Int'l Conf. On Audio and Video-Based Biometric Person Authentication, Switzerland, pp. 103-110, March 1997.

[15]   R. Brunelli and D. Falavigna, "Person identification using multiple cues," IRST(Insituto per la Ricerca Scientifica e Tecnologica), TR9401-08, 1994

An Evolutionary Algorithm Based Approach for Dynamic Thresholding in Multimodal Biometrics

Kalyan Veeramachaneni, Lisa Ann Osadciw, and Pramod Varshney

Department of Electrical Engineering and Computer Science
Syracuse University, Syracuse, NY 13244-1240
{kveerama,laosadci,varshney}@syr.edu

Abstract. This paper presents an evolutionary approach to the sensor management of a biometric security system that improves robustness. Multiple biometrics are fused at the decision level to support a system that can meet more challenging and varying accuracy requirements as well as address user needs such as ease of use and universality better than a single biometric system or static multimodal biometric system. The decision fusion rules are adapted to meet the varying system needs by particle swarm optimization, an evolutionary algorithm. This paper focuses on the details of this new sensor management algorithm and demonstrates its effectiveness. The evolutionary nature of AMBM allows it to react in pseudo-real-time to changing security needs as well as user needs. Error weights are modified to reflect the security and user needs of the system. The AMBM algorithm selects the fusion rule and sensor operating points to optimize system performance in terms of accuracy.

1 Introduction

This paper presents an algorithm that adaptively derives the optimum Bayesian fusion rule as well as individual sensor operating points in a system. The algorithm bases its design on system requirements, which may be time-varying. Other researchers have published numerous experimental results demonstrating the effectiveness of fusion in biometrics [1], [2], [3]. Ad-hoc techniques have been demonstrated to be effective but not always optimum in terms of accuracy [4]. These either contain only a few rules or fix the F_{AR} (false acceptance rate) and F_{RR} (false rejection rate). In contrast, the adaptive, multimodal biometric management algorithm (AMBM), presented here, comprehensively considers all fusion rules and all possible operating points of the individual sensors. The optimum Bayesian fusion rule is designed based on the current situation. As the situation changes with time, AMBM revises the fusion rule and the operating points. The paper discusses the application of a sensor management algorithm to a biometric security problem. A block diagram of this is shown in Figure 1. The system has N biometric sensors, a Particle Swarm Optimizer (PSO), a Bayesian Decision Fusion processor and a Mission manager. The mission manager provides the PSO the costs, which reflect system requirements. The PSO searches for an optimal configuration of the sensors with a corresponding fusion rule matching those requirements. The threshold values for each biometric sensor are also selected since they are function of the sensor F_{AR} determined by PSO. These sensors now operate at these thresholds determined by PSO. The Bayesian Decision Fusion

processor fuses the decisions from multiple sensors using the optimum fusion rule from the PSO into a final decision. New rules and thresholds are found when the PSO receives new costs from the mission manager

The next section describes the particle swarm optimization algorithm and the mapping of the multimodal biometrics problem to this algorithm. Results achieved by applying PSO and the AMBM are discussed in Section 3. Finally, conclusions are presented in Section 4.

2   Particle Swarm Optimization

Particle swarm optimization (PSO) was originally introduced in terms of social and cognitive behavior by Kennedy and Eberhart in 1995 [6]. The individuals, called particles, are flown through the multidimensional search space. Each particle tests a possible solution to the multidimensional problem as it moves through the problem space. The movement of the particles is influenced by two factors: the particle's best solution (*pbest*) and the global best solution found by all the particles (*gbest*). These factors influence the particle's velocity through the search space by creating an attractive force. As a result, the particle interacts with all the neighbors and stores in its memory optimal location information. After each iteration the *pbest* and *gbest* are updated if a more optimal solution is found by the particle or population, respectively. This process is continued iteratively until either the desired result is achieved or the computational power is exhausted.
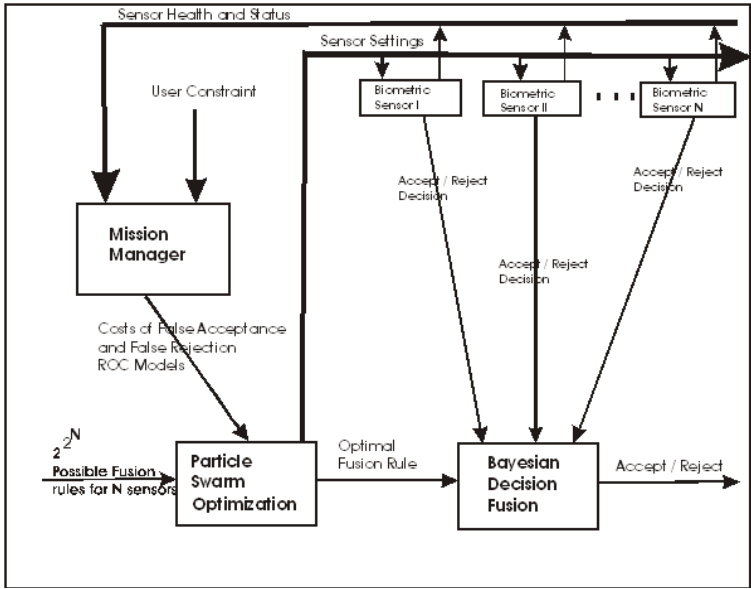

Fig. 1. Illustration of Adaptive Multimodal Biometric Fusion Algorithm

The PSO formulae define each particle in the D-dimensional space as $X_m = (x_{m1}, x_{m2}, x_{m3}.......x_{mD})$ where the subscript m represents the particle number and the second subscript is the dimension. The memory of the previous best position is represented as $P_m = (p_{m1}, p_{m2}, p_{m3}, ... p_{mD})$ and a velocity along each dimension as $V_m = (v_{m1}, v_{m2}, v_{m3}, ... v_{mD})$. After each iteration, the velocity term is updated. The particle's motion is influenced by its own best position, P_m, as well as the global best position, P_g. The velocity is updated by

$$V_{md}^{(t+1)} = \omega \times V_{md}^{(t)} + rand(1) \times \psi_1 \times (p_{md} - X_{md}^{(t)}) + rand(1) \times \psi_2 \times (p_{gd} - X_{md}^{(t)}) \quad (1)$$

and the position is updated by

$$X_{md}^{(t+1)} = X_{md}^{(t)} + V_{md}^{(t+1)} \quad (2)$$

Constants ψ_1 and ψ_2 determine the relative influence of the *pbest* and *gbest*. Often these are set to same value giving equal weight to both. The memory of the PSO is controlled by ω.

Each particle in multimodal biometrics problem has '$N+1$' dimensions, where N is the number of sensors in the sensor suite. The first N dimensions are the sensor thresholds for all sensors in the sensor suite. The '$N+1$' th dimension is the fusion rule, which determines how all the decisions from the sensors are fused**.** Hence the representation of each particle is

$$X_m = \{\lambda_{m1}, \lambda_{m2}, \lambda_{m3}, ...\lambda_{mn}, f_m\} \quad (3)$$

The sensor thresholds are continuous. In contrast, the fusion rule (*f*) is a binary string having a length of

$$s = \log_2 p \quad (4)$$

where $p = 2^{2^N}$. The rule f may be represented by an integer value varying from $0 \le f \le p-1$. This real value representation proved to be inefficient due to all the bounding that was required in AMBM. For this binary search space, the binary decision model and search strategy as described in [5] is being used. This binary decision model results in a much faster convergence than the integer number model.

In the algorithm instead of evolving the thresholds explicitly, the false acceptance rates (F_{AR}) are evolved in the PSO for each of the sensors. Thresholds are calculated based on the F_{AR} and sensor models determined a priori and updated. The PSO minimizes the cost, (5). The global F_{AR} and the F_{RR} for the optimum fusion rule, f, are calculated directly from the fusion rule and individual sensor F_{AR_i} and F_{RR_i} [7].

The PSO cost function, which is objective function optimized by the PSO, is

$$E_a = C_{FA} \times (F_{AR_a} - F_{AR_d}) + (2 - C_{FA}) \times (F_{RR_a} - F_{RR_d}) \quad (5)$$

F_{AR_a} and F_{RR_a} are the achieved global false acceptance and rejection rates estimated using the a priori sensor models. F_{AR_d} and F_{RR_d} are the desired global false acceptance and rejection rates from the mission manager. AMBM adapts to the required error rates preventing overdesign of the rule by using (5) as the PSO objective function. The PSO algorithm uses this cost function to evaluate the particles and determine the global best *"gbest"* among them, compare the particle's previous best *"pbest"* and the *"present"* positions. A population of 10 particles is randomly initialized in the search space and the algorithm is run for a pre-specified number of iterations evolving the solution closer to optimum.

3   The AMBM Performance Results

This section presents the solutions for the complete range of costs (0 through 2). A Monte-Carlo simulation ran the AMBM algorithm for 100 times for the same cost with 1000 PSO iterations per run. The biometric sensor suite consists of two sensors, which are modelled by the Gaussian distributions described in Table 1. In this case, sensor 2 is more accurate and dominant in the sensor suite. Each Monte Carlo run successfully converged to an appropriate solution with comparable minimum PSO costs for every false acceptance cost. Although the runs did not consistently select the same solution, the selected solution met the AMBM performance criteria.

The solutions consist of the rule and the sensor operating point defined by its false acceptance rate and false rejection rate. The sensor threshold can be computed from the sensor's error rates and distributions. Figure   shows the percentage of trials for which a particular rule has been selected as a function of the false acceptance cost. Table 3 summarizes the results showing a ranges of costs and the most probable rule selected. The "OR" rule, f_8, is more probable when the cost of false acceptance is low (0-0.5). Due to sensor 2's dominance, there is a range of costs (0.5-1.5) for which the system simply ignores sensor 1's decisions. For higher costs of false acceptance (1.5-1.9), the "AND" rule, f_2, is most frequently selected

Table 1. Means and Standard Deviations of the Gaussian Distributions for Two Sensors

Parameter	Sensor 1	Sensor 2
Mean, μ_1, for imposter	12	12
Std. Deviation, σ_1, for imposter	8	8
Mean, μ_2, for genuine	36	40
Std. Deviation, σ_2, for genuine	12	4

Table 2. Particle Swarm Optimization Parameters

ψ_1	ψ_2	ω	No. of particles
1	1	0.8	10

The swarm optimization algorithm, however, is able to find multiple solutions involving different fusion rules for one particular global false acceptance cost with equivalent PSO costs. For example, the swarm selected rule f_2 76% of the times, f_6 19% of the times and f_8 5% for a C_{FA} of 1.9. A comparison of these three rules (for a C_{FA} of 1.9) in terms of minimum PSO cost indicate two things: the operating points were different and standard deviation of the minimum PSO cost was as low as 6.9 e-4 with a mean of 0.0105. The operating points of sensors differed according to the selected rule as demonstrated in Table 4. The low standard deviation of the resulting minimum PSO cost (6.9 e-04 for a C_{FA} of 1.9) over the 100 trials indicates that optimality has been achieved independent of the rule chosen. Thus, the AMBM algorithm selected optimal rules and sensor operating points for two sensors with approximately equal PSO costs. A second example is provided in Table 4 for a cost of 0.1. The operating points and rules are very different for nearly equal PSO costs. This can be extended to any number of sensors in the sensor suite.
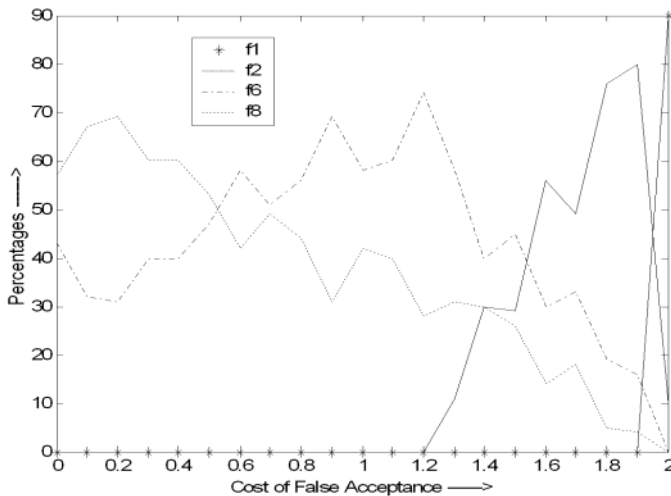
Fig. 2. Probability of Selecting Fusion Rule vs. Cost of False Acceptance

Table 3. Table Showing the Ranges of CFA and Corresponding Rule

Cost of False Acceptance	Recommended Rule
0.0 - 0.5	f_8 (OR)
0.5 - 1.5	f_6 (Second Sensor Only)
1.5 - 1.9	f_2 (AND RULE)
1.9 - 2.0	f_1 (NONE)

Table 4. Detailed Analysis of PSO Solution Demonstrating Convergence

μ: Mean ; σ : variance; f: fusion rule

C_{fa}	Rule	SENSOR 1				SENSOR 2				Minima Achieved	
		FAR		FRR		FAR		FRR			
		μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
0.1	f_3^*	1e-3	5e-4	0.49	0.03	0.03	2e-3	9e-4	2e-4	4e-3	3e-4
	f_6	0.27	0.23	0.10	0.09	0.03	4e-5	5e-4	2e-6		
	f_2	0	0	0	0	0	0	0	0		
1.9	f_2^*	0.46	0.02	0.03	3e-3	5e-3	2e-4	0.03	1e-3	0.01	7e-4
	f_6	0.1	0.18	0.2	0.1	4e-3	8e-5	0.05	1e-3		
	f_3	3e-4	2e-4	0.62	0.05	3e-3	6e-4	0.07	0.01		

*most probable rule

Table 5. Comparison Analysis of Most Probable PSO Solution over Small Cost Difference

μ : Mean; σ: variance; f : fusion rule

C_{fa}	Rule*	SENSOR 1				SENSOR 2				Minima Achieved	
		FAR		FRR		FAR		FRR			
		μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
1.4	f_6 40%	0.30	0.26	0.1	0.09	9e-3	3e-3	0.01	9e-3	0.018	7e-7
1.6	f_2 56%	0.83	0.02	4e-3	6e-4	8e-3	2e-4	0.01	7e-4	0.018	3e-3

Table 5 shows the values for two more costs, i.e, CFA=1.4 and CFA=1.6. AMBM selects f6 40% of the time for cost 1.4 and f2 56% of the time for a cost 1.6. The swarm also selects the operating points to achieve optimal solutions. With a cost of 1.4 the swarm uses only single sensor, which is sensor 2 which has higher accuracy and ignores the other sensor. Sensor 1's operating parameters actually don't matter for this case and have no impact on performance. As the cost of false acceptance increases to 1.6, AMBM selects the AND rule so that even sensor 1 is needed to achieve the desired accuracy. In conclusion, the analysis of the rules and sensor configurations in Table 3, Table 4, and Table 5 supports the following; the use of false acceptance cost as a mechanism to effectively control system performance, PSO's ability in finding an optimal solution for different costs (CFA) by changing operating points of each sensor as well as the fusion rule.

4   Conclusions

The paper presents a evolutionary algorithm based approach to set the sensor thresholds and also fusion rule for multimodal biometric system. The evolutionary algorithm presented here successfully searches the entire search space and results in a optimum setting for all the sensors as well as the system.

The AMBM algorithm, as presented, focuses on system accuracy. Universality and ease of use can easily be incorporated into the PSO cost so that AMBM automatically takes these factors into account in the selection of the solution. The AMBM algorithm presented can be easily modified to address these important issues and any others that may emerge in a biometric security system. The real-time adaptability of the AMBM algorithm is the key attribute making it a strong sensor management solution.

References

[1]   Lin Hong and Anil Jain, "Integrating Faces and Fingerprints for Personal Identification", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 12, Dec., 1998, pp. 1295 - 1307.
[2]   Salil Prabhakar and Anil Jain, "Decision-level Fusion in Fingerprint Verification", Pattern Recognition, vol. 35, 2002, pp. 861-874.
[3]   K. Jain, S. Prabhakar, S. Chen, "Combining multiple matchers for a high security fingerprint verification system", Pattern Recognition Letters 20 (11-13) (1999) 1371-1379.
[4]   Robert W. Frischholz , Ulrich Deickmann, " BioID: A Multimodal Biometric Identification System", IEEE Computer, Vol. 33, No. 2, February 2000.
[5]   Kennedy, J., Eberhart, R. C., and Shi, Y. H., Swarm Intelligence, Morgan Kaufmann Publishers, 2001.
[6]   Eberhart, R. and Kennedy, J., "A New Optimizer Using Particles Swarm Theory", Sixth International Symposium on Micro Machine and Human Science, 1995, Nayoga, Japan.
[7]   Kalyan Veeramachaneni, Lisa Ann Osadciw, and Pramod K Varshney, "Adaptive Multimodal Biometric Fusion Algorithm Using Particle Swarm", SPIE Aerosense, April 21-25, 2003, Orlando, Florida.

Fusion of Auxiliary Information
for Multi-modal Biometrics Authentication

Kar-Ann Toh, Wei-Yun Yau, Eyung Lim, Lawrence Chen, and Chin-Hon Ng

Institute for Infocomm Research
21 Heng Mui Keng Terrace, Singapore 119613
{katoh,wyyau,tpchen,chng}@i2r.a-star.edu.sg
eelim@ntu.edu.sg

Abstract. In this paper, we propose to use the multivariate polynomials for fusion of fingerprint and voice verification systems. On top of using just the fingerprint and voice data for fusion, additional information such as fingerprint image quality and deliberate control inputs are also integrated within the same framework. Numerical experiments using three multivariate polynomial models show significant improvements of matching performance as compared to the optimal weighting method in terms of receiver operating characteristic curves.

1 Introduction

Fusion of biometrics can be traced back to classifiers combination. Two main types of combination can be identified: *classifier selection* and *classifier fusion*. The difference between these two types lies on whether the classifiers are assumed to be complementary or competitive. Classifier selection assumes that each classifier is a "local expert" while classifier fusion assumes that all classifiers are trained over the entire feature space (see e.g. [1]). In this paper, our focus will be on classifier fusion using fingerprint and voice-based biometrics. Our main effort will be on arriving at a fusion technique that optimizes the accuracy of the combined decision.

In general, different classifiers can be combined at one of the following levels according to the information adopted: (i) abstract level, (ii) rank level, and (iii) measurement level. At abstract level, the output information taken from each classifier is only a possible label for each pattern class, whereas at rank level, the output information taken from each classifier is a set of ordered possible labels which is ranked by decreasing confidence measure. At measurement level, the output information taken from each classifier is a set of possible labels with associated confidence measure. We shall work at the measurement level to combine the fingerprint and speaker verification systems. In this way, with the measurement outputs taken from each individual system, the decision is brought forward to the final output of the combined system.

2   Multivariate Polynomial Regression (MP)

The multivariate Optimal Weighting Method (OWM [2] or we called it MP1 here, Multivariate Polynomial with first-order) provides an effective way to linearly combine predictors or classifiers. However, important interacting relationships among the data may be ignored, thereby giving rise to inaccurate results. To cater for possible nonlinear effects and interactions, the multivariate polynomial regression is considered.

A common approach to use the multivariate polynomial model is to progressively add individual polynomial terms, with and without interaction among the input variables, and test for the effect of adding such terms. The *extra sum of squares* provides a measure to gauge the marginal reduction of error sum of squares when one or several predictor variable terms are added to the regression model. For good generalization, both the training and validation test data should be checked for the effects of different combination of terms.

Let $\hat{f}_1...\hat{f}_l$ be the outputs of l single biometrics. Denote $f(\boldsymbol{x}, \boldsymbol{y})$ as the target output of biometric fusion where $\boldsymbol{x}$ and $\boldsymbol{y}$ are the input-output features used in individual biometric processing. Let $\boldsymbol{\alpha}$ be the weight parameters for the polynomial terms. The sum of squared errors chosen for training is thus written as:

$$s_{MP}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\alpha}) = \sum_{i=1}^{m} \left[f(\boldsymbol{x}_i, \boldsymbol{y}_i) - \hat{f}_{MP}(\boldsymbol{x}_i, \boldsymbol{y}_i, \boldsymbol{\alpha}) \right]^2, \tag{1}$$

where

$$\hat{f}_{MP}(\boldsymbol{x}_i, \boldsymbol{y}_i, \boldsymbol{\alpha}) = \alpha_0 + \sum_{j} \alpha_j [\hat{f}_1^{n_1}(\boldsymbol{x}_i, \boldsymbol{y}_i) \hat{f}_2^{n_2}(\boldsymbol{x}_i, \boldsymbol{y}_i) \cdots \hat{f}_l^{n_l}(\boldsymbol{x}_i, \boldsymbol{y}_i)],$$

$$i = 1, ..., m; \; j = 1, 2, ... \tag{2}$$

and the summation is taken over all non-negative integers $n_1, n_2, ..., n_l$ for which $\sum_{i=1}^{l} n_i \leqslant r$ with r being the order of approximation. $\boldsymbol{\alpha}$ here includes the bias term and can be written as $\boldsymbol{\alpha} = [\alpha_0, \alpha_1, \alpha_2, ...]^T$.

Although the accuracy of fit is highly dependent on the degree of freedom of the model which is related to the number of terms in the model, a large number of polynomial terms does not necessarily guarantee a good prediction model since multi-collinearity may become so severe that it corrupts the estimation. Moreover, there exists the problem of over-fitting when there are too many predictor terms.

In biometric applications, for each new registration of a user, the system must learn including this new user. For training based methods, the system have to be retrained including this new user. Hence application of methods such as those using the *extra sum of the squares* would be laborious. To simplify the training process and for practical reasons, we adopt fixed models using low order polynomials. In this application, three multivariate polynomial models of up-to the sixth-order will be investigated since, beyond this point, not only the model

may tend to over-fit the data, but also huge amount of parameters may arise. Also, the constant bias term α_0 is not included since the normalized output from different modalities will be used. To simplify the notations, we omit dependency on the individual biometric feature variables $(\boldsymbol{x}, \boldsymbol{y})$ and the three models are chosen as:

$$
\begin{aligned}
MP1 &: \hat{f}_{MP1}(\boldsymbol{\alpha}) = \alpha_1 \hat{f}_1 + \alpha_2 \hat{f}_2 \\
MP2 &: \hat{f}_{MP2}(\boldsymbol{\alpha}) = \alpha_1 \hat{f}_1 + \alpha_2 \hat{f}_1^2 + \alpha_3 \hat{f}_2 + \alpha_4 \hat{f}_2^2 + \alpha_5 \hat{f}_1 \hat{f}_2 \\
MP3 &: \hat{f}_{MP3}(\boldsymbol{\alpha}) = \alpha_1 \hat{f}_1 + \alpha_2 \hat{f}_1^2 + \alpha_3 \hat{f}_1^3 + \alpha_4 \hat{f}_2 + \alpha_5 \hat{f}_2^2 + \alpha_6 \hat{f}_2^3 + \alpha_7 \hat{f}_1 \hat{f}_2 + \alpha_8 \hat{f}_1 \hat{f}_2^2 \\
&\quad + \alpha_9 \hat{f}_1^2 \hat{f}_2 + \alpha_{10} \hat{f}_1 \hat{f}_2^3 + \alpha_{11} \hat{f}_1^3 \hat{f}_2 + \alpha_{12} \hat{f}_1^2 \hat{f}_2^2 + \alpha_{13} \hat{f}_2^2 \hat{f}_2^3 + \alpha_{14} \hat{f}_1^3 \hat{f}_2^2 + \alpha_{15} \hat{f}_1^3 \hat{f}_2^3
\end{aligned}
\tag{3}
$$

Here, we note that MP1 is actually the optimal weighting method (OWM) that we are interested to compare with. A weight decay regularization was performed when minimizing the least squares objective function.

3   Data Quality and Deliberate Control Inputs

3.1   Fingerprint Image Quality

Besides the inherent degree of freedom from the biometrics adopted, accurate verification depends on good data quality. For example, in fingerprint verification, the image quality affects the accuracy of feature extraction which in turn determines the matching accuracy. In [3], person authentication results from several modalities such as still image and speech are combined using fuzzy k-means, fuzzy vector quantization and median radial basis function network. The reliability of each modality is presented using a *quality* measure which is used in the fuzzification process. It is noted here that the quality measure in [3] refers only to the reliability of authentication results of each modality whereas in our context, the data quality can include an independent measure of input signal condition, not necessarily reliability related. In other words, our quality measure can be taken independently and not related to any authentication means, which is different from that in [3] where the quality measure is derived from the authentication process itself. It is further noted that for such cases of independent measure of input signal condition, additional degrees of freedom deriving from the new measurements are taken into account in the process of multi-modal decision data fusion and this should enhance the fusion performance.

In each biometric, a certain quality measure related to each piece of data collected can be quantified. For fingerprint images, measurements like image contrast and orientation certainty level [4] can be used for such quality measure. Using the OWM or MP model, the quality measures (q_{ij} for i^{th} data using j^{th} biometric) can be incorporated into the minimization objective as:

$$s_{MP1Q}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\theta}) = b\|\boldsymbol{\alpha}\|_2^2 + \sum_{i=1}^{m} \left[f(\boldsymbol{x}_i, \boldsymbol{y}_i) - \sum_{j=1}^{l} (\alpha_j + \beta_j q_{ij}) \hat{f}_j(\boldsymbol{x}_i, \boldsymbol{y}_i) \right]^2$$

$$= b\|\boldsymbol{\alpha}\|_2^2 + \sum_{i=1}^{m} (e_{MP1Q})_i^2 \qquad (4)$$

where $\boldsymbol{\theta}$ denotes the parameter vector $[\boldsymbol{\alpha}^T, \boldsymbol{\beta}^T]^T$.

Let $\mathbf{F}_\alpha$ and $\mathbf{F}_\beta$ be the Jacobian of $[\alpha_j \hat{f}_j(\boldsymbol{x}_i, \boldsymbol{y}_i)]$ and $[\beta_j q_{ij} \hat{f}_j(\boldsymbol{x}_i, \boldsymbol{y}_i)]$ respectively. The first derivative of $s_{MP1Q}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\theta})$ can then be written as

$$\frac{\partial s_{MP1Q}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = -2 \begin{bmatrix} \mathbf{F}_\alpha^T \\ \mathbf{F}_\beta^T \end{bmatrix} e_{MP1Q}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\theta}) = -2\hat{\mathbf{F}}^T e_{MP1Q}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\theta}). \qquad (5)$$

Equating the first derivative to zero and solving it, we have

$$\boldsymbol{\theta} = (\hat{\mathbf{F}}^T \hat{\mathbf{F}} + b\mathbf{I})^{-1} \hat{\mathbf{F}}^T \mathbf{f}, \qquad (6)$$

where

$$\hat{\mathbf{F}}^T = \begin{bmatrix} \mathbf{F}_\alpha^T \\ \mathbf{F}_\beta^T \end{bmatrix}. \qquad (7)$$

For multivariate polynomial models MP2 and MP3, each of $\hat{f}_j(\boldsymbol{x}_i, \boldsymbol{y}_i)$ and $q_{ij} \hat{f}_j(\boldsymbol{x}_i, \boldsymbol{y}_i)$ terms are treated as independent input terms for the multivariate polynomials corresponding to the weighting parameters α_j and β_j respectively. The number of terms for the model will then be increased (if every biometric has a quality measure, then this total number of terms will be doubled) according to the number of quality terms incorporated.

3.2   Partial Information and Deliberate Control

When full information for multi-modalities is available and when redundant information is minimal, data fusion of these multi-modalities should provide better accuracy since a larger degree of freedom is available for the information. Assuming randomness and independency among individual modalities, [5] has shown that it is possible to improve performance by integrating multiple biometrics.

In real-life application, full information for all modalities may not be always available. When combining a strong verification system like fingerprint with a weaker verification system like hand-geometry, it is natural to have a stronger weight assigned to the stronger verification system. If for some reasons, the fingerprint is too dry or too wet to be used for verification, then it would be difficult for this particular user to get through this verification system since the combined verification is heavily dependent on the stronger system of fingerprint. This will be the case that causes high false rejection rate.

If however, for the same system the weights for all modalities are set equal, then the weaker biometric may increase the false acceptance rate of the combined system since for such cases of poor fingerprint images, authentication may come from the weaker biometric.

This problem can be partially handled using the data quality control input described above. For example, when combining fingerprint and voice data, the detected temporary poor fingerprint images due to contamination can always be re-captured. As for the case of permanent data deficiency in one biometric (e.g. due to exceptionally dry fingers), and consider combination of two biometrics, the remaining biometric will then be the only source of information. In this case, multiple entries from the remaining biometric can be called for upon detection of such extreme case.

Another possible solution is by deliberate user control. This means that the user can input certain specific deliberate information for identity verification. In speaker verification, pronouncing a specific code can be incorporated into the combined system whereby a correct matching of such code-word utterance results in an additional control input to the combined system. In this way, an additional control signal is incorporated into the system for a combined decision.

4   Biometrics Fusion: Fingerprint and Speaker Verifications

In this section, we perform experiments on the multivariate polynomial fusion using physical data from two biometrics: fingerprint and voice data. We shall compare the performances of MP2 and MP3 with OWM, which is labelled as MP1, in the form of Receiver Operating Characteristic (ROC) Curves.

4.1   Combining Fingerprint Verification and Speaker Verification

In this experiment, both the databases for fingerprint verification and speaker verification consists of 16 different identities, each of which contributes 6 different fingers or 6 different words with each fingerprint or word containing 10 samples. A total of 960 samples were thus used for each fingerprint and voice verification system. The fingerprint images were collected using Veridicom's *i*Touch sensor and the voice data were taken from TIDIGIT database. An arbitrary one-to-one correspondences was taken between the two biometric databases. Both databases are partitioned into two equal sets for training and testing, i.e. set-1 with 480 ($16 \times 6 \times 5$) samples for training and, set-2 with 480 ($16 \times 6 \times 5$) samples for testing.

4.2   Effect of Model-Order

Fig. 1 (a) and (b) show the training and test results respectively for multi-modal data fusion using MP1, MP2 and MP3. The ROC curves for the original fingerprint and speaker verifications are also included in the same set of plots
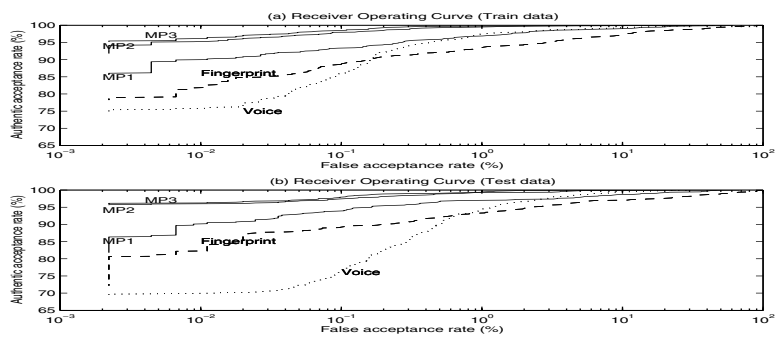
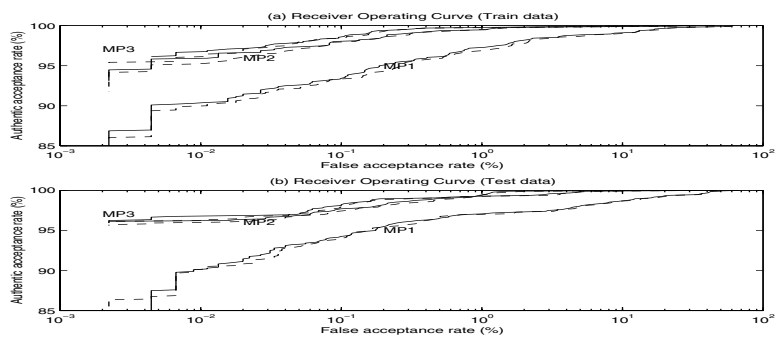**Fig. 1.** ROC curves for MP1, MP2 and MP3

Fig. 2. ROC curves for multi-modal training and test (solid line: with quality input, dashed line: without quality input)

for comparison purpose. It can be seen from these figures that MP1 improves significantly over the entire operating range of individual fingerprint and speaker verifications. As the model order increases using MP2 and MP3, the improvement is even more significant. For MP3, the authentic acceptance rates are found to be over 95%.

4.3   Effect of Quality Measure Input

In this example, we incorporate the fingerprint image quality measure [4] as part of the multi-modal input for data fusion. The results (solid-lines) are compared with those without the quality measure input (dashed-lines) in Fig. 2 (a) and (b), respectively for training data and test data. It can be seen that for both training and test sets, about 1-2 % improvement upon the ROC performance was achieved using the additional image quality input when combining the data.
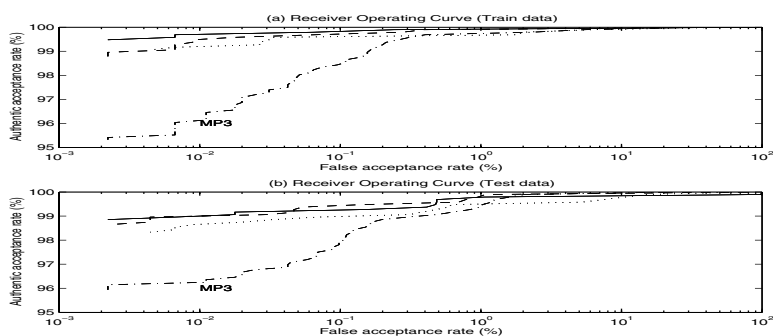
Fig. 3. ROC curves for multi-modal training set (dotted: MP1 with control input, dashed: MP2 with control input, solid: MP3 with control input, dashed-dotted: MP3 without control input)

4.4 Effect of Deliberate User Control Input

The effect of including deliberate user control input is experimented here. To provide a high security, the user is to provide a speech codeword in addition to the biometric inputs. When the speech codeword from a user is matched with that stored in the system, a '1' is generated, otherwise a '0' is generated for non-matched codeword. This codeword activated score is added as an input to the regression model for multi-modal authentication. The results for including such a control input are shown in Fig. 3(a) and (b) respectively for training and test data. For comparison purpose, the plots for MP3 from Fig.1 are also included in the figures. Since the user control provides a definite discrete input into the combined system, the improvement to the accuracy is significant and was seen to be over 99% for the training data and near 99% for the test data. It is noted that this approach to combine the user control input provides an integrated mode of data fusion with matching accuracy taken into account. This means that any user who knows the codeword do not necessarily have full control of access as in the case of using the 'OR' logic for integration. However, as compared to the 'AND' logic for integrating the codeword, this approach of data fusion provides less room for isolating individual modalities in any attempt to use each modality independently for access authentication.

5   Conclusion

This paper proposed to use the multivariate polynomials for biometrics fusion. Main advantage of this approach is that it provides effective nonlinear decision surfaces with low computing effort. The proposal includes the fusion of data quality and deliberate inputs to strengthen the accuracy. Three low-order polynomials were investigated using both artificial data and physical data. The

matching performances in the form of the receiver operating characteristic curves are shown to be superior when quality and control data are incorporated during fusion.

References

[1] L. I. Kuncheva, J. C. Bezdek, and R. Duin, "Decision templates for multiple classifier design: An experimental comparison," *Pattern Recognition*, vol. 34, no. 2, pp. 299–314, 2001. 678

[2] K.-A. Toh and W.-Y. Yau, "Combination of hyperbolic functions for multi-modal biometrics data fusion," *IEEE Trans. Systems, Man and Cybernetics, Part-B*, vol. 34, no. 3, 2004. (to appear). 679

[3] V. Chatzis, A. G. Bors, and I. Pitas, "Multimodal decision-level fusion for person authentication," *IEEE Trans. Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol. 29, no. 6, pp. 674–680, 1999. 680

[4] E. Lim, X. Jiang, and W.-Y. Yau, "Fingerprint quality and validity analysis," in *IEEE International Conference on Image Processing (ICIP'02)*, vol. 1, pp. I:469–I:472, 2002. 680, 683

[5] L. Hong, A. Jain, and S. Pankanti, "Can multibiometrics improve performance?," in *Proc. Auto ID*, (Summit, NJ), pp. 59–64, 1999. 681

[6] K.-A. Toh, W.-Y. Yau, X. Jiang, T.-P. Chen, J. Lu, and E. Lim, "Minutiae data synthesis for fingerprint identification applications," in *Proceedings of the 2001 International Conference on Image Processing (ICIP)*, vol. 3, (Greece), pp. 262–265, October 2001.

Decision Fusion for Face Authentication

Jacek Czyz[1], Mohammad Sadeghi[2], Josef Kittler[2], and Luc Vandendorpe[1]

[1] Communications Laboratory
Université catholique de Louvain, B-1348 Louvain-la-Neuve, Belgium
{czyz,vdd}@tele.ucl.ac.be
[2] Centre for Vision, Speech and Signal Processing
University of Surrey, Guildford, Surrey, GU2 5XH, UK
{m.sadeghi,j.kittler}@surrey.ac.uk

Abstract. In this paper we study two aspects of decision fusion for enhancing face authentication. First, sequential fusion of scores obtained on successive video frames of a user's face is used to reduce the error rate. Secondly, the opinions of several face authentication algorithms are combined so that the combined decision is more accurate than the best algorithm alone. The experiments performed on a realistic database demonstrate that the fully automatic multi-frame – multi-experts system proposed in this work allows a significant improvement over the static – single-expert system.

1 Introduction

Biometrics, which measures a physiological or behavioural characteristic of a person, such as voice, face, fingerprints, iris, etc., provides an effective and inherently reliable way to carry out personal identification. The face modality is very important for real world applications because it is very well accepted by the users. In return, the acquired face images contain lots of variability. The pixel map of facial images varies drastically under variable illumination and 3D pose. Also the localisation and registration of the face sub-image is difficult when the background image is uncontrolled.

Robustness of face-based authentication can be improved by combining or fusing different sources of information related the identity to authenticate. For example one could use several cameras oriented at different angles, or add other type of sensors like a microphone or a fingerprint sensor. In all cases strategies must be devised to combine the information coming from different sources.

In this paper we study two different aspects of decision fusion in the context of fully automatic face authentication. Firstly decision fusion is used combine the outputs of several face authentication algorithms. This type of fusion is referred to as intramodal fusion. Intramodal fusion has been recently studied for different biometric modalities [1, 2]. Secondly we study sequential fusion, that is, the fusion of outputs of a single face authentication algorithm obtained on several video frames. During an access attempt the user is interacting with the authentication system over a certain period of time. Over this period many video

frames are available for identity verification. For both fusion aspects, strategies for conciliating the different decisions are presented. Differences between intramodal fusion and sequential fusion are pointed out. The main contribution of the paper is a fusion architecture which takes into account the distinctive features of the intramodal and sequential fusion. Our experiments on a realistic face database show that the proposed architecture allows a significant improvement over a single frame – single expert approach. The paper is organised as follows. In the next section we present biometric authentication and the decision fusion aspects considered in this work. In Section 3, face authentication algorithms and the experimental setup is described. Experimental results are given and discussed afterwards. In the last section we present our conclusions.

2   Intramodal and Sequential Fusion of Face Authentication Experts

Biometric identity authentication can be stated as follows. When performing verification, a biometric trait $\mathbf{x}$ of the person making the claim is recorded and compared to a template that has been previously recorded. A score s reflecting the quality of the match between the template and the unknown biometric trait is compared to a threshold η to determine whether the claim is genuine (class ω_a) or false (class ω_b), i.e.

$$s(\mathbf{x}) \underset{\omega_b}{\overset{\omega_a}{\lessgtr}} \eta \tag{1}$$

Two types of errors can be distinguished whether a genuine claim is rejected or an impostor claim is labelled as genuine. The former is referred to as False Rejection Rate (FRR) while the latter is referred to as False Acceptance Rate (FAR).

2.1   Intramodal Fusion

In order to increase the verification performance, one may take advantage of multiple authentication algorithms, or experts, that provide their opinions on the same biometric data, and perform *intramodal fusion*. Various levels of combination are possible [3]: fusion at the feature level, fusion at the confidence level (also known as soft fusion) and fusion at the abstract level, where accept/reject decisions are combined (hard fusion). In this work we opt for confidence level fusion, that is, where the scores reported by the experts are combined. We believe that for authentication, confidence level fusion is a good compromise between dimensionality and information loss.

Given a measurement $\mathbf{x}$, each expert i outputs a score $s^{(i)}(\mathbf{x})$ based on the same measurement $\mathbf{x}$. These scores can be concatenated into a score vector $\mathbf{s}$ and a second-level classifier can be trained to learn a decision boundary in the score space. In [1], a non-parametric Parzen estimation technique is used to estimate the joint score density for combining several fingerprint matchers. Here we perform the fusion using a weighted averaging technique and using

a Support Vector Classifier. In the weighted averaging method the decision is based on a new score s_w which is obtained by linear combination of the experts score, i.e. $s_w = \mathbf{w}^T \mathbf{s}$, where the weights $\mathbf{w}$ are obtained by minimising the Equal Error Rate (EER) on a training set. In the second fusion method, an SVC with a linear kernel is trained to separate genuine from impostor score vectors.

2.2   Sequential Fusion

When multiple video frames of the same user's face are available, a score s_i is obtained from each frame i. Let us emphasise the difference with the intramodal case. In the sequential frame combination, all scores are emitted by the same expert, so that they can be seen as multiple random outcomes of the same score distribution (the score distribution depends on the expert). For this reason, the combination should either (i) give the same importance to all scores, i.e. average the scores, or (ii) have a mechanism of for selecting the "best" frame or best score. In case (i) the scores are averaged so that each s_i has the same importance in the decision. The scores s_i are drawn from a random variable S with score probability distributions $p(s|\omega_b)$ and $p(s|\omega_a)$ in case of impostors or clients. It is well known that the sample average $\bar{S} = 1/N \sum_{j=1}^{N} S_j$ of N samples S_j (considered here as random variables) drawn from a given distribution has the same mean than the distribution. Also, if the samples are drawn independently, the variance of $\bar{S}$ is σ^2/N where σ^2 is the variance of the score distribution S. Therefore $p(\bar{s}|\omega_c)$ ($c \in \{a, b\}$) has the same mean than $p(s|\omega_c)$ but a variance divided by N. Because the error rate depends directly on the overlap between the impostor and genuine sample mean densities, if the decision is taken using $\bar{s}$ rather than s, the error rate decreases as N increases. In case (ii), a simple solution for choosing the best frame consists of using a template matching-based method: select the frame that gives the best match with the template in the sense of a distance measure. In the case of a dissimilarity (similarity) score, this results in taking the minimum (maximum) score for making the decision, i.e. $\min(s_1, s_2, ..., s_N) \underset{\omega_b}{\overset{\omega_a}{\lessgtr}} \eta$.

This may favour both genuine accesses and impostor accesses. The merit of this combination rule depends essentially on the score probability function as demonstrated in the simulations below.

Suppose that the impostor and genuine score distributions are Gaussian with equal variance σ^2 but different means. We draw independently N samples s_j from the genuine or the impostor distribution and base our decision on the average of s_j or the minimum s_j. Figure 1(left) shows the classification error versus the number of samples N for sample average based and minimum based decision. Both integration methods improve the decision over the one sample score case, but clearly the average rule outperforms the minimum rule.

Gaussian hypothesis for scores may not be satisfied in practice. In particular the genuine score density is usually asymmetric with a heavy tail or bimodal. The secondary mode is due to users who consistently return large scores (called "goats" in [4]). Another contributing factor to the heavy tail of the genuine density comes from failures during pre-processing. A more realistic choice is therefore
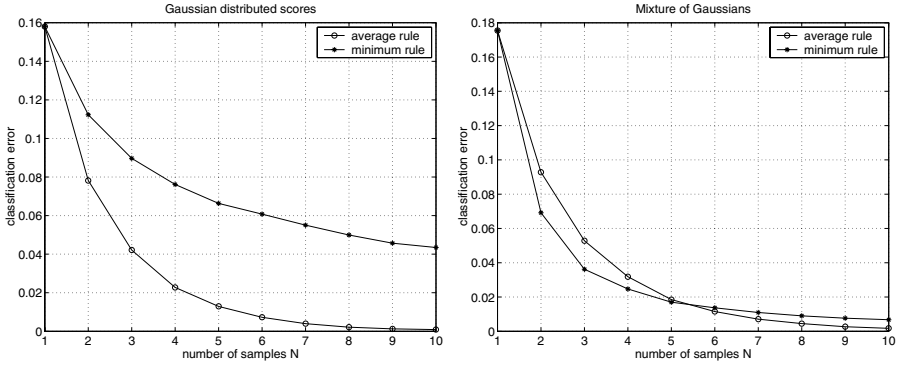
Fig. 1. Classification error versus the number of samples N for Gaussian distributed scores (left) and Mixture of Gaussian score (right)

to represent the genuine score density by a bimodal mixture of Gaussians. In this case, the error rates are obtained through simulations and results are presented in Figure 1(right). Note that the "sample average" curve is quite similar to the pure Gaussian case. In contrast, the secondary mode changes drastically the "sample minimum" curve, which now outperforms the average rule for the first five frames that are combined.

From the simulations, it appears that the average rule is advantageous in the Gaussian case, while the minimum rule starts to outperform the average rule when the genuine density has a heavy tail.

2.3   Proposed Fusion Architecture

From the discussion above, we propose the following fusion architecture. For each expert i, the multiple scores $s_j^{(i)}$ $j = 1, 2, ..., N$ corresponding to multiple frames are first fused using either the average or the minimum rule (depending on the score distribution). The R resulting scores $s^{(i)}$ $i = 1, 2, ..., R$ are then fused using a second level classifier. The final decision is based on the output of the second level classifier. The experiments presented below show that this architecture allows a significant improvement over a single frame – single expert approach.

3   Experiments

3.1   Face Authentication Experts

Once the face and eyes are located, the face is registered and histogram equalised. The normalised face image is then used to generate the accept/reject decision. In the results presented, we have used two different face verification algorithms,

namely a Linear Discriminant Analysis (LDA) based and an SVM based algorithm. Both methods are described in [5], we give hereafter a short description.

The LDA approach is used to extract features from the gray level face image. LDA effectively projects the face vector into a subspace where within-class variations are minimised while between-class variations are maximised. The LDA score $s^{(1)}$ is computed by matching the newly acquired LDA face projection to the user template using normalised correlation. In the SVM-based method, to label the face vector $\mathbf{x}$ as genuine or impostor, the classifier evaluates the quantity $s^{(2)} = \sum_{i=1}^{l} y_i \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b$, where $\mathbf{x}_i$ is the input vector of the ith training example, l is the number of training examples, the α_i and b are the parameters of the model, and $K(\mathbf{x}, \mathbf{x}_i)$ is the kernel function.

To locate automatically the face in the image, two different face localisation methods have been used. In the first method, the whole image is exhaustively scanned at different scales using a small window. The content of each window is classified by a SVM classifier into face or non-face classes. See [5] for more details. The expert using LDA face verification and the SVM-based face localisation is referred to as **LDA1**. The expert called **SVM** is using the SVM-based verification and localisation. In the second face localisation method, Gabor filters are used to detect facial features such as corners of eyes, nostrils, etc. in the input image. Feature configurations that correspond possibly to a face sub-image are transformed into a normalised face space where classification in face/non-face classes is performed. More details can be found in [6]. The expert using LDA face verification and the face localisation method just described is referred to as **LDA2**.

3.2   Database and Experimental Results

The experiments presented in this section were performed on the English part of the BANCA database [7]. The data set contains video recordings of 52 people in several environmental conditions. Each subject recorded 12 sessions distributed over several months, each of these sessions containing 2 records: one true user access and one impostor attack. The 12 sessions were separated into 3 different scenarios: controlled, degraded and adverse. A low-cost camera has been used to record sessions in the degraded scenario. For this scenario, the background and the lighting were uncontrolled, simulating a user authenticating himself in an office or at home using a low cost web-cam. A more expensive camera was used for the controlled and adverse scenarios. The adverse scenario simulates a cash withdrawal machine, and was recorded outdoors. From one video session (about 30 seconds), five frames per person were randomly selected for face verification. In the experiments presented, two protocols are considered. The first protocol, referred to as protocol G in [7], uses the first session of the 3 scenarios to enrol a new user, that is, to create its user template. The second protocol (protocol P) uses session 1 only to enrol a new user.

Figure 2 shows the average Half Total Error Rate HTER = (FAR + FRR)/2 obtained on the BANCA database using the minimum and the average rules
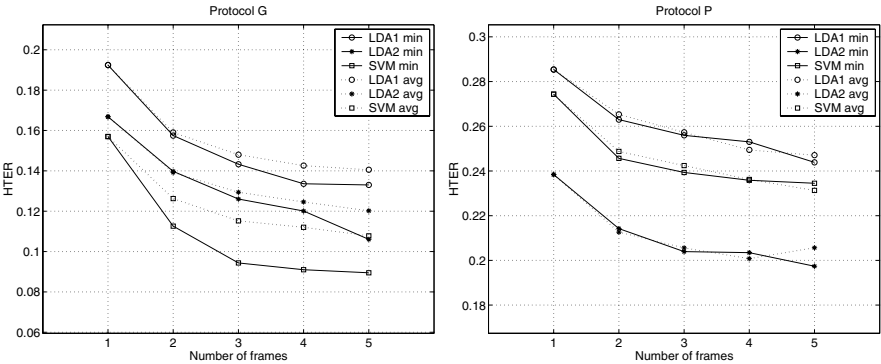
Fig. 2. Sequential fusion results using minimum (min) and average (avg) rules obtained on the BANCA database for protocol G (left) and protocol P(right)

above for protocol G (left) and protocol P (right). For the three experts considered, we evaluate the HTER as function of the number of frames fused. Starting with one frame, we add successively a frame to the set and base the decision on the set of frames. It appears that, in the case of protocol G, the minimum rule gives the best improvement, up to several percents, over the single frame technique. The average rule gives a slightly weaker improvement. In the P protocol case (Figure 2(right)), the HTER is much higher than for the G protocol because only one session is available for training. Again, with 5 frames, the HTER is significantly smaller with respect to a single frame based decision. Note that this time the two rules perform approximatively the same. As expected, when the number of test frames increases, the HTER decreases, but the improvement seems to saturate quickly. It is likely that no further improvement could be obtained with a larger number of frames. Table 1 summarises the HTER obtained in this multi-frame – single expert case.

Following the fusion architecture described in Section 2.3, the scores obtained after sequential fusion can be combined using the second level classifier. These intramodal fusion results are reported in Table 2. Note that the minimum rule has been used to fuse the sequential scores. From the table it appears that in all cases, the intramodal fusion further decreases the HTER over the multi-frame – single expert results. In particular the fusion of experts SVM and LDA2 leads to

Table 1. Multi-frame - single expert performance

Protocol	Experts		
	LDA1	LDA2	SVM
G	13.30	10.68	8.95
P	24.39	19.74	23.45

Table 2. HTER obtained on the BANCA database using intramodal fusion for protocol P and G

Protocol	Fusion techn.	Combined Experts		
		LDA1 & LDA2	LDA2 & SVM	LDA1 & SVM
G	w.avrg	9.53	6.27	8.44
G	SVC	9.34	5.58	7.32
P	w.avrg	19.60	18.38	20.83
P	SVC	18.43	17.65	20.06

an error rate of 5.58% in protocol G and 17.65% in protocol P, using the SVC-based fusion. For comparison the best result in the single frame – single expert case is 15.70% in protocol G and 23.84% in protocol P. Note that the two fusion techniques allow approximatively the same improvement, with a slight advantage for SVC. Interestingly, the fusion of experts LDA1 and LDA2, improves the HTER although they differ only by the face localisation procedure.

4   Conclusion

We discussed how decision fusion can be used to improve the performance of automatic face authentication. Intramodal and sequential fusion are used at two different stages in the authentication process. For the sequential fusion, two combination rules are presented and it is shown that the minimum rule is advantageous over that average rule when the genuine score density has a heavy tail. Both rules allow a significant improvement over the single frame system. For the intramodal fusion, the very simple weighted averaging and the more complex Support Vector Classifier have shown to perform similarly on the BANCA face database. Experiments show that the error rates are substantially improved thanks to the multi-frame – multi-expert architecture, which gives practical relevance to the proposed approach. Recently Zhou and Chellappa proposed to use recursive Bayesian filtering for face authentication or recognition in video [8]. A performance comparison between this approach and the sequential fusion presented in this paper could be an interesting future study.

References

1. Prabhakar, S., Jain, A.K.: Decision-level fusion in fingerprint verification. Pattern Recognition **35** (2002) 861–874 686, 687
2. Czyz, J., Kittler, J., Vandendorpe, L.: Combining face verification experts. In: Proc. of Int. Conf. on Pattern Recognition, Quebec, Canada. (2002) 686
3. Ross, A., Jain, A.K., Qian, J.Z.: Information fusion in Biometrics. In: Proc. Int. Conf. on Audio- and Video-based Person Authentication. (2001) 355–359 687
4. Doddington, G., Liggett, W., Martin, A., Przybocki, M., Reynolds., D.: 'sheep, goats, lambs and wolves': a statistical analysis of speaker performance in the NIST

speaker recognition evaluation. In: Int. Conf. on Spoken Language Processing. (1998) 688

[5] Sadeghi, M., Kittler, J., Kostin, A., Messer, K.: A comparative study of automatic face verification algorithms on the BANCA database. In: Proc. of the Int. Conf. on Audio- and Video-based Biometric Person Auth. (2003) 690

[6] Hamouz, M., Kittler, J., Kamarainen, J., Kalviainen, H.: Hypotheses-driven affine invariant localisation of faces in verification systems. In: Proc. of the Int. Conf. on Audio- and Video-based Biometric Person Auth. (2003) 690

[7] Bailly-Baillière, E., Bengio, S., Bimbot, F., Hamouz, M., Kittler, J., Mariéthoz, J., Matas, J., Messer, K., Popovici, V., Porée, F., Ruiz, B., Thiran, J.P.: The BANCA database and evaluation protocol. In: Int. Conf. on Audio- and Video-Based Biometric Person Auth.,, Springer-Verlag (2003) 690

[8] Zhou, S., Chellappa, R.: Probabilistic human recognition from video. In: Proc. of Int. Conf. on Automatic Face and Gesture Recognition. (2002) 692

User Authentication through Typing Biometrics Features

Lívia C. F. Araújo, Luiz H. R. Sucupira Jr., Miguel G. Lizárraga,
Lee L. Ling, and João B. T. Yabu-uti

School of Electrical and Computer Engineering, State University of Campinas
Albert Einstein Avenue, 400, PO Box 6101, Postal Code 13083-970, Campinas, SP, Brazil
{liviacri,luigijr,lizarrag,lee,yabuuti}@decom.fee.unicamp.br

Abstract. This paper uses a static typing biometrics in user authentication. The inputs are the key down and up times and the key ASCII codes captured while the user is typing a string. Four features (key code, two keystroke latencies and key duration) were analyzed, and, seven experiments were performed combining these features. The results of the experiments were evaluated involving three types of user: the legitimate, the impostor and the observer impostor users. The best results were achieved utilizing all features, obtaining a false rejection rate (FRR) of 1.45% and a false acceptance rate (FAR) of 1.89%. This approach can be used to improve the usual login-password authentication when the password is no more a secret. This paper innovates using the combination of four features to authenticate users.

1 Introduction

The control access is a very important issue in the computer systems. The login-password authentication is the most usual mechanism used to grant access. This authentication is fragile when there is a careless user and/or a weak password, however, it is low-cost and familiar to the users. The purpose of this paper is to improve the login-password authentication using biometric characteristics. Biometric characteristics are defined as behavioral or physiological characteristics that distinguish one person from another [1]. The inclusion of biometric characteristics in automated systems for personal recognition increase their trustworthiness degree. This increase occurs because of biometric characteristics are unique to each person and could not be stolen, lost or forgotten.

The biometric technology employed in this paper is the typing biometrics, also known as keystroke dynamics. The typing biometrics is the process of analyzing the way a user types at a terminal by monitoring the keyboard inputs in attempt to identify users. The typing biometrics authentication can be classified as static or continuous. The static approach analyses the inputs just in a particular moment, and the continuous one analyses the inputs during all user's session [2].

The methodology of this paper is low-cost and unintrusive, and, uses a verification authentication in a static approach.

This paper is organized as follow: In section 2, there is a resume of related studies published; in section 3, the methodology is explained; in section 4, the results are presented; and finally, in the section 5, there are the conclusions and the future works.

2   Related Work

Some researches were published [2-12] in the authentication via typing biometrics since 1990. Some aspects presented in these works are resumed below:

- *Target String*: It is the input string that will be typed. In [8], four strings were used as target. Two target strings were analyzed in [9]: a 31-char string and a login. In [7], target strings were divided in three levels based on a difficulty degree. String length is a very important issue, considering that in [11] was stated that misclassification increases when the string contains fewer than ten characters;
- *Amount of Samples (Training Set)*: Samples are collected during the enrollment process to compound the training set. Its amount varies a lot, it was 3 samples per users in [6] and, 30 samples per user in [9]. In [2], it was concluded that fewer than six samples is not recommended to obtain good results in performance;
- *Features*: A good feature has to be highly repeatable in the same user and different between users [3]. Two most used features is *duration of the key*, that is the time interval that a key remain pressed [6], and *keystroke latency*, that is the time interval between successive keystrokes [6]. In [5] and [12] the combination of these features brought better results than using them isolated;
- *Timing Accuracy*: As the most of the typing biometrics features are time-based, the precision of the key up and the key down times have to be analyzed. The timing accuracy in the researches varies between 0.1 millisecond [5] and 1 second) [7].
- *Adaptation Mechanism*: Biometric characteristics change over time. An adaptation or a re-enrollment could be performed to maintain the templates updated. The majority of the researches did not mention this issue, but, in [3], an adaptation mechanism was activated every time a successful authentication was performed, creating a new template updated with the new sample and without the oldest one.
- *Classifier*: In [2, 5, 7-9] and [11], a statistical classifier was applied, using known techniques as k-means, Bayes, etc. In [4] and [12], fuzzy logic was applied. In [10], a statistical, a neural, and a fuzzy classifier were combined.

3   Methodology

Each time a user tries to access a system, he indicates an account *a*, and, types the target string. This string is chosen by the user and its length has to be of at least 10 characters. While the user is typing, *keystroke data* is captured in a 1 millisecond timing accuracy and, a *sample* is created containing the *features* calculated using this data. If the account is new, then 10 (ten) samples will compound the training set, and, a *template* is created. A sample will only be stored if the *key code* feature match. If the account already exists, then a sample will be compared with the account's template. According to the *classifier decision*, the authentication will be successful or not. In an authentication session a user has two trials to be authenticated. If the user's identity is validated, then he could access the system and an *adaptation* mechanism could be called to compute a new template update.

The main issues related to the methodology are described in the next sub-sections.

3.1 Keystroke Data

A string with m characters will result in n keystrokes, where $m \leq n$, since some characters need more than one keystroke (e.g. "X" needs "x" and "$shift$" keys). $K_{a,w} = \{ k_1(a,w), ..., k_n(a,w) \}$ is the keystroke data captured in the sample w in the account a.

Each keystroke $k_i(a,w)$ is compound of the key down time $t_{i.down}(a,w)$ (the instant when the key is pressed), the key up time $t_{i.up}(a,w)$ (the instant when the key is released) and the key code $c_i(a,w)$ (the ASCII code).

3.2 Features

Features are calculated using keystroke data. Four features were analyzed in this paper:

- Key Code: the ASCII code that represents each key. When a string contains capital letters, there are more than one possible set of key codes, otherwise there is a single possible set of key codes. The key codes contained in the template of the account a are represented as $C_a = \{c_1(a), ..., c_n(a)\}$ and, the key codes contained in the sample w in the account a is represented as $C_{a,w} = \{c_1(a,w), ..., c_n(a,w)\}$.
- Down-Down Time: a keystroke latency defined as the time interval between successive keystrokes [7]. This feature is represented as $DD_{a,w} = \{dd_1(a,w), ..., dd_n(a,w)\}$ where $dd_i(a,w) = t_{i+1.down}(a,w) - t_{i.down}(a,w)$ is related to (k_i, k_{i+1}).
- Up-Down Time: a keystroke latency feature, and, is represented as $UD_{a,w} = \{ud_1(a,w), ..., ud_{n-1}(a,w)\}$ where $ud_i(a,w) = t_{i+1.down}(a,w) - t_{i.up}(a,w)$ is related to (k_i, k_{i+1}).
- Down-Up Time: the duration of key that is defined as the time interval that a key remain pressed [7]. This feature is represented as $DU_{a,w} = \{du_1(a,w), ..., du_n(a,w)\}$ where $du_i(a,w) = t_{i.up}(a,w) - t_{i.down}(a,w)$ is related to k_i.

The combination of these features is novel, previous researches used at most two features.

3.3 Template

The template contains the key code itself C_a, and, the mean $\mu(feat_i(a))$ and standard deviation $\sigma(feat_i(a))$ of each element i of each feature $feat$ (DD, DU or UD).

3.4 Classifier

The sample w of the account a is analyzed by the classifier. $C_{a,w}$ is compared with C_a: if they are different, the authentication fails, otherwise for each feature $feat$, a distance between the template and the sample is calculated by (1):

$$D_{feat}(a,w) = \frac{1}{n} \sum_{i=1}^{n} d_i(a,w) \cdot \qquad (1)$$

where n is the total of elements of the feature $feat$ and $d_i(a,w)$ is the distance related to each element i between the template and the sample, and, is given by (2):

$$d_i(a,w) = \frac{feat_i(a,w) - \mu(feat_i(a))}{\sigma(feat_i(a))} \qquad (2)$$

Finally, the authentication will be successful if the condition (3) is satisfied:

$$(D_{dd}(a,w) \leq T_{dd}(a)) \text{ and } (D_{du}(a,w) \leq T_{du}(a)) \text{ and } (D_{ud}(a,w) \leq T_{ud}(a)) \tag{3}$$

where $T_{dd}(a)$, $T_{du}(a)$ and $T_{ud}(a)$ are the thresholds for the down-down, down-up, and up-down, respectively in the account a.

An analysis was performed in real data and, it was observed that a feature with a higher variation demands a lower threshold, meanwhile a feature with a lower variation demands a higher threshold. So, the threshold for each feature in each account is obtained based on its standard deviation.

3.5 Adaptation Mechanism

The adaptation mechanism consists of creating an updated template, including a new sample and discarding the oldest one. This mechanism is performed with a successful sample w if the majority elements i of the time-features *feat* satisfy the condition (4).

$$\left(d_{feat_i}(a,w) \leq T_{feat}(a) \right) \tag{4}$$

As the adaptation mechanism is activated, the standard deviation for each feature is modified and also the thresholds are modified.

4   Experiments

The experiments were conducted in three machines (a laptop and two PCs) with two different keyboard layouts, and, 30 (thirty) users were enrolled and participated of the experiments in three situations of authentication:

− Legitimate user: the users tried to be authenticated in their accounts.
− Impostor user: the users tried to be authenticated in other user's accounts, knowing the string typed by their owners.
− Observer impostor user: the users observed how the other user types their strings, then they tried to be authenticated in their accounts.

Seven experiments were analyzed, combining the features: (I) *DD*; (II) *UD*; (III) *DU*; (IV) *DD* and *UD*; (V) *DD* and *DU*; (VI) *UD* and *DU*; (VII) *DD, UD and DU.*

4.1 Results

The performance of biometrics systems are generally measured by two rates [1]:

− False Acceptance Rate (FAR): the probability that the system will fail to reject an impostor user.
− False Rejection Rate (FRR): the probability that the system will fail to verify the legitimate user claimed identity.
− Table 1 shows the FRR for legitimate users and FAR for impostor and observer impostor users.

Table 1. Experiments comparative results for legitimate, impostor and observer impostor users

Experiment		(I)	(II)	(III)	(IV)	(V)	(VI)	(VII)
Legitimate	Sessions	553	553	553	553	553	553	553
	Errors	9	12	13	12	7	9	8
	FRR	1.63	2.17	2.35	2.17	1.27	1.63	1.45
Impostor	Sessions	2916	2916	2916	2916	2916	2916	2916
	Errors	580	179	795	151	163	91	55
	FAR	19.90	6.14	27.26	5.18	5.59	3.12	1.89
Observer Impostor	Sessions	492	492	492	492	492	492	492
	Errors	144	58	166	46	48	34	18
	FAR	29.27	11.79	33.74	9.35	9.75	6.91	3.66

Table 2. Results achieved in a imposed string and in a one-trial authentication

Experiment (VII)	Imposed String	One-Trial
Sessions	533	553
Errors	92	64
FRR	17.26%	11.57%

Table 3. Results achieved without adaptation

Experiment (VII)	Legitimate User	Impostor User
Sessions	553	2916
Errors	23	137
Rates	4.16% FRR	4.70% FAR

As noted in table 1, the best results were achieved in the experiment (VII). Some particular aspects in the methodology were analyzed in this experiment to verify their impacts in the results:

− Key Code Feature: Its impact in the results was really noticed in target strings that contain capital letters (70% of impostor's sessions were detected). Thus, the choice of a target string with capital letters, which combines shift and Caps Lock keys, increases the difficulty of authentication of an impostor user.
− Familiarity of the Target String: The second column of the table 2 shows the results achieved using an imposed string ("unicamp@003"). The FRR increased, showing that the familiarity of the target string to the user has a significant impact.
− One-Trial Authentication: The last column of table 2 shows the results achieved conferring just one trial. The FRR increased, showing that a user must have two trials to be authenticated.
− Adaptation Mechanism: Table 3 shows the results without this mechanism. Both rates ARE increased, showing that an adaptation mechanism must be performed.
− Timing Accuracy: Table 4 shows the results in a lower timing accuracy of 10 milliseconds Both rates are increased, showing that a 1 millisecond timing should be applied.

− Amount of Samples: Figure 1 shows the behavior of FRR and FAR with the amount of samples used in enrollment. With the reduction of the amount of samples, FRR increases and FAR varies a little.

The rates obtained are both competitive with the published work as observed in Table 5. In this table there is a resume of some researches in keystroke dynamics including a previous one based on fuzzy logic conducted by us [12].

5   Conclusion

This paper presents a methodology through typing biometrics features to improve the usual login-password authentication. Some experiments were conducted and the best performance was achieved using a statistical classifier based on distance and the combination of four features (key code, down-down, up-down and down-up times), obtaining a 1.45% FRR and a 1.89% FAR. These rates are both competitive with other researches. This work innovates using four features to authenticate users.

For future work, we intend to increase the database and to extend the methodology to numeric keyboard of cell phones and the ones used to access restricted area.

Table 4. Results achieved in a lower timing accuracy

Experiment (VII)	Legitimate User	Impostor User
Sessions	553	2916
Errors	9	110
Rates	1.63% FRR	3.77% FAR
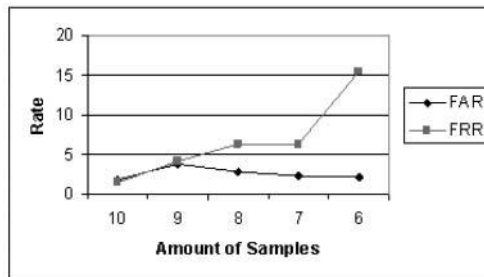
Fig. 1. The variation of the rates with the amount of samples

Table 5. A resume of some researches and ours

Research	Samples	Target String	FRR	FAR
de Ru and Elof [4]	Varies	Two	7.4%	2.8%
Joyce and Gupta [8]	Eight	Four	13.3%	0.17%
Haidar et al. [10]	Fifteen	One	2.0%	6.0%
Araújo et al. [12]	Eight	One	3.5%	2.9%
Our Research	Ten	One	1.45%	1.89%

Acknowledgement

References

[1]   Matyas Jr, S.M., Stapleton, J.: A Biometric Standard for Information Management and Security. Computers & Security, Vol. 19 (2000) 428-441

[2]   Monrose, F., Rubin, A.D.: Keystroke Dynamics as a Biometric for Authentication. Future Generation Computer Systems, Vol. 16, No. 4 (2000) 351-359

[3]   Monrose, F., Reiter, M.K., Wetzel, S.: Password Hardening Based on Keystroke Dynamics. 6th ACM Conference on Computer Security (1999)

[4]   de Ru, W.G., Eloff, J.H.P.: Enhanced Password Authentication through Fuzzy Logic. IEEE Expert / Intelligent Systems & Their Applications, Vol. 17, No. 6 (1997) 38-45

[5]   Robinson, J.A., Liang, V.M., Michael, J.A., MacKenzie, C.L.: Computer User Verification Login String Keystroke Dynamics. IEEE Trans. Syst., Man, Cybern., Vol. 28, No. 2 (1998) 236-241

[6]   Lin, D.T.: Computer-Access Authentication with Neural Network Based Keystroke Identity Verification. International Conference on Neural Networks, Vol. 1 (1997) 174-178

[7]   Coltell, O., Badfa, J.M., Torres, G.: Biometric Identification System Based in Keyboard Filtering. Proceedings IEE 33rd Annual 1990 International Carnahan Conference on Security Technology, (1999) 203-209

[8]   Joyce, R., Gupta, G.: Identity Authentication Based on Keystroke Latencies. Communication of ACM, Vol. 33, No. 2 (1990) 168-176

[9]   Bleha, S., Slivinsky, C., Hussain, B.: Computer-Access Security Systems Using Keystroke Dynamics. IEEE Trans. Pattern Anal. Machine Intell., Vol. 12, No. 12 (1990) 1217-1222

[10]   Haidar, S., Abbas, A., Zaidi, A.K.: A Multi-Technique Approach for User Identification through Keystroke Dynamics. IEEE Int. Conference of Syst., Man and Cybern., Vol. 2 (2000) 1336-1341

[11]   Bleha, D., Obaidat, M.: Dimensionality Reduction and Feature Extraction Applications in Identifying Computer Users. IEEE Trans. Syst., Man, Cybern., Vol. 21 (1991) 452-456

[12]   Araújo, L.C.F., Sucupira Jr., L.H.R., Lizárraga, M.G., Ling, L.L., Yabu-uti, J.B.T.: A fuzzy Logic Approach in Typing Biometrics User Authentication. Proc. First Indian International Conference on Artificial Intelligence, (2003) 1038-1051

When Faces Are Combined with Palmprints: A Novel Biometric Fusion Strategy

Guiyu Feng[1], Kaifeng Dong[1], Dewen Hu[1,2], and David Zhang[2]

[1] Department of Automatic Control, College of Mechatronics and Automation
National University of Defense Technology, Changsha, 410073, China
smartfgy@yahoo.com.cn
dongkaifeng@hotmail.com
dwhu@nudt.edu.cn
[2] Department of Computing, Biometrics Technology Center
The Hong Kong Polytechnic University, Kowloon, Hong Kong
csdzhang@comp.polyu.edu.hk

Abstract. This paper presents a novel fusion strategy for personal identification using face and palmprint biometrics. In the context of biometrics, three levels of information fusion schemes have been suggested: feature extraction level, matching score level and decision level. This work considers the first level fusion scheme. The purpose of our paper is to investigate whether the integration of face and palmprint biometrics can achieve higher performance that may not be possible using a single biometric indicator alone. Both Principal Component Analysis (PCA) and Independent Component Analysis (ICA) are considered in this feature vector fusion context. We compare the results of the combined biometrics with the results of the individual face and palmprint. It is found that the performance is significantly improved in both cases, especially in the case of feature fusion using ICA obtaining encouraging results with a 99.17% recognition accuracy rate using a test set sized of 40 people.

1 Introduction

Biometrics deals with automatically identifying individuals based on their distinctive physiological or behavioral characteristics. It is widely acknowledged that only biometric identifiers come close to actually identifying the persons rather than their possessions or their exclusive knowledge [1]. Unlike the possession-based and knowledge-based personal identification schemes, the biometric identifiers cannot be misplaced, forgotten, guessed, or easily forged. Despite these inherent advantages, the wide scale deployment of biometrics-based personal identification has been hindered due to several reasons: Firstly, the less than desirable accuracy in several application domains, for example, face recognition. The accuracy of face recognition is affected by illumination, pose and facial expression [2]. Secondly, the biometric system cannot eliminate spoof attacks. Thirdly, some persons cannot provide the required standalone biometric, owing to illness or disabilities, etc.

Some of these problems can be solved, or at least partly solved, by fusing several biometric channels. There has been a substantial amount of work reported on multi-biometric systems in the literature. Ross and Jain [3] gave a framework of information fusion in biometrics and presented experimental results combining face, fingerprint, and hand geometry biometrics at the matching score level. Hong and Jain [4] integrated face and fingerprint images for personal identification. Wang et al. [5] combined face and iris biometrics for personal verification. Kumar et al. [6] proposed a new approach for the personal verification using palmprint and hand geometry biometric. Chang et al. [7] introduced comparison and combination results of ear and face images in appearance-based biometrics.

The main purpose of our paper is to consider a novel multiple biometrics strategy using face and palmprint for personal identification thst is different from [15], in which the authors integrated the two biometrics at decision level using neural network to deal with user authentication. Face recognition is friendly and non-invasive, but the reliability of personal identification using the face is currently low as researchers today continue to grapple with the problem of pose, lighting, orientation and posture [2]. Palmprint recognition is a relatively new branch of biometric technology, performance results comparable to fingerprint recognition have been reported [8],but the interested person must be cooperative and a specific acquisition device is needed. In this paper, the performances of the biometrics combined at the feature level and those of the individual face and palmprint classifiers are compared and analyzed. It is found that the performance is significantly improved in both cases, especially in the case of feature level fusion using ICA. Encouraging results with a 99.17% recognition accuracy rate using a test set sized of 40 people are obtained. Our results also strongly support the conclusion that a multimodal biometric using both palmprint and face outperforms a single biometric.

The rest of the paper is organized as follows: Section 2 gives the detail of the feature extraction using PCA and ICA, which are used in both face feature extraction and palmprint feature extraction cases. Section 3 presents the description of the feature extraction level fusion using PCA and ICA. In section 4 we describe the database and experiments carried out and analyze the results. Section 5 presents some conclusions and future work.

2   Feature Extraction Using PCA and ICA

2.1   Feature Extraction Using PCA [9] [14]

Let the training samples of the biometric images be $x_1, x_2, \ldots, x_M$, where M is the number of images in the training set. The average image of the training set is defined by $\mu = \frac{1}{M} \sum_{i=1}^{M} x_i$. The difference between each image and the average image is given by $\varphi_i = x_i - \mu$. Then, we can obtain the covariance matrix of $\{x_i\}$ as follows:

$$C = \frac{1}{M} \sum_{i=1}^{M} (x_i - \mu)(x_i - \mu)^T = \frac{1}{M} X X^T \tag{1}$$
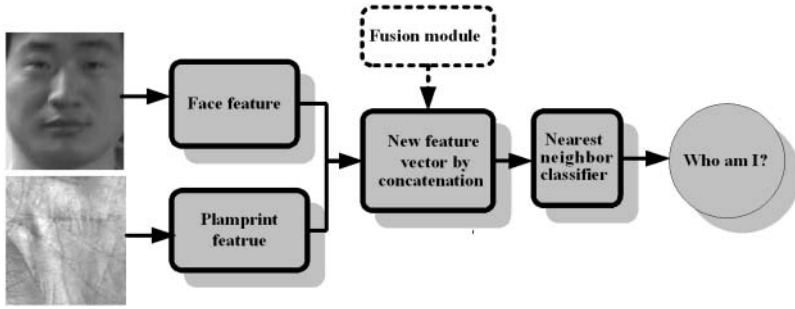
Fig. 1. Bimodal biometric fusion system at the feature extraction level

where the matrix $X = [\varphi_1, \varphi_2, \ldots, \varphi_M]$. It is evident that the matrix C can span an algebraic eigenspace and provide an optimal approximation for those training samples in terms of the mean-square error. It is well known that the following formula is satisfied for the matrix C:

$$C\mu_k = \lambda_k \mu_k (k = 1, 2, ..., M) \tag{2}$$

where μ_k refers to the eigenvector of matrix C, and λ_K is the correlative eigenvalue of matrix C. The theory of PCA states that it is not necessary to choose all of the eigenvectors as the base vectors. Those eigenvectors which correspond to the largest eigenvalues can represent the characteristic of the training set quite well. The significant eigenvectors μ_k with the largest associated eigenvalues are then selected to be the component of the eigenfaces or eigenpalms ($U = \{\mu_k, k = 1, 2, \ldots, M'\}$), which can span M' dimensional subspace of all possible images. A new image is transformed into its eigenspace by the following operation:

$$p_i = U^T (x_i - \mu)(i = 1, 2, ..., M) \tag{3}$$

2.2 Feature Extraction Using ICA

ICA was originally proposed to solve the blind source separation problem of recovering independent source signals S(e.g., different voice, music, or noise sources) after they are linearly mixed by an unknown matrix, $X = AS$.

In the context of biometrics, the training images are the variables, X. Let x_i' be a biometric image, we can construct a training images set $\{x_1', x_2', \cdots, x_m'\}$ with m random variables which are assumed to be linear combination of n unknown independent components (ICs), denoted by $s_1', s_2', \cdots, s_n'$. For all i, $i=1,2\ldots$ m, the image x_i' and the independent component s_i' are converted into vector x_i and s_i by concatenation and denoted as $X = (x_1, x_2, ..., x_m)^T$ and $S = (s_1, s_2, ...s_n)^T$, respectively. Note that the relation between X and S can be modeled as $X = AS$. Then $S = (s_1, s_2, ...s_n)^T$ construct a linear non-orthogonal co-ordinate system for every palmprint image. The rows of A represent features,
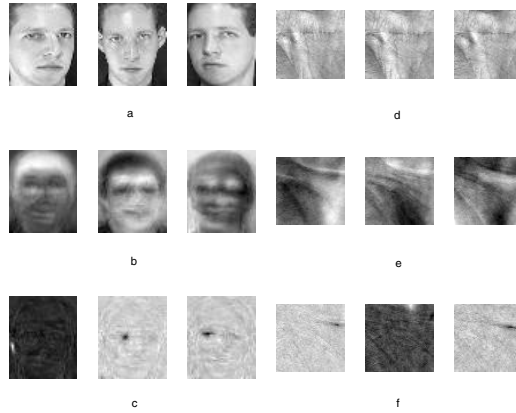
Fig. 2. Three face images of the first person (a), three eigenfaces of face images (b), three ICs of face images (c), three corresponding palmprint images (d),three eigenpalms of palmprint images (e), and three ICs of palmprint images (f)

and s_i is the coefficient of the i-th feature in an observed data vector X. Here we use the same method as [12].

Unlike PCA, ICA does not provide an intrinsic order for the ICs. We adopt the method of minimization the ratio $r = \sigma_{\text{between}}/\sigma_{\text{within}}$ of between-class to within-class variance [13] to obtain the order of ICs, where $\sigma_{\text{between}} = \sum_j (\bar{x}_j - \bar{x})^2$ is the variance of the j class means and $\sigma_{\text{within}} = \sum_j \sum_i (x_{ij} - \bar{x})^2$ is the sum of the variances within each class.

3 Feature Extraction Level Fusion Using PCA, ICA

Among various face recognition algorithms, appearance-based approaches [9] [10] [11] [12] [13] are the most popular. Palmprint recognition is a relatively new branch of biometric technology, the popular approaches are based on algebraic features, for example [14] [16].

For details of fusion at the feature level, see [3]. Fig. 1. shows our bimodal biometric fusion system at the feature extraction level. Here we explore firstly PCA and then ICA to accomplish the fusion task at the feature extraction level. Let $a_i = \{a_{i1}, a_{i2}, ...a_{iK}\}$ be the i-th person's face feature extracted by PCA, $p_i = \{p_{i1}, p_{i2}, ...p_{iL}\}$ be the i-th person's palmprint feature extracted by PCA. Now, we concatenate the face feature vector and palmprint feature vector to form a combined feature vector $f_i = \{a_{i1}, a_{i2}, ...a_{iK}, p_{i1}, p_{i2}, ...p_{iL}\}$ and then the personal recognition is done in the $(K + L)$ dimensional feature space with nearest neighbor classifiers (in L2 norm sense).

This can be easily generalized to the feature level fusion using the ICA case.
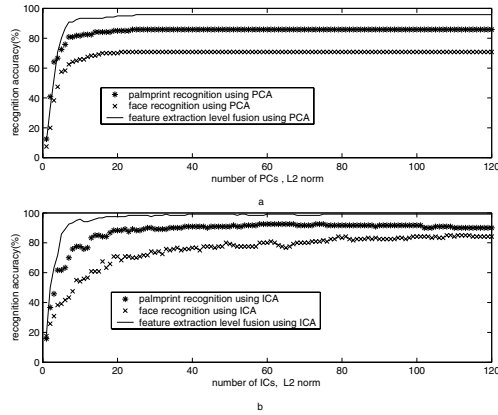
Fig. 3. Comparison between feature level fusion using PCA and individual face, palmprint classifier(a), and comparison between feature level fusion using ICA and individual face, palmprint classifier(b)

4   Experimental Results

4.1   Database

The face images we used are from the ORL database (Available at http://www.uk.research.att.com/facedatabase.html), this database contains 40 subjects and 400 images. We choose 6 images each in our experiment at random. Fig. 2a shows three images from the first person. There are no public domain palmprint image databases. Palmprint images from 300 individuals were collected using the palmprint capture device, which includes a ring source, CCD camera, lens, frame grabber, and A/D converter. The subjects mainly consisted of volunteers from the students and staff at the Hong Kong Polytechnic University. Every subject was asked to provide 6 images with the left palm. Concerning the compatibility to the ORL database, we only choose 240 images from 40 different palms. See Fig. 2d for examples (For the details of main process of preprocessing, see [8]).

Obviously, the ORL face database does not come with corresponding palmprint images, so to each face image, we assign an arbitrary (but fixed) palmprint class. Thus we obtain a database of 40 subjects, with 6 face images and 6 palmprint images per subject.

4.2   Experiments on Fusion Data Set and Analysis

The entire fusion database is divided into two parts. Three face images and corresponding palmprint images of each subject are used to construct the training data and the left are used for testing. The recognition accuracy is defined as the ratio between the number of correctly recognized persons and the number of

persons in the total test set. We use PCA to extract the features of face images and palmprint images, and then ICA. Fig. 2b shows three eigenfaces, Fig. 2c, three ICs of face images, Fig. 2e, three eigenpalms, and Fig. 2f, three ICs of palmprint images.

Fig. 3a shows the comparison between feature level fusion and standalone face and palmprint classifier using PCA. It can be seen that the standalone face classifier (PCA) can get an accuracy rate of 70.83% (which may result from the relatively large difference between training faces and test faces, because they are chosen 6 out of 10 and divided into two parts at random), the standalone palmprint classifier (PCA), much higher, 85.83%, and the feature extraction level fusion using PCA gives an improvement in terms of recognition accuracy rate, as high as 95.83%.

Fig. 3b shows the comparison between feature level fusion and standalone face and palmprint classifier using ICA. It can be seen that the standalone face classifier and palmprint classifier can get an accuracy rate of 85.00% and 92.50% respectively. When combined using feature extraction level fusion, however, the combined system shows a nearly perfect result, 99.17% recognition accuracy rate, which is very encouraging. We can see that the fusion system can get an improvement in performance when the performances of individual system are comparable. If the performances are almost similar, the improvement should be significant.

We have also tried SVM [11] classifier instead of the nearest neighbor classifier, but the results at present are not so encouraging as expected.

5   Conclusions and Future Work

Face recognition is non-invasive, but personal identification using the face is currently unreliable. Palmprint recognition is a relatively new branch of biometric technology, but the interested person must be cooperative and a specific acquisition device is needed. The results of this paper suggest that a multimodal biometric integrating of faces and palmprints can offer substantial performance gain that may not be possible with a single biometric indicator alone. Using ICA fusion on feature level with a moderate test set sized of 40 samples, recognition accuracy rate of up to 99.17% has been obtained. The size and the format of face and palmprint biometrics are compatible and the biometric characteristics are to some extent complementary, making the two biometrics suitable for feature level information fusion. The face and palmprint images can even be taken by one sensor, which leaves plenty of room to explore the information fusion of this bimodal biometric sources at feature extraction, matching score and decision levels. As is well known that PCA and ICA are two computationally expansive methods during training and thus may not be suitable for frequent addition of new users. Considering other methods with good computational properties, for example, SVD [10] may give some relief. In future work, we will also try on a larger database to test our proposed strategy. Our ultimate goal is to build up a real world biometrics system using the fusion of faces and palmprints.

Acknowledgement

References

[1] Jain, A. K., Bolle, R., Pankanti, S.(ed.): Biometrics: Personal Identification in Networked Society. Kluwer Academic, Dordrecht (1999)
[2] Zhao, W., Chellappa, R., Rosefeld, A., Phillips, P. J.: Face Recognition: A Literature Survey. Technical report, Computer Vision Lab, University of Maryland (2000)
[3] Ross, R., Jain, A. K.: Information Fusion in Biometrics, Pattern Recognition Letters 24 (13) (2003) 2115-2125
[4] Hong, L., Jain, A. K.: Integrating Faces and Fingerprints for Personal Identification. IEEE Trans. Pattern Anal. Mach. Intell. 20 (12) (1998) 1295-1307
[5] Wang, Y. H., Tan, T., Jain, A. K.: Combining Face and Iris Biometrics for Identity Verification. Proc. of 4th Int'l Conf. on Audio- and Video-Based Biometric Person Authentication, Guildford, UK, Jun. (2003) 805-813
[6] Kumar, A., Wong, D. C. M., Shen H. C., Jain, A. K.: Personal Verification Using Palmprint and Hand Geometry Biometric. Proc. of 4th Int'l Conf. on Audio- and Video-Based Biometric Person Authentication, Guildford, UK, Jun. (2003) 668-675
[7] Chang K., Bowyer K. W., Sarkar S., and Victor B.: Comparison and Combination of Ear and Face Images in Appearance-based Biometrics. IEEE Trans. Pattern Anal. Mach. Intell. 25 (9) (2003) 1160-1165.
[8] Zhang D., Kong W. K., You J., Wong M.: Online Palmprint Identification. IEEE Trans. Pattern Anal. Mach. Intell. 25 (9) (2003) 1041-1050
[9] Turk M., Pentland A.: Eigenfaces for Recognition. J. of Cognitive Neuroscience 3 (1) (1991) 71-86
[10] Hong Z.: Algebraic Feature Extraction of Image for Face Recognition. Pattern Recognition 24(1991) 211-219
[11] Guo G. D., Li Stan Z., Chan K. L.: Support Vector Machines for Face Recognition. Image and Vision Computing 19 (2001) 631-638
[12] Yuen P. C., Lai J. H.: Face Representation Using Independent Component Analysis. Pattern Recognition 35 (2002) 1247-1257
[13] Bartlett M. S.,Movellan, J.R, and Sejnowski T. J.: Face Recognition by Independent Component Analysis. IEEE Trans. Neural Network 13 (6) (2002) 1450-1464
[14] Lu G. M., Zhang D., Wang K. Q.: Palmprint Recognition Using Eigenpalms Features. Pattern Recognition Letters 24 (2003) 1463-1467
[15] Kumar A. and Zhang D.: Integrating Palmprint with Face for User Authentication. Proc. of Multi Modal User Authentication Workshop, Santa Barbara, CA, USA, Dec. (2003) 107-112
[16] X. Q. Wu, D. Zhang, and K. Q. Wang.:Fisherpalms Based Palmprint Recognition. Pattern Recognition Letters 24 (2003) 2829-2838

Biometric User Authentication
for Heightened Information Security

Qinghan Xiao

Defence R&D Canada – Ottawa, Ottawa, ON, K1A 0Z4 Canada
Qinghan.Xiao@drdc-rddc.gc.ca

Abstract. The need for heightened information security has expanded the research focus from securing the network to authenticating individual users. The aim of this paper is to present an infrastructure of trusted user authentication for heightened information security. Following techniques are developed to achieve the goal. Unlike the common post-boot authentication method, an approach of pre-boot biometric authentication is presented with different solutions to enhance network security. In real applications, privacy is a major concern in using biometric authentication systems. A two-way authentication method is therefore proposed, which will not only let the system authenticate the users, but also allow the users to verify the system before submitting their biometrics information. Based on these authentication strategies, a user authentication scheme for protecting high security resources is presented, which differentiates itself from most existing solutions by stressing security more than cost and ease-of-use.

1   Introduction

The security of modern information systems is composed of three primary components - authentication, authorization, and accountability. Authentication is the most fundamental of these three elements because it precedes the other two. In the information technology environment, authentication means either the process of verifying the identities of communicating equipment, or verifying the identities of the equipment's users. This paper is focusing on the later, in other words, "user authentication".

Presently, the majority of information systems use passwords as credentials. The major problem with this type of identification mechanism is that given a password, we cannot confirm that it belongs to the person who presents it. To provide secure information operating environments, a trusted authentication system must be developed to prevent unauthorized users from gaining access to classified or sensitive data. A strong authentication can be achieved by combining different authentication factors. This is especially important when a user is enrolled into a secure system and allowed to access the classified or sensitive data. Some unique, unchangeable aspect of the user must be used to verify his or her identity — biometrics fills that role.

The objectives of this paper are to position biometrics in the field of heightened information security and propose a trusted authentication solution for protecting sensitive resources. The paper focuses on two major issues facing the biometrics implementations – security enhancement and privacy protection, even though they

might be considered as opposite aspects. The techniques presented in this paper are pre-boot biometric authentication to ensure security, and two-way authentication that not only improves network security, but also enhances user privacy. Since the intention is to protect high security resources, the proposed solution emphasizes on security more than the cost and ease-of-use.

The five sections of this paper are organized in the following manner. Section 1 introduces the background and initiatives of the research. Section 2 presents a general overview of user authentication. Section 3 compares the security threats of pre-boot and post-boot authentications. Two schemes of pre-boot authentication are presented with analyses on their advantages and disadvantages. Section 4 develops a two-way authentication strategy to enhance both system security and user privacy. Based on the techniques developed in the previous sections, Section 5 presents an authentication system that preserves user privacy while protecting high security resources. Section 6 discusses the advantages of the proposed methods and concludes the paper.

2   User Authentication

User authentication is the process of confirming the identity claimed by a user. It can be accomplished by using one or more of the validation approaches: the knowledge factor (something users know), the possession factor (something users have), or the biometrics factor (something users are) [1]. The goal in user authentication is to allow valid parties access to databases and information services from anywhere at anytime. With the rapid evolution of information technology, the military's reliance on IT systems to carry mission-critical information is rapidly growing. As a result, the ability to achieve highly accurate authentication is becoming more critical for protecting sensitive information.

An authentication system is one used to validate the identity of the user that logging onto the system is the right person who claims to. Today's authentication systems have evolved based on decades of attacks, many of which were successful, starting with password-based systems in the early days of timesharing. For example, network sniffing can recover a user's password; a weakly protected password file allows its contents to be stolen; and interception and replay of a one-time password will block the legitimate user from successfully logging on. In the field of information security, the actual person at the other end of the network link must be identified, making simple user ID or token logons insufficient [2], [3]. To legally sign a user into the system and access to sensitive data, some physical part of the user should be verified to decrease certain types of impersonation attacks. Biometrics, which uses a unique physical or behavioral feature of the person being authenticated, satisfies this requirement better than the traditional methods that check the user identities by personal belongings. One of the methods to achieve a strong authentication incorporates two or three factors so that the benefits of one factor can compensate for the shortcomings of another.

Although various biometric authentication products exist, they are not yet good enough to work under combat conditions. Jonathan and Matthew [4] pointed out "the movement of biometrics into the mainstream is being fuelled not solely by companies'

desire to improve security per se, but by the recognition that biometrics can reduce user frustration". When implementing a biometrics application, both security and privacy are major issues even though they are considered opposite aspects. In the following sections, pre-boot biometric authentication and two-way authentication are presented with a system solution to enhance both network security and user privacy.

3   Pre-Boot Authentication

The authentication schemes used to protect information systems can be classified into two general patterns (Fig. 1):

- Pre-boot authentication: requires the correct user authenticated before the operating system loads. This can render the computer system completely useless to anyone but the authorized user.
- Post-boot authentication: allows the operating system to start up and become functional before user authentication happens.

Currently, most of the operating systems, such as Windows platforms, use post-boot authentication strategy. It is unable to provide boot protection, as their user-based security works only after starting the operating system. In such a way, since the authentication process has to go through a running workstation, we will have to take the risk of hidden Trojan horse programs that can launch the following attacks.

- Sniffing attack: using a sniffer program that copies information out of the authentication device input buffer after the user typed in his/her password to collect a biometric reading.
- Replay attack: replying biometric sensor reading to the authentication server to masquerade as the victim.

(a) Pre-boot authentication

(b) Post-boot authentication
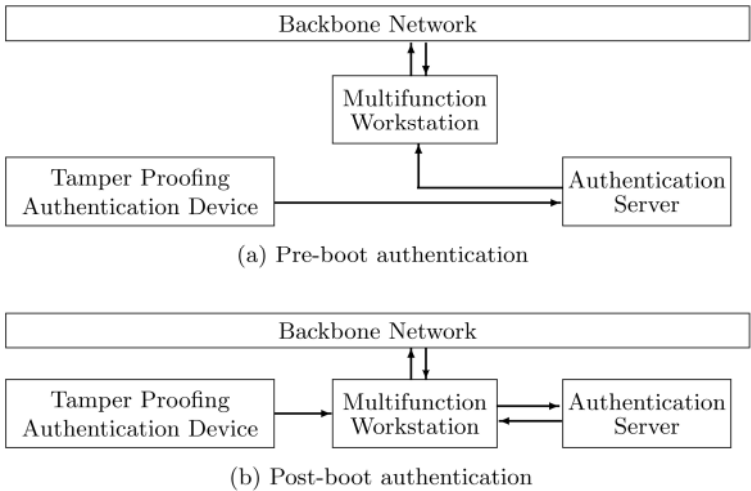
Fig. 1. Pre-boot and post-boot authentications

The pre-boot authentication scheme requires a user to be authenticated before booting the operating system. Without correct authentication, the information system cannot be started. Therefore, it will prohibit unauthorized assess to the system and prevent hackers from gaining control. There are several ways to implement pre-boot authentication functionality with different considerations on degrees of difficulty, security level and cost. The major difference is either using the existing workstation functionalities or building a separate authentication device, as shown in Fig. 2.

Because typical biometric authentication products need the computational power of the multifunction workstation to authenticate the user, the pre-boot authentication can be handled at the basic input/output system (BIOS) level before the operating system is read from the hard disk (Fig. 2 (a)). The advantage of this design lies in that it uses as much the workstation's resources as possible to reduce the cost. The disadvantage is that a Trojan horse on the workstation may affect the security of the communication. If there is no cryptographic module on the biometrics capture device, there is no way to send an encrypted biometric trait to the authentication server. This will be a critical problem.

A higher security pre-boot authentication scheme can be achieved by adding a separate tamper proofing authentication device that will generate all of the information required to boot the system after the user's identity is successfully verified by the authentication server (Fig. 2(b)). The advantage of adding a separate authentication device is that the network interface card of the workstation cannot be activated before the user is authenticated, which will reduce the system vulnerability. In such a device, the minimum requirements include operating system, hard drive controller, network interface card and cryptographic module if there is none on the biometrics capture device. Therefore, the major disadvantage is a higher cost.
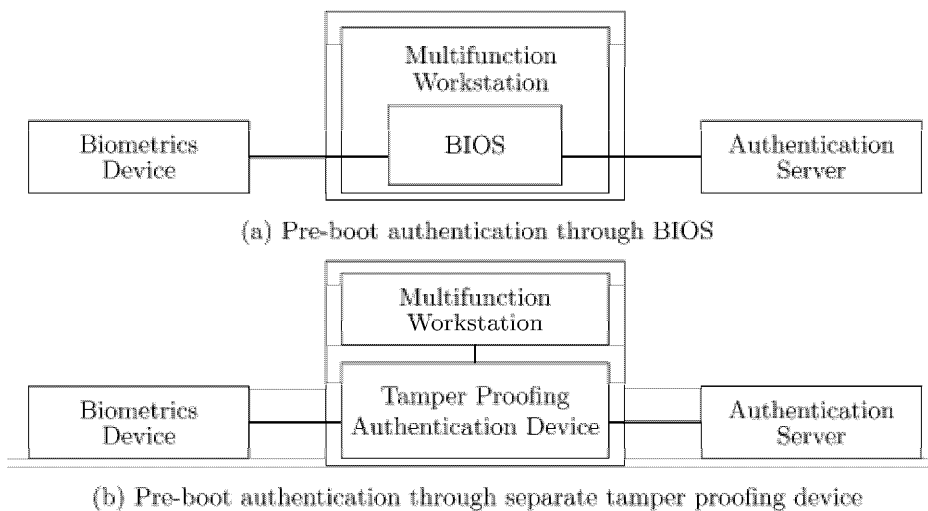
Fig. 2. Different pre-boot authentication schemes

4   Privacy Protection: Two-Way Authentication

To make biometric authentication successful, it is important to consider the user resistance, such as fear of unfamiliar technology, fear of exposure and fear of surveillance. Because the cooperation of the user can bring great impact on accuracy performance, the attitude of the users towards the intended biometrics can make or break the implementation of a biometric system. The major privacy concern is about the potential for mislaid biometric information and the danger of user profiles being used by hackers for other purposes.

It has been reported that certain hacking programs can pop up a login prompt on the victim's terminal [1]. If the user cannot recognise this as a fake prompt and input the login information, then his or her user name and password will be captured and no longer remain a secret. The same situation can happen to biometric authentication so that the user's biometric traits can be stolen too.

In order to solve this problem, a two-way authentication strategy is presented. That is the users are not merely authenticated by the system. They will verify the system too. It works in the following way. During the enrolment, a user is asked to select a password as well as a personal "secret" greeting phrase. Public Key Infrastructure (PKI) can be used to encrypt and decrypt this secret phrase. During the authentication, after the user presents the correct password, the corresponding personal greeting is displayed so that the user knows he will be submitting the biometrics to the right system. This solution will not only increase the security level of the system, but also make the users feel more confident in that their biometric traits are being collected by a system they can trust. When doing the enrolment, a user will have little privacy concerns due to trust placed in the organization collecting the biometrics. But when accessing the system remotely, the user cannot trust the station and network connection. Instead of going to the organization, the user's biometric trait may go to the hacker's hand and be misused later on. Therefore, it is crucial to let the user know that the biometric trait is sent to the right place. User acceptance is improved by giving user a sense of security and trust.
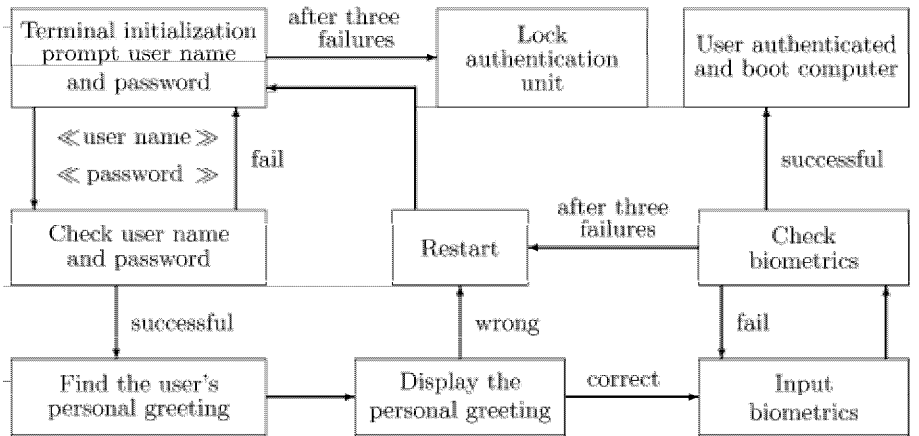
Fig. 3. Procedures of two-way authentication

Fig. 3. shows an abstraction scheme of the proposed procedures. After the user inputs the correct user name and password, the system will find the corresponding personal greeting phrase and send it to the user with a time stamp. If a wrong personal greeting phrase is displayed on the terminal, the user can restart the terminal and try again or contact the system manager. When getting a correct greeting phrase, the user can submit the biometrics. Because of the error rate of the biometrics capture device, it allows three failures before the system restarts. If the biometrics check gets a right result, the workstation will be booted, and the user can start to use the computer. If there are three failures at the terminal initialization stage, the authentication unit will lock out the user, and only the system manager can unlock the service for this person.

5   System Solution

Based on the methods discussed in the previous sections, an infrastructure of trusted authentication is presented to protect the information with heightened security. The communication and data transmission have to be protected. The information being sent through the network is convenient for users and cyberthieves alike. A problem exists if biometrics can be spoofed by hackers. Because the biometric templates will be transmitted over a network, attacks could involve the interception of biometric data and replay the data to access a security system or network later on. How this can be done depends on how the biometric data is transmitted on the network. Encryption of any user data moving over the network is a standard solution to this problem.

A basic strategy for cryptographically protecting biometric data is demonstrated as follows. The biometric device collects data and cryptographically protects the data. The device resides within a physically protected environment and uses a base secret to protect the data before and during transmission through the network. The receiver uses its own copy of the base secret to decrypt and verify the biometric data. For example, PKI can provide cryptography-based security services to network applications and services.

In a high security environment, the entity that is really at the other end of the network link being simply authenticated is not good enough. Furthermore, it has to be monitored throughout the information transmission. Currently, once logging onto a system there is no mechanism to monitor the user's identity. This security gap must be filled. Otherwise, the information can still be leaked to unauthorized people. To solve this problem, a function of user monitoring is proposed by detecting and tracking the facial images. After a successful authentication, user's face image is captured to generate a facial template. The identity of the user is checked by capturing and matching the facial images at a certain time interval. If the user disappears or changes to someone else, the system will log off automatically. In such a way, the proposed system not only authenticates the user who originally logged onto the system, but also monitors the user who is conducting the communications. By incorporating multiple usages of biometrics, a higher degree of security can be achieved. Fig. 4 presents a solution in which fingerprint identification is used to authenticate the users, and face recognition is utilized to monitor their status. The system can only be booted after successful authentication. Two-way authentication is introduced to enhance the user privacy, and the information transmission is protected by PKI encryption.
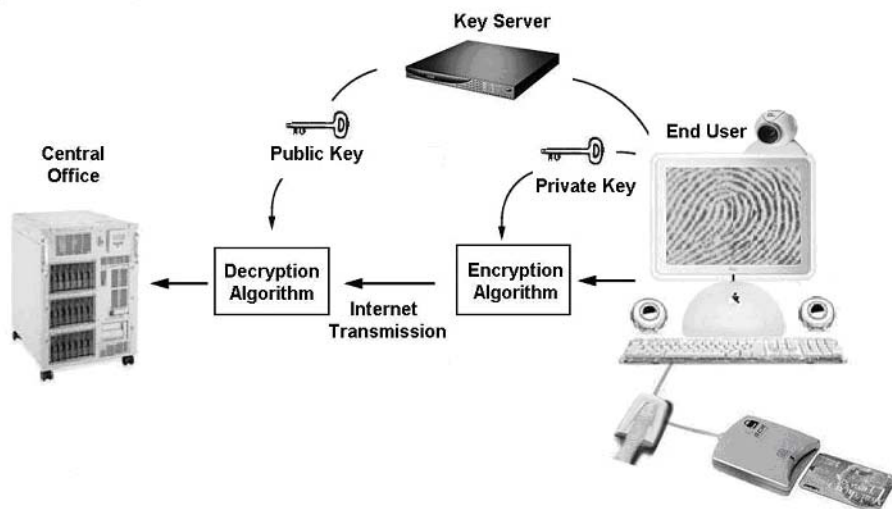
Fig. 4. Trusted user authentication and monitoring system

6   Discussion and Conclusion

This paper addresses the importance of user authentication in information and network securities. With the forces relying more and more on information technology to conduct daily business in a more efficient and responsive manner, the technology of user authentication must be improved to secure classified or sensitive information. When developing a biometric system, both security and privacy are major issues that could make a system's success or failure. The security and privacy issues are addressed in this paper by the development of pre-boot authentication, two-way authentication, and user monitoring. Based on these techniques, an infrastructure of trusted authentication system is presented to secure classified or sensitive information. Although there are various off-the-shelf solutions, the proposed approach is distinguished from them in the following aspects.

Pre-boot authentication technique prohibits unauthorized assess to the system and prevents hackers from gaining control over the system. Two pre-boot solutions with or without utilizing the computational power of the workstation are presented. The trade-off is cost verses security level.

One of the challenges for implementing biometric authentication system is user acceptance. This is the key to biometric implementation because until now most of the applications require user cooperation. A two-way authentication method that not only allows the system to authenticate users, but also let users authenticate the device before submitting their biometrics is presented to address this problem. In such a way, the users will feel more confident when using a biometrics device, while the security level is enhanced as well.

A system solution is presented, which considers different security aspects of designing a trusted authentication system, such as pre-boot authentication, two-way

authentication method, and user monitoring. Further research needs to be undertaken to consider overcoming spoofing problems by either multi-modal biometric technologies or hardware/software solutions for liveness detection.

Acknowledgements

References

[1]   Smith, R. E.: Authentication: From Passwords to Public Keys. Addison-Wesley, Upper Saddle River, NJ (2001).
[2]   Reid, P.: Biometrics for Network Security. Prentice Hall, Upper Saddle River, NJ (2003).
[3]   Matyas, V. Jr. and Riha, Z.: "Toward reliable user authentication through biometrics," IEEE Security & Privacy Magazine, Vol.1, Issue: 3, May-June (2003) 45–49.
[4]   Gossels, J. G. and Martin, M.: "Should you care about biometrics?" System Experts, Available: http://www.systemexperts.com/tutors/biometrics.pdf, (2001).

BioEVA: An Evaluation Tool for Biometric Algorithms

Luiz H. R. Sucupira Jr., Lívia C. F. Araújo , Miguel G. Lizárraga, and Lee L. Ling

School of Electrical and Computer Engineering, State University of Campinas
Albert Einstein Avenue, 400, PO Box 6101, Postal Code 13083-970, Campinas, SP, Brazil
`luigijr@yahoo.com`
`{liviacri,lizarrag,lee}@decom.fee.unicamp.br`

Abstract. This paper presents a tool, named BioEVA, to apply two forms of evaluation processes in biometric algorithms: comparative and qualitative. BioEVA has an internal engine that implements some metrics that we called quality parameters. We define some simple submission rules (protocol) must be followed before to submit a biometric algorithm to BioEVA's evaluation. The tool receives a biometric algorithm as a "black-box" and performs its automated evaluation. Using BioEVA, three biometric algorithms were evaluated: two based on static signatures and one based on keystroke dynamics.

1 Introduction

The main purpose for development of biometric algorithms and hardware devices for biometric samples acquisition is to guarantee more security in a user's authentication process. A third-party quality evaluation of biometric algorithms and systems will give validity to the developer's tests and results. In this evaluation, it is possible to compare algorithms, but only with the same database and the test environment [1]. However, in a competitive scenario, comparative tests sometimes are not desired because the consumers think that the best products are expensive, the worst ones are not able to be trusted, and the intermediate ones follow the relation "cost-benefit".

We are proposing a qualitative evaluation, based on the results produced on tests with biometric algorithms, with the possibility to compare them too. Then, we developed a tool called BioEVA that implements an environment to receive a biometric algorithm as a "black box", i.e., without the source code opened. Because of the possibility of apply an evaluation in only one biometric algorithm, we need to establish quality parameters to evaluate each kind of biometric technology, determining how good or bad they are under some criteria and definitions.

This paper is divided in 6 sections. In section 2, we will discuss about the ideas that provides BioEVA development. In section 3, we will show how to submit a biometric algorithm for its evaluation. In section 4, we will present the modules that compound the tool. In section 5, we will show the results of applying the tool on an evaluation of three biometric algorithms. Finally, in section 6, we will make some conclusions about the tool and the results produced in section 5.

2   Background

Before implementing BioEVA, some ideas of two previous works in the qualitative and comparative evaluations in biometric products were used to develop BioEVA. In next subsections, we will present these works and comment them.

2.1 Fingerprint Verification Competition (FVC)

The FVC is a competition that focuses on fingerprint verification algorithms. This competition provides us the following ideas:

− The protocol about how to submit an algorithm to a competitive evaluation.
− Definitions and criteria to compare biometric algorithms, which were used to define and configure some of the quality parameters used in this work.

The protocol used in FVC is based on command-line arguments. Besides, a participant needs to take two C-language skeletons (one for enrollment and the other for authentication), inserting their algorithms there. We propose the elimination of these skeletons, performing language and platform independence. Another issue is that the databases are not collected according to a formal protocol, and the participants have to adapt their algorithms to FVC collected samples or not submit them. We also propose an improvement based on the idea of a qualitative evaluation.

2.2 Methodology for Evaluation of Biometric Software Packages

A proposed methodology for evaluation of biometric software packages is one of our previous works [2]. We will use this methodology to establish a qualitative evaluation process. The common agreement established in [2] defines the specification of the evaluation process in manufacturer's presence, and we adapt it to a form that we named common agreement form. The common agreement form will be used to collect information about the biometric samples (fig. 1, process (1) - Preparing the Common Agreement). If the biometric samples are the same that the ones in BioEVA's database, then the process (6) − Collecting Samples for Tests in fig. 1 will use them. Otherwise, a new database must be collected by the Samples Collector in fig. 1, based on the developer's information on common agreement form.

The severity levels defined in [2] will be used to define the degree of performance and efficiency of the biometric algorithm. Five levels were defined: E is the lowest one and A is the highest one. To define the level of a biometric algorithm, first we apply the process (9) − Invoking the Evaluation Module in fig. 1, and BioEVA produces a biometric algorithm's score. Finally, the thresholds defined in [2] are applied: if the score is greater or equal than the threshold level, it is accepted to the respective severity level; otherwise, it is classified in a lower severity level.
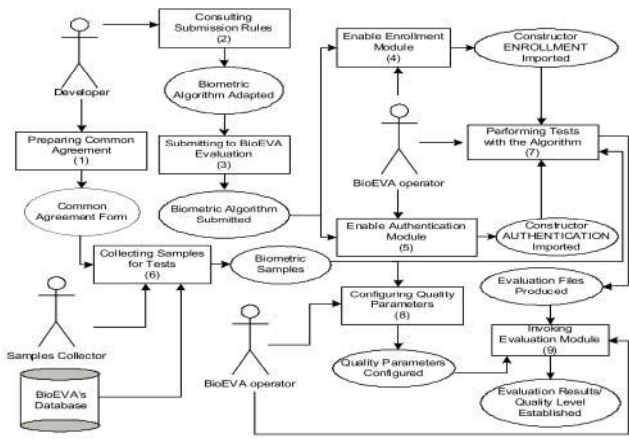
Fig. 1. BioEVA Evaluation Scheme

3   Submission Rules (Protocol)

The developer of the biometric algorithm must obey some simple rules (process (2) −
Consulting Submission Rules in fig. 1) before to submit it to BioEVA evaluation
(process (3) − Submitting to BioEVA Evaluation in fig. 1):

1. If the algorithm was developed in java, then the algorithm is submitted as a .jar
   file; otherwise it is submitted as a library file according to the platform. Thus, we
   provide to the participants the advantage of language and independence platform.
2. Two constructors (.jar file) or procedures (other files) must be created:
   ENROLLMENT and AUTHENTICATION. The BioEVA operator (fig. 1) will
   import the libraries that contain both constructors, using an import declaration
   (java sources) or the Java Native Interface (other sources), enabling the enrollment
   and the authentication modules (process (4) − Enable Enrollment Module and
   process (5) − Enable Authentication Module in fig. 1).
3. ENROLLMENT constructor's call must obey the following syntax for input data:

```
ENROLLMENT (id, samples, template_destination)
```

   Table 1 shows the input parameters for the ENROLLMENT constructor.

4. AUTHENTICATION constructor's call follows the syntax for input data below:

```
AUTHENTICATION (id, sample, templates_location)
```

   Table 2 shows the input parameters for the AUTHENTICATION constructor.

5. The outputs of both constructors will be used to generate the evaluation results
   (process (7) − Performing Tests with the Algorithm), using a method (function):

```
string output_results ( )
```

where `string` is the output that depends on the constructor. The syntax for ENROLLMENT constructor is "`Value+Total_samples`". `Value` indicates if the enrollment process was successful (`Value = 0`) or not (`Value ≠ 0`). If not, `Value` could be 1 (FAIL), 2 (TIMEOUT), or 3 (CRASH) [1]. `Total_samples` is the number of samples to enroll a user. The syntax for AUTHENTICATION constructor is "`&id-value`" (verification) or "`&id1-value1&id2-value2&id3-value3...&idN-valueN`" (identification). `Id` indicates the user's id related to a template which one the input sample was compared. `Value` indicates a value produced by the comparison of the input sample and one template. Using strings became the participant's efforts more simple to implement the constructor's output and solve the problems with more complex data structures.

Table 1. Input Parameters for ENROLLMENT Constructor

Input Parameters	Type	Description
`Id`	`string`	User's identification
`Samples`	`array of string`	Complete samples' path of the created templates
`template_destination`	`string`	Destination path of the created templates

Table 2. Input Parameters for AUTHENTICATION Constructor

Input Parameters	Type	Description
`Id`	`string`	User's identification (verification) or `null` (identification)
`Sample`	`string`	Complete samples' path
`template_location`	`string`	Destination path of the templates

4   BioEVA Tool

BioEVA main purposes are testing algorithm's performance and efficiency and it was implemented using Java 2 Platform. Three modules were implemented: enrollment, authentication and evaluation. In Enrollment and Authentication Modules, it is possible to BioEVA Operator selects if the algorithm is single or multimodal. Three files will be generated using the output of the method `output_results` (process (7) − Performing Tests with the Algorithm) for ENROLLMENT constructor:

− ParameterET.dat: it receives the times, in milliseconds, that the algorithm completes the enrollment processes.
− ParameterFTE.dat: it receives the values related to successful or fail in the enrollment processes.
− ParameterTSE.dat: it receives the values related to the amount of samples used to enroll the users on each enrollment process.

As result of the process (7) − Performing Tests with the Algorithm for AUTHENTICATION constructor, BioEVA will generate four files:

– Parameter_v'id'_AT.dat: it receives the times, in milliseconds, that the algorithm completes the verification processes. The 'id' in the file name identifies the user's id provided by BioEVA operator (fig. 1) in the authentication module.
– Parameter_v'id'_FFC_IGD.dat: it receives the resultant value of a user's template matching with a user's sample. The 'id' in the file name identifies the user's id provided by BioEVA operator (fig. 1) in the authentication module.
– Parameter_i_AT.dat: it receives the times, in milliseconds, that the algorithm completes the identification processes.
– Parameter_i_FFC_IGD.dat: it receives the resultant values of all templates matching in database with a user's sample.

In the Evaluation Module, all files generated in the previous modules (enrollment and authentication) are used to evaluate the algorithm. When the BioEVA Operator invoke this module (process (9) – Invoking Evaluation Module in fig. 1), BioEVA automatically uses the data stored in these files to calculate the score that the algorithm had obtained. Six quality parameters were implemented:

1. Impostor and Genuine Distribution (IGD): In these distributions each threshold has a number of users that are accepted as genuine and the remaining accepted as impostors. The module observes how much they are separated.
2. FAR / FRR curves (FFC): the evaluation is made through the EER (Equal Error Rate). These curves are built calculating the users' percentile value that were false rejected (FRR) and the ones that were false accepted (FAR) on each threshold.
3. Enrollment Time (ET): the module extracts the mean and standard deviation. Both are used to evaluate the enrollment time of the enrollment process.
4. Authentication Time (AT): the module extracts the mean and standard deviation. The module evaluates verification and identification separately. The AT score is calculated using a mean of the scores obtained in both authentication processes.
5. FTE rate (FTE): the module verifies how many enrollment trials are successful or not. In the case of failed enrollment process, we classify them according to [1].
6. Total Samples to Enroll (TSE): the module verifies the average, maximum and minimum samples that were used to enroll the users.

The quality values of each quality parameter are adjusted by the BioEVA operator (Process (8) – Configuring Quality Parameters in fig. 1). They are mapped in a satisfaction scale which range is [0,100]. A mean of the percentile values of all quality produces a final percentile value that will be used to classify the biometric algorithm according to a quality level as we shown in section 2.2.

5 Practical Results

Two algorithms based on static signatures (ASig1 and ASig2), and one based on keystroke dynamics (AKey1) were evaluated. The signatures database for ASig1 and ASig2 algorithms was compounded by signatures of 112 different users that signs 5 times for enrollment and 4 to 5 times for authentication, resulting in a database size of 1098 samples. The keystroke database for AKey1 algorithm was compounded by the

typing rhythm of 50 different users. Each user types 10 times for enrollment and 50 to 60 times for authentication. These results in a database size of 3250 samples.

Table 3 and 4 show the conFig.d scales of the quality parameters according to references [1], [3]-[12]. We configured the standard deviation presented on ET and AT quality parameters without use any references because this value can measure the stability degree of a biometric algorithm related to the mean time in enrollment and authentication processes.

The results of the BioEVA tool tests for algorithms ASig1, ASig2 and AKey1 are shown at second, fourth and sixth columns of table 5 respectively. The third, fifth and seventh columns of table 5 show the mapped percentile values. The mean of these values results in the final score of the ASig1 (74.83%), ASig2 (97.41%) and AKey1 (97.90%) respectively. According to the thresholds established in [2], ASig1 is a Level E, and ASig2 and AKey1 are Level A. In a comparative evaluation, observing the second and fourth columns in table 5, the performance and efficiency of ASig2 algorithm is better than ASig1.

Table 3. Scales for Quality Parameters FFC, FTE, TSE, ET and AT for ASig1 and ASig2

Parameters \ Scale			Excellent	Good	Regular	Insufficient
FFC			v≤2%	2%<v≤5%	5%<v≤10%	v>10%
FTE			v≤1%	1%<v≤3%	3%<v≤5%	v>5%
TSE			v≤5	5<v≤10	10<v≤15	v>15
ET	μ		v≤15s	15s<v≤20s	20s<v≤30s	v>30s
	σ		V≤5s	5s<v≤10s	10s<v≤15s	v>15s
AT	Verification	μ	V≤2s	2s<v≤5s	5s<v≤8s	v>8s
		σ	v≤0.5s	0.5s<v≤1s	1s<v≤3s	v>3s
	Identification	μ	V≤30s	30s<v≤60s	60s<v≤120s	v>120s
		σ	V≤10s	10s<v≤20s	20s<v≤30s	v>30s

Table 4. Scales for Quality Parameters FFC, FTE, TSE, ET and AT for AKey1

Parameters \ Scale			Excellent	Good	Regular	Insufficient
FFC			v≤3%	3%<v≤6%	6%<v≤12%	v>12%
FTE			v≤1%	1%<v≤3%	3%<v≤5%	v>5%
TSE			v≤10	10<v≤15	15<v≤30	v>30
ET	μ		v≤15s	15s<v≤20s	20s<v≤30s	v>30s
	σ		v≤5s	5s<v≤10s	10s<v≤15s	v>15s
AT	Verification	μ	v≤2s	2s<v≤5s	5s<v≤8s	v>8s
		σ	v≤0.5s	0.5s<v≤1s	1s<v≤3s	v>3s

Table 5. Evaluation Results for Algorithms ASig1, ASig2 and AKey1

Quality Parameters			ASig1		ASig2		AKey1	
			Results	**Mapped**	**Results**	**Mapped**	**Results**	**Mapped**
GID			-	50%	-	92%	-	88.6%
FFC			9%	13%	2.6%	93%	3.1%	98.9%
FTE			0.89%	100%	0%	100%	0%	100%
TSE			4 samples	100%	5 samples	100%	8 samples	100%
ET	μ		11.7s	100%	3.4s	100%	0.056s	100%
	σ		1.1s		0.97s		0.03s	
AT	Verification	μ	2.6s	86%	1.2s	99.5%	0.022s	100%
		σ	0.9s		0.5s		0.025s	
	Identification	μ	32.7s		20.5s		-	
		σ	4.3s		5.1s		-	

6   Conclusion

The BioEVA tool was developed to evaluate biometric algorithms like a "black-box", performing comparative and qualitative evaluations. Some rules must be followed by the developers. The tests results in enrollment and authentication modules are stored in files that will be used to perform an automated evaluation. We establish 6 quality parameters and a novel quality value using the standard deviation for AT and ET. The BioEVA tool performs an evaluation in single or multimodal biometric algorithms.

This work improves some issues presented in the FVC competition. The C-language skeletons were removed, which provides language independence, and a formal document was established named common agreement form. BioEVA supports platform independence: the Java Virtual Machine (JVM) and the Java Native Interface (JNI) will provide this feature through library files. Finally, we tested three algorithms: two based on static signatures and one based on keystroke dynamics.

Acknowledgement

References

[1]   Maio, D., Maltoni, D., Cappeli, R., Wayman, J.L., Jain, A. K.: FVC2002: Second Fingerprint Verification Competition, *Proc. of Int. Conf. on Pattern Recognition*, Quebec City (2002).

[2]   Sucupira Jr., L.H., Araújo, L.C.F., Lizárraga, M.G., Ling, L.L., "Evaluating Biometric Software Packages using a Practioners Methodology", *Proc. of ISCA 16th Int. Conference on Computer Applications in Industry and Engineering (CAINE-2003)*, USA (2003) 310-313.

[3]   Wirtz, B.: Average Prototypes for stroke-based signature verification, *ICDAR 97*, vol. 1, Germany, (1997) 268-272.
[4]   Lee, L.L., Lizárraga, M.G.: Off-line methods for human signature verification, *Proceedings of IASTED International Conference on Signal and Image Processing SIP-96*, USA (1996).
[5]   Lizárraga, M.G.: Um Sistema Biométrico de Identificação Pessoal via Internet com ênfase em Assinaturas Estáticas. Doctorade Thesis, UNICAMP, August (2000).
[6]   Plamondon, R., Srihari, S.N.: On-Line and Off-Line Handwriting Recognition: Compreensive Survey, *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 22, N° 1, (2000) 63-84.
[7]   Joyce, R., Gupta, G.: Identity authorization based on keystroke latencies, *Communications of the ACM*, Vol 33. N° 2, (1990) 168-176.
[8]   De Ru, W.G., Eloff, J.H.P.: Enhanced Password Authentication through Fuzzy Logic, *IEEE Expert / Intelligent Systems & Their Applications*, Vol. 17, No. 6, (1997) 38-45.
[9]   Robinson, J.A., Liang, V.M., Michael, J.A.: Computer User Verification Login String Keystroke Dynamics, *IEEE Trans. Syst., Man, Cybern.*, Vol. 28, No. 2, (1998) 236-241.
[10] Lin, D.T.: Computer-Access Authentication with Neural Network Based on Keystroke Identity Verification, *International Conference on Neural Networks*, Vol. 1, (1997) 174-178.
[11] Bleha, S., Slivinsky, C., Hussain, B.: Computer-Access Security Systems Using Keystroke Dynamics, *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 12, No. 12, (1990) 1217-1222.
[12] Haidar, S., Abbas, A., Zaidi, A.K., A Multi-Technique Approach for User Identification through Keystroke Dynamics, *IEEE Int. Conf Syst.,Man Cybern.*, Vol. 2, (2000) 1336-1341.

Application of Multi-criteria Analysis for the Creation of a Risk Assessment Knowledgebase for Biometric Systems

Christos K. Dimitriadis[1] and Despina Polemi[2]

[1] Expertnet SA, 244 Kifisias Av., 15231 Athens, Greece
Christos.Dimitriadis@expertnet.net.gr
[2] University of Piraeus, 80 A. Dimitriou, 18534 Piraeus, Greece
dpolemi@unipi.gr

Abstract. This paper, presents an enhanced version of a knowledgebase of vulnerabilities, risks and countermeasures for biometric systems. The knowledgebase was created by the application of the Multi-criteria Analysis methodology to the results of desk research, laboratory testing and interviews. The knowledgebase aims to improve risk assessment procedures, by adding the capability of analyzing the risk of the biometric component of an information system. The application of the knowledgebase is demonstrated for clarifying its functions.

1 Introduction

Risk assessment is a critical process towards the design and deployment of information technology security architectures [20][21]. Regardless of the risk assessment methodology deployed, it is a common practice to utilize a database of common risks and countermeasures (called knowledgebase) for assuring the effectiveness of the process [4]. Such knowledgebases are sources of expertise regarding security issues for the various components of information systems, assuring that no risks will be missed and adequate countermeasures will be proposed during the process. Despite the existence of biometric-specific standards [22] and best practices [1], no detailed knowledgebases exist for assisting the risk assessment process, as far as biometrics are concerned. This paper, presents an extended risk assessment knowledgebase, which aims to enhance existing risk assessment methodologies and tools, by adding the capability of analyzing the risk of the biometric component of the system. The functionality of the knowledgebase is demonstrated by the conduct of risk analysis in a real system. Part of this work is the author's contribution to the EC project BIOSEC [15]. The remainder of the paper is organized in the following main sections:

- Methodology: Describing the methodology for the creation of the knowledgebase.

- Biometric risk assessment Knowledgebase (BK): Containing the vulnerabilities and risk factors of biometric systems, as well as the countermeasures for risk reduction.
- Demonstration of BK's functions: Presenting a risk analysis procedure based on BK, on a real system that incorporates biometrics.

2   Methodology

A general risk assessment model [2][3], which is implemented by most of the standard methodologies, was followed. This model includes: asset identification, threat identification (events that can potentially cause undesirable effects), vulnerability identification (security weaknesses of the system), risk identification (probability of a threat exploiting a vulnerability) and identification of countermeasures for risk reduction.

Asset and threat identification are covered sufficiently by standard risk assessment methodologies without the need of a specialized BK. Vulnerability identification is addressed by the development of a catalogue of vulnerabilities for biometric systems, which was populated by conducting extensive desk research on known or possible attacks against various biometric technologies, penetration tests on biometric systems in a dedicated lab and interviews with experts in the field. Risk identification, depends on the risk assessment methodology [5][6]. A successful practice is to deploy tools [7], which utilize predetermined risk scores for each identified vulnerability, based on the estimation of experts who studied the likelihood of occurrence of vulnerability exploits. For the creation of BK, a quantitative approach was chosen, calculating a risk factor for each vulnerability. The risk factor is an indicator of the importance of the vulnerability and the sum of all risk factors provides the total risk factor of the biometric component of the information system under review. In order to calculate the risk factor and provide an objective evaluation of each vulnerability a methodology called Multi-Criteria Analysis (MCA) was deployed. MCA [8][19] is a methodology to evaluate different alternatives (currently biometric vulnerabilities) and to determine an order of ranking of these alternatives. MCA, through the attachment of weighing factors to the different criteria, takes into account that some specific criteria should be more influential in the determination of the ranking between alternatives. Finally, the identification of countermeasures is addressed through the conduct of tests, desk research and interaction with experts in the field of security and biometrics.

The final form of BK, is a set of vulnerabilities followed by the corresponding risk factors and proposed countermeasures.

3   Biometric Risk Assessment Knowledgebase

A short description of the identified vulnerabilities is given below:

1. Spoofing: Poor biometric implementations are vulnerable to spoofing and mimicry attacks. An artificial finger made of commercially available silicon or

gelatin, can deceive a fingerprint biometric sensor [9][10]. The use of pictures, masks, voice recordings or speech synthesis tools is possible to deceive iris, face, and voice recognition systems.

2. Fake templates: Server based architectures, where the biometric templates are stored centrally, inherit the vulnerabilities of such systems [14]. A possible attack can be realized when the impostor inserts his template in the system under someone else's name.

3. Replay: Data could be captured from a communication channel, between the various components of a biometric system [14], in order to be replayed at another time for gaining access.

4. Cross system: The utilization of the template in two or more applications with different security levels (i.e. convenience applications and security applications) tends to equalize these security levels, by decreasing the higher security level to the lower one - if a template is compromised in one application, it can be used for gaining access to the other.

5. Component alteration: A possible attack can be realized with a Trojan Horse on the feature extractor, the matching algorithm or the decision algorithm of the system, acting as a manipulator of each component's output.

6. Enrolment, administration and system use: Poor enrolment, system administration and system use procedures, expose the biometric system. During the enrolment phase, raw biometric data and biometric templates can be compromised and databases can be altered or filled with imprecise user data. Poor system administration procedures, in addition to the above, might lead to altered system configuration files, with decreased FAR, making false acceptance easier, thus security weaker. Similarly, a user might exceed his/her authority, threatening the system.

7. Noise and power loss: Off-limit power fluctuation or flooding of a biometric sensor with noise data - for example flashing light on an optical sensor, changing the temperature or humidity of the fingerprint sensor, spraying materials on the surface of a sensor or vibrating the sensor outside its limits - might cause the biometric device to fail.

8. Power and timing analysis: Capturing the power consumption of a chip can reveal the software code running on the chip, even the actual command[12][13]. The application of Simple Power Analysis and Differential Power Analysis techniques is possible to break the matching mechanism of the biometric system or reveal the biometric template. Timing attacks are similar and measure the processing time instead of the power consumption.

9. Residual characteristic: The residual biometric characteristic of a user on the sensor may be sufficient to allow access to an impostor (e.g. a fingerprint the sensor). The attack is realized on a fingerprint sensor with a residual fingerprint from the previous measurement, by pressing a thin plastic bag of warm water on the sensor, by breathing on the sensor or by using dust with graphite, attaching a tape to the dust and pressing the sensor [14].

10. Similar template - Similar characteristics: A user having a similar template or a similar characteristic with a legitimate one, might deceive the system, especially

in identification applications, where one to many template comparisons are conducted.

11. Brute force: The impostor is attempting continuously to enter the system, by sending incrementally increased matching data to the matching function until a successful score is accomplished [16][17]. Biometrics however are more resistant to this attack, than traditional systems, since the impostor has to find a way to insert the trial data to the system, thus combine this vulnerability with one of those described above.

12. Identity management implementation: Poor identity management implementations in biometric systems, can lead to privacy violation, in terms of user tracking and transaction tracking [15].

A detailed description of the identified vulnerabilities and countermeasures can be found in [11]. MCA was applied for each vulnerability, according to the following steps:

1. Criteria, for evaluating vulnerabilities and which influence their probability of occurrence were selected. These are the difficulty to exploit the vulnerability (C1) in terms of technical expertise required and complexity, the effectiveness (C2), in terms of level of exposure to threats - binding the vulnerability with the threat and the cost (C3) in terms of special equipment required.

2. For each vulnerability, a score was calculated per criterion, after processing results from the desk research, biometric lab tests and interviews. The scores were based on a common quantitative scale (from 1-10). For C1 the highest score (10) represents the lowest difficulty, for C2 the highest score (10) represents the highest effectiveness and for C3 the highest score (10) represents the lowest cost.

3. The criteria were prioritized by the assignment of weighing factors. Criteria C1 and C3 were assigned with higher weighing factors (equal to 3) than C2 (weighing factor equal to 1), reflecting the most common attack profiles and following the observation that attackers test vulnerability exploits when they are easy to exploit and inexpensive, considering effectiveness at a latter stage [21].

4. The vulnerabilities were ranked according to the injunction MCA method. This method multiplies the scores of the criteria with the correspondent weighing factors and calculates the sum of these products. For example, the scores for the power and timing analysis (C1: 1, C2: 8, C3: 1), which is a difficult, effective and expensive attack (in relation to others), were multiplied with the corresponding weighing factors (C1: 3, C2: 1, C3: 3) and the sum of these products was calculated.

5. All scores were transformed to percentages of the maximum total score of all vulnerabilities. This action was performed, in order to achieve a maximum of 100% when all vulnerabilities are present and at the same time preserve a common denominator for all vulnerabilities. In the example, this resulted the risk factor of the power and timing analysis vulnerability to be 3%.

The risk factors were individually produced in the cases of vulnerabilities that were specific for each biometric technology (fingerprint, iris, face, voice). The following table presents the calculated risk factors and the corresponding

vulnerabilities and proposed countermeasures per technology (fingerprint: Fi, Iris: Ir, Face: Fa, Voice: V).

Table 1. Identified *Vulnerablities*, calculated *Risk Factors* and proposed *Countermeasures*

Vulnerability	Risk Factor (%)				CM No.
	Fi	Ir	Fa	V	
1. Spoofing	10	8	11	13	i, ii, iii
2. Fake templates	14				iv, v
3. Replay	10				iv, vi, xiii
4. Cross system	5				vii
5. Component alteration	10				iv, vi
6. Enrolment, administration and system use	17				iv
7. Noise and power loss	3	3	3	5	iv
8. Power and timing analysis	3				viii
9. Residual characteristic	6	0	0	0	iii, ix
10. Similar template - Similar characteristics	2	2	6	6	ix, x
11. Brute force (verification applications)	3				iv, xi,
12. Identity management implementation	14				xii
Countermeasures					
i. Vitality detection: measurement of one or more attributes (the relative dielectric constant, the conductivity, the heartbeat, the temperature, the blood pressure, the detection of vitality under the epidermis, or the spontaneous dilation and constriction of the pupil or eye movement)					
ii. Multimodal architecture.					
iii. Interactive authentication – challenge and response.					
iv. Well-implemented security policy according to standards.					
v. Storage of the template in a secure medium.					
vi. System integration into a hardware security module.					
vii. Custom biometric encoding algorithms – hash functions.					
viii. Noise generators, low power consumption chips and time-neutral software design.					
ix. Technology assessment.					
x. Calibration review.					
xi. Traditional controls - account lock after a number of attempts.					
xii. Identity privacy support (temporary identities, pseudonyms etc).					
xiii. Authentication between the components of the system.					

The main functions of BK are the identification of the applicable vulnerabilities, the calculation of the sum their risk factors and the implementation of countermeasures for risk reduction. These functions are demonstrated in the next section.

4   Demonstration of BK's Functions

Risk analysis was conducted, during the implementation of the project BioAthletic [18], for evaluating the security level of an access control system for stadiums based on biometric technologies. The system is composed of terminals with embedded fingerprint biometric devices, which are located in the entrances of the stadium. The terminals communicate with a central server through a wireless LAN. The central server hosts an administration system and a database filled with data concerning access privileges for sports fans. Each sports fan possesses a smart card with his template stored in it. Biometric authentication is realized locally in the terminal and if positive an identifier is submitted to the central server for requesting access privileges.

The vulnerabilities that are applicable to the system are the following:

- Spoofing: since no vitality detection features were present, nor any compensating controls in place.
- Cross system: since the biometric algorithms were not custom.
- Enrolment, administration and system use: Since enrolment, administration and system use was poor and not designed and implemented according to standards.
- Power and timing analysis: since the smart card had no countermeasures implemented against these types of attacks.
- Identity management implementation: Since the identity management mechanism was poor with no identity privacy protection functions.
- The vulnerabilities that are not applicable to the system are the following:
- Fake templates: Since the template was stored in the protected memory of the smart card.
- Replay and component alteration: Since the biometric component was limited in a hardware security module (black box) and the environment was controlled by the personnel of the stadium.
- Noise and power loss: Since the corresponding part of the security policy implementation ensured a controlled environment for the biometric devices.
- Residual characteristic Similar template - Similar characteristics: Since the algorithm performance had adequate performance references.

Brute force (verification applications): Since the corresponding part of the security policy implementation ensured a controlled environment for the biometric devices.

The calculation of the sum of the risk factors for all applicable vulnerabilities, revealed a total risk factor of 49%. A cost benefit analysis indicated the implementation of countermeasures for the enrolment, administration and system use and identity management implementation vulnerabilities. A recalculation after implementing the countermeasures indicated that the residual risk factor was 18%.

5   Conclusions

Special care should be given to user enrolment, system administration and use, implementing as a mandatory control, concrete security policies based on international standards. Central storage of templates, heavily increase the risk level of

the system, uncovering the demanding need for encryption and strong intrusion prevention, detection and response countermeasures. Vitality detection was also identified as a demanding need, which can be relatively compensated by interactive authentication techniques. The restriction of the biometric template to a hardware security module and the elimination of the template submission over communication links and networks, addresses a great number of vulnerabilities and reduces the total risk factor significantly. The conduct of risk assessment is a significant step towards the creation of security architectures, which promote the advantages of biometric systems in a risk-proof manner.

References

[1]   Wayman, J.L., Mansfield, A.J.: Best practices of testing and reporting performance of biometric devices. `http://www.cesg.gov.uk/site/ast/biometrics/media/BestPractice.pdf.` (2002)

[2]   Certified Information Systems Auditor Manual. Information Systems Audit and Control Association (2003)

[3]   Peltier, T.R.: Information Security Risk Analysis. CRC press LLC USA (2001)

[4]   King, M., Dalton, C., Osmanoglu, T.: Security Architecture. RSA press USA (2001)

[5]   Operationally Critical Threat, Asset, and Vulnerability Evaluation method (OCTAVE). `http://www.cert.org/octave`

[6]   CCTA Risk Analysis and Management Method (CRAMM). `http://www.cramm.com.`

[7]   Consultative, Objective and Bi-functional Risk Analysis (COBRA). `http://www.security-risk-analysis.com/introcob.htm`

[8]   Multi-Criteria Analysis manual. `http://www.odpm.gov.uk`

[9]   Matsumoto, T., Matsumoto, H., Yamada, K., Hoshino, S.: Impact of artificial fingers on fingerprint systems. Proceedings of SPIE, Vol. 4677. Yokohama (2002)

[10]   Van der Putte, T., Keuning, J.: Biometrical fingerprint recognition – don't get your fingers burned. IFIP TC8/WG8.8 Fourth Working Conference on Smart Card Research and Advanced Applications. Kluwer Academic Publishers. (2000) 289-303

[11]   Dimitriadis, C., Polemi, D.: Risk analysis of biometric systems. ICEIS –WOSIS, Lecture Notes in Computer Science. Springer-Verlag (2004) – to appear

[12]   Gandolfi, K., Mourtel, C., Olivier, F.: Electromagnetic Analysis: Concrete Results. Lecture Notes in Computer Science, Vol. 2162. Springer-Verlag (2001) 251-261

[13]   Kocher, P., Jaffe, J., Jun, B.: Introduction to Differential Power Analysis and Related Attacks. `http://www.cryptography.com/technology/dpa/DPATechnicalInfo.PDF.` (1998)

[14]   IST-1999-20078 Business environment of biometrics involved in e-commerce - BEE. `http://expertnet.net.gr/bee` (2002)

[15]   IST-2002-001766 Biometrics and Security – BIOSEC. `http://biosec.tid.es`

[16]   Bolle, R.M., Connell, J.H., Ratha, N.K.: Biometric perils and patches. Pattern Recognition, Vol. 35, no. 12 (2002) 2727-2738

[17]   Smith, R.: The biometric Dilemma. Secure Computing (2002)

[18]   GSRT – Access Control System for Stadiums base on Biometrics - BIOATHLETIC

[19]   ISO/IEC 17799 IT – Code of practice for information security management

[20]   Information Systems Audit and Control Association: COBIT: Control Objectives for Information and related Technology

[21]   Know your enemy series. The Honeynet project. `http://www.honeynet.org`

[22]   ANSI X9.84: Biometric Information Management and Security

Soft Biometric Traits
for Personal Recognition Systems

Anil K. Jain[1], Sarat C. Dass[2], and Karthik Nandakumar[1]

[1] Department of Computer Science and Engineering
Michigan State University, MI - 48824, USA
{jain,nandakum}@cse.msu.edu
[2] Department of Statistics and Probability
Michigan State University, MI - 48824, USA
sdass@stt.msu.edu

Abstract. Many existing biometric systems collect ancillary information like gender, age, height, and eye color from the users during enrollment. However, only the primary biometric identifier (fingerprint, face, hand-geometry, etc.) is used for recognition and the ancillary information is rarely utilized. We propose the utilization of "soft" biometric traits like gender, height, weight, age, and ethnicity to complement the identity information provided by the primary biometric identifiers. Although soft biometric characteristics lack the distinctiveness and permanence to identify an individual uniquely and reliably, they provide some evidence about the user identity that could be beneficial. This paper presents a framework for integrating the ancillary information with the output of a primary biometric system. Experiments conducted on a database of 263 users show that the recognition performance of a fingerprint system can be improved significantly ($\approx 5\%$) by using additional user information like gender, ethnicity, and height.

1 Introduction

Biometric systems automatically recognize individuals based on their physiological and/or behavioral characteristics like fingerprint, face, hand-geometry, iris, retina, palm-print, voice, gait, signature, and keystroke dynamics [1]. Biometric systems that use a single trait for recognition, called unimodal biometric systems, are affected by problems like noisy sensor data, non-universality and/or lack of distinctiveness of the chosen biometric trait, unacceptable error rates, and spoof attacks. Some of the problems associated with unimodal biometric systems can be overcome by the use of multimodal biometric systems that combine the evidence obtained from multiple sources [2]. A multimodal biometric system based on different biometric identifiers like fingerprint, iris, face, and hand-geometry can be expected to be more robust to noise, address the problem of non-universality, improve the matching accuracy, and provide reasonable protection against spoof attacks. However, such a system will require a longer verification time thereby causing inconvenience to the users.

A possible solution to the problem of designing a reliable and user-friendly biometric system is to use ancillary information about the user like height, weight, age, gender, ethnicity, and eye color to improve the performance of the primary biometric system. Most practical biometric systems collect such information about the users during enrollment. However, this information is not currently utilized during the automatic identification/verification phase. Only when a genuine user is falsely rejected by the system, a human operator steps in to verify the soft biometric traits of the user. If these characteristics can be automatically extracted and utilized during the decision making process, the overall performance of the system will improve and the need for manual intervention will be reduced. The ancillary information by itself is not sufficient to establish the identity of a person because these traits are indistinctive, unreliable, and can be easily spoofed. Hence, we define *soft biometric traits* as *characteristics that provide some information about the individual, but lack the distinctiveness and permanence to sufficiently differentiate any two individuals.* The soft biometric traits can either be continuous (e.g., height and weight) or discrete (e.g., gender, eye color, ethnicity, etc.). In this paper, we describe a framework for integrating the information provided by the soft biometric indicators with the output of the primary biometric system. We also analyze the performance gains obtained by integrating the ancillary information like gender, ethnicity, and height with the output of a fingerprint biometric system.

2   Related Work

The first personal identification system developed by Alphonse Bertillon [3] for identification of criminals was based on three sets of features: (i) anthropometric measurements like height and length of the arm, (ii) morphological description of the appearance and body shape like eye color and anomalies of the fingers, and (iii) peculiar marks observed on the body like moles and scars. Although the Bertillon system was useful in tracking criminals, it had an unacceptably high error rate because the features used are indistinctive (several individuals can have the same set of measurements) and non-permanent (for the same individual, the measurements can change over time). Heckathorn et al. [4] have shown that a combination of personal attributes like gender, race, eye color, height, and other visible identification marks can be used to identify an individual only with a limited accuracy. Hence, a system that is completely based on soft biometric traits cannot meet the accuracy requirements of real-world applications. However, soft biometric traits can be used to improve the performance of traditional biometric systems.

Wayman [5] proposed the use of soft biometric traits like gender and age, for filtering a large biometric database. Filtering refers to limiting the number of entries in a database to be searched, based on characteristics of the interacting user. For example, if the user can somehow be identified as a middle-aged male, the search can be restricted only to the subjects with this profile enrolled in the database. This greatly improves the speed or the search efficiency of the biomet-

ric system. In general, filtering reduces the time required for identification but errors in filtering can degrade the recognition performance. Some studies [6, 7] have shown that factors such as age, gender, race, and occupation can affect the performance of a biometric system. For example, a young female Asian mineworker is seen as the most difficult subject for a fingerprint system [7]. This provides the motivation for tuning the biometric system parameters like threshold on the matching score in a unimodal biometric system, and thresholds and weighting of the different modalities in a multimodal biometric system to obtain the optimum performance for a particular user or a class of users. Filtering and system parameters tuning require an accurate classification of a user into a particular class or bin (e.g., male or female, blue or brown eyes, Caucasian or Asian or African). This requires a pre-identification module that can accurately perform this classification.

3   Framework for Integration of Soft Biometrics

In our framework, the biometric recognition system is divided into two subsystems. One subsystem is called the primary biometric system and it is based on traditional biometric identifiers like fingerprint, face and hand-geometry. The second subsystem, referred to as the secondary biometric system, is based on soft biometric traits like age, gender, and height. Figure 3 shows the architecture of a personal identification system that makes use of both primary and soft biometric measurements. Let $\omega_1, \omega_2, \cdots, \omega_n$ represent the n users enrolled in the database. Let $\mathbf{x}$ be the feature vector corresponding to the primary biometric. Without loss of generality, let us assume that the output of the primary biometric system is of the form $P(\omega_i \mid \mathbf{x}), \ i = 1, 2, \cdots, n$, where $P(\omega_i \mid \mathbf{x})$ is the probability that the test user is ω_i given the feature vector $\mathbf{x}$. If the output of the primary biometric system is a matching score, it is converted into posteriori probability using an appropriate transformation. For the secondary biometric system, we can consider $P(\omega_i \mid \mathbf{x})$ as the prior probability of the test user being user ω_i.

Let $\mathbf{y} = [y_1, \ \cdots, \ y_k, \ y_{k+1}, \ \cdots, \ y_m]$ be the soft biometric feature vector, where y_1 through y_k are continuous variables and y_{k+1} through y_m are discrete variables. The updated probability of user ω_i, given the primary biometric feature vector $\mathbf{x}$ and the soft biometric feature vector $\mathbf{y}$, i.e., $P(\omega_i \mid \mathbf{x}, \ \mathbf{y})$ can be calculated using the Bayes rule as

$$P(\omega_i|\mathbf{x}, \ \mathbf{y}) = \frac{p(\mathbf{y}|\omega_i) \ P(\omega_i|\mathbf{x})}{\sum_{i=1}^{n} p(\mathbf{y}|\omega_i) \ P(\omega_i|\mathbf{x})} \ . \tag{1}$$

If we assume that the soft biometric variables are independent, equation (1) can be rewritten as

$$P(\omega_i|\mathbf{x}, \ \mathbf{y}) = \frac{p(y_1|\omega_i) \cdots \ p(y_k|\omega_i) \ P(y_{k+1}|\omega_i) \cdots \ P(y_m|\omega_i) \ P(\omega_i|\mathbf{x})}{\sum_{i=1}^{n} p(y_1|\omega_i) \cdots \ p(y_k|\omega_i) \ P(y_{k+1}|\omega_i) \cdots \ P(y_m|\omega_i) \ P(\omega_i|\mathbf{x})} \ . \tag{2}$$
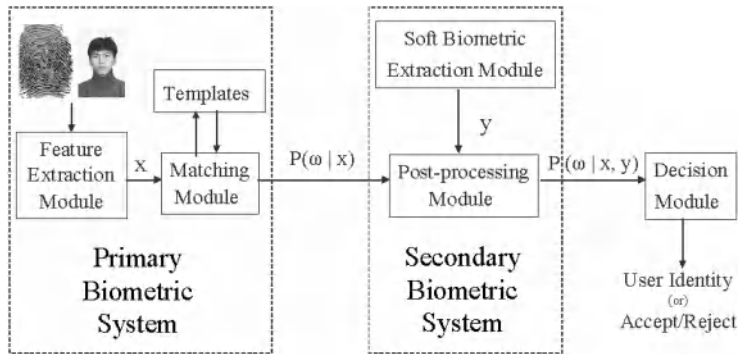
Fig. 1. Integration of Soft Biometric Traits with a Fingerprint Biometric System (x is the fingerprint feature vector, y is the soft biometric feature vector)

In equation (2), $p(y_j|\omega_i)$, $j = 1, 2, \cdots, k$ represents the conditional probability of the continuous variable y_j given user ω_i. This can be evaluated from the conditional density of the variable j for user ω_i. On the other hand, discrete probabilities $P(y_j|\omega_i), j = k+1, k+2, \cdots, m$ represents the probability that user ω_i is assigned to the class y_j. This is a measure of the accuracy of the classification module in assigning user ω_i to one of the distinct classes based on biometric indicator y_j. In order to simplify the problem, let us assume that the classification module performs equally well on all the users and therefore the accuracy of the module is independent of the user.

The logarithm of $P(\omega_i|\mathbf{x}, \mathbf{y})$ in equation (2) can be expressed as

$$\log P(\omega_i|\mathbf{x}, \mathbf{y}) = \log p(y_1|\omega_i) + \cdots + \log p(y_k|\omega_i) + \log P(y_{k+1}|\omega_i) + \cdots$$

$$+ \log P(y_m|\omega_i) + \log P(\omega_i|\mathbf{x}) - \log p(\mathbf{y}), \qquad (3)$$

where $p(\mathbf{y}) = \sum_{i=1}^{n} p(y_1|\omega_i) \cdots p(y_k|\omega_i) \ P(y_{k+1}|\omega_i) \cdots (y_m|\omega_i) \ P(\omega_i|\mathbf{x})$.

This formulation has two main drawbacks. The first problem is that all the m soft biometric variables have been weighed equally. In practice, some soft biometric variables may contain more information than the others. For example, the height of a person may give more information about a person than gender. Therefore, we must introduce a weighting scheme for the soft biometric traits based on an index of distinctiveness and permanence, i.e., traits that have smaller variability and larger distinguishing capability will be given more weight in the computation of the final matching probabilities. Another potential pitfall is that any impostor can easily spoof the system because the soft characteristics have an equal say in the decision as the primary biometric trait. It is relatively easy to modify/hide one's soft biometric attributes by applying cosmetics and wearing other accessories (like mask, shoes with high heels, etc.). To avoid this problem, we assign smaller weights to the soft biometric traits compared to those assigned

to the primary biometric traits. This differential weighting also has another implicit advantage. Even if a soft biometric trait of a user is measured incorrectly (e.g., a male user is identified as a female), there is only a small reduction in that user's posteriori probability and the user is not immediately rejected. In this case, if the primary biometric produces a good match, the user may still be accepted. Only if several soft biometric traits do not match, there is significant reduction in the posteriori probability and the user could be possibly rejected. If the devices that measure the soft biometric traits are reasonably accurate, such a situation has very low probability of occurrence. The introduction of the weighting scheme results in the following discriminant function for user ω_i:

$$g_i(\mathbf{x}, \ \mathbf{y}) = a_0 \log P(\omega_i|\mathbf{x}) \ + \ a_1 \log p(y_1|\omega_i) + \ \cdots \ + \ a_k \log p(y_k|\omega_i) +$$

$$a_{k+1} \log P(y_{k+1}|\omega_i) \ + \ \cdots \ + \ a_m \log P(y_m|\omega_i), \tag{4}$$

where $\sum_{i=0}^{m} a_i = 1$ and $a_0 >> a_i$, $i = 1, 2, \cdots, m$. Note that a_i's, $i = 1, 2, \cdots, m$ are the weights assigned to the soft biometric traits and a_0 is the weight assigned to the primary biometric identifier. It must be noted that the weights a_i, $i = 1, 2, \cdots, m$ must be made small to prevent the domination of the primary biometric by the soft biometric traits. On the other hand, they must large enough so that the information content of the soft biometric traits is not lost. Hence, an optimum weighting scheme is required to maximize the performance gain.

4   Experimental Results

Our experiments demonstrate the benefits of utilizing the gender, ethnicity, and height information of the user in addition to the fingerprint. Our fingerprint database consisted of impressions of 160 users obtained using a Veridicom sensor. Each user provided four impressions of each of the four fingers, namely, the left index finger, the left middle finger, the right index finger, and the right middle finger. Of these 640 fingers, 263 were selected and assigned uniquely to the users in the face database described in [8]. Gender and ethnicity information of users were automatically extracted from their face images. Fingerprint matching was done using minutia features [9]. Two fingerprint impressions of each user were used as templates and the other two impressions were used for testing. The fingerprint matching score for a particular user was computed as the average of the scores obtained by matching the test impression against the two templates of that user. The separation of the fingerprint database into training and test sets, was repeated 20 times and the results reported are the average for the 20 trials.

The ethnicity classifier proposed in [8] was used in our experiments. This classifier identifies the ethnicity of a test user as either Asian or non-Asian with an accuracy of 96.3%. If a "reject" option is introduced, the probability of making an incorrect classification is reduced to less than 1%, at the expense of rejecting 20% of the test images. A gender classifier was built following the same methodology used in [8] for ethnicity classification. The accuracy of the gender classifier

without the "reject" option was 89.6% and the introduction of the "reject" option reduces the probability of making an incorrect classification to less than 2%. In cases where the ethnicity or the gender classifier cannot make a reliable decision, the corresponding information is not utilized for updating the matching score of the primary biometric system.

Since we did not have the height information about the users in the database, we randomly assigned a height 'H_i' to user ω_i, where H_i is drawn from a Gaussian distribution with mean 165 cm and standard deviation 15 cm. The height of a person can be measured during the recognition phase using a sequence of real-time images as described in [10]. However, the measured height will not be equal to the true height of the user stored in the database due to the errors in measurement and the variation in the user's height over time. Therefore, it is reasonable to assume that the measured height H_i^* will follow a Gaussian distribution with a mean H_i cm and a standard deviation of 5 cm.

Let $P(\omega_i|s)$ be the posterior probability that the test user is user ω_i given the fingerprint matching score 's' of the test user. Let $y_i = (G_i,\ E_i,\ H_i)$ be the soft biometric feature vector corresponding to the user ω_i, where G_i, E_i, and H_i are the true values of gender, ethnicity, and height of ω_i. Let $y^* = (G^*,\ E^*,\ H^*)$ be the observed soft biometric feature vector of the test user, where G^* is the observed gender, E^* is the observed ethnicity, and H^* is the observed height. Now the final score after considering the observed soft biometric characteristics is computed as

$$g_i(s,\ y^*) = a_0 \log P(\omega_i|s) + a_1 \log p(H^*|H_i) + a_2 \log P(G^*|G_i) + a_3 \log P(E^*|E_i)\ ,$$

where $a_2 = 0$ if $G^* =$ "reject", and $a_3 = 0$ if $E^* =$ "reject".

Figure 2(a) shows the Cumulative Match Characteristic (CMC) of the fingerprint biometric system operating in the identification mode, and the improvement in performance achieved after the utilization of soft biometric information. The weights assigned to the primary and soft biometric traits were selected intuitively such that the performance gain is maximized. However, no formal procedure was used and an exhaustive search of all possible sets of weights was not attempted. The use of ethnicity, gender, and height information along with the fingerprint leads to an improvement of approximately 5% over the primary biometric system. Figure 2(b) shows the Receiver Operating Characteristic (ROC) of a biometric system operating in the verification mode, using fingerprint as the primary biometric identifier and ethnicity, gender, and height as the soft biometric traits. An improvement of about 4% in the Genuine Acceptance Rate (GAR) can be observed over a wide range of values of False Acceptance Rate (FAR).

5   Conclusions

We have formulated a mathematical framework based on the Bayesian decision theory for integrating the soft biometric information with the output of the primary biometric system. We have demonstrated that the utilization of ancillary
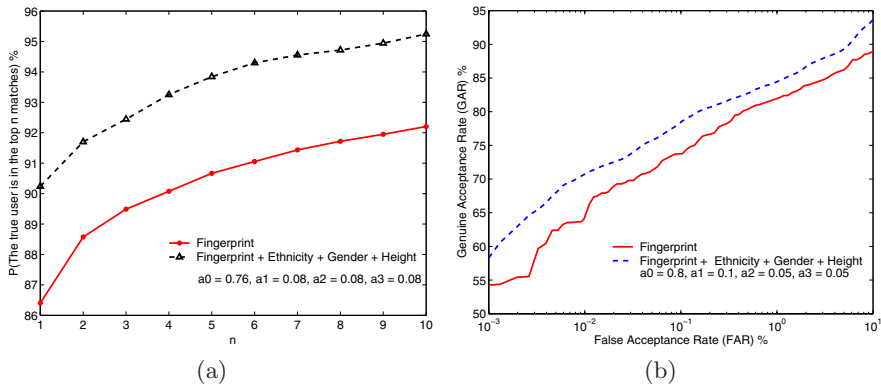
Fig. 2. Improvement in recognition performance of a fingerprint system after utilization of soft biometric traits (a) Identification mode (b) Verification mode

user information like gender, height, and ethnicity can improve the performance of the traditional biometric systems like fingerprint. Although these soft biometric characteristics are not as permanent and reliable as the traditional biometric identifiers like fingerprint, they provide some information about the identity of the user that leads to higher accuracy in establishing the user identity. However, an optimum weighting scheme based the discriminative abilities of the primary and the soft biometric traits is needed to achieve an improvement in recognition performance.

Our future research work will involve establishing a more formal procedure to determine the optimal set of weights for the soft characteristics based on their distinctiveness and permanence. Methods to incorporate time-varying soft biometric information such as age and weight into the soft biometric framework will be studied. The effectiveness of utilizing the soft biometric information for "indexing" and "filtering" of large biometric databases must be studied. Finally, more accurate mechanisms must be developed for automatic extraction of soft biometric traits.

References

[1] Jain, A. K., Bolle, R., Pankanti, S., eds.: Biometrics: Personal Identification in Networked Security. Kluwer Academic Publishers (1999) 731
[2] Hong, L., Jain, A. K., Pankanti, S.: Can Multibiometrics Improve Performance? In: Proceedings of IEEE Workshop on Automatic Identification Advanced Technologies, New Jersey, U. S. A. (1999) 59–64 731
[3] Bertillon, A.: Signaletic Instructions including the theory and practice of Anthropometrical Identification, R.W. McClaughry Translation. The Werner Company (1896) 732

 [4] Heckathorn, D. D., Broadhead, R. S., Sergeyev, B.: A Methodology for Reducing Respondent Duplication and Impersonation in Samples of Hidden Populations. In: Annual Meeting of the American Sociological Association, Toronto, Canada (1997) 732

 [5] Wayman, J. L.: Large-scale Civilian Biometric Systems - Issues and Feasibility. In: Proceedings of Card Tech / Secur Tech ID. (1997) 732

 [6] Givens, G., Beveridge, J. R., Draper, B. A., Bolme, D.: A Statistical Assessment of Subject Factors in the PCA Recognition of Human Subjects. In: Proceedings of CVPR Workshop: Statistical Analysis in Computer Vision. (2003) 733

 [7] Newham, E.: The Biometrics Report. SJB Services (1995) 733

 [8] Jain, A. K., Lu, X.: Ethnicity Identification from Face Images. In: Proceedings of SPIE International Symposium on Defense and Security : Biometric Technology for Human Identification (To appear). (2004) 735

 [9] Jain, A. K., Hong, L., Pankanti, S., Bolle, R.: An identity authentication system using fingerprints. Proceedings of the IEEE **85** (1997) 1365–1388 735

[10] Kim, J. S., et al.: Object Extraction for Superimposition and Height Measurement. In: Proceedings of Eighth Korea-Japan Joint Workshop on Frontiers of Computer Vision. (2002) 736

A New Approach to Personal Identification in Large Databases by Hierarchical Palmprint Coding with Multi-features

Jane You, Wai-Kin Kong, David Zhang, and King Hong Cheung

Department of Computing
The Hong Kong Polytechnic University, Kowloon, Hong Kong
{csyjia,cswkkong,csdzhang,cskhc}@comp.polyu.edu.hk

Abstract. This paper presents a new approach to personal identification using palmprints. To tackle the key issues such as feature extraction, representation, indexing, similarity measurement and fast search for the best match, we propose a hierarchical multi-feature coding scheme to facilitate coarse-to-fine matching for efficient and effective palmprint verification and identification in a large database. In contrast to the existing systems that employ a fixed mechanism for feature extraction and similarity measurement, we extract multiple features and adopt different matching criteria at different levels to achieve high performance by coarse-to-fine guided search. Our experimental results demonstrate the feasibility and effectiveness of the proposed method.

1 Introduction

Biometrics, which deals with identification of individuals based on their biological or behavioural characteristics, has been emerging as a new and effective identification technology to achieve accurate and reliable identification results. Palmprint is referred to principal lines, wrinkles and ridges on a palm. Like fingerprints, palmprint can be used as a powerful means in law enforcement for criminal identification because of its stability and uniqueness [7].

At a first glance, palmprint and fingerprint patterns appear to resemble each other in some ways. Both consist of a large amount of ridges. Although the minutiae based matching which utilizes terminations and bifurcations of the ridges is powerful for fingerprint verification and identification, such an approach is not suitable for palmprint patterns due to the change of orientations. Zhang and Shu [14] proposed to use datum point invariance and line feature matching for palmprint feature extraction and verification. Dute *et al.* investigate the feasibility of matching palmprints based on feature points [3]. However, these approaches are subject to certain limitations due to the lack of flexibility. To achieve flexibility and multiple feature integration, we propose a hierarchical palmprint coding scheme to facilitate coarse-to-fine matching for efficient and effective identifying a palmprint in large database. More specifically, we extract different palmprint features at different levels: Level-1 feature: global geometry based key point distance; Level–2 feature: global texture energy; Level–3 feature: fuzzy "interest" line and Level–4 feature: local directional texture energy vector for fine palmprint matching. We start with global geometry feature to localize

the region of interest of palmprint sample at coarse level and apply a distance measurement of palm boundary to guide the dynamic selection of a small set of similar candidates from the database for further processing. We also use the global texture energy (GTE) for fast search for the best match. Such a mask-based texture feature representation is characterized with high convergence of inner-palm similarities and good dispersion of inter-palm discrimination. We then adopt fuzzy set theory to detect "interest" feature lines to guide the search for the best match at fine level. Finally, we apply local texture measurement to establish a feature vector for palmprint matching.

2   Hierarchical Palmprint Coding

It is very difficult, if not impossible, to use one feature model for palmprint matching with high performance in terms of accuracy, efficiency and robustness. To speed up the search process for the best match with reliable features and flexible matching criteria, we adopt multi-level feature extraction and flexible similarity measurement. Instead of using a fixed feature extraction mechanism and a single matching criterion as in [3], [14], we apply a hierarchical coding scheme to extract multiple palmprint features at both global and local levels. Fig. 1 illustrates the system structure of our approach and our proposed algorithms for multiple feature extraction are summarized as follows.

2.1 Level-1 Global Geometry Feature: Key Point Distance

The key point distance is measured based on boundary segments between fingers. To obtain a stable palmprint image for reliable feature extraction, six pegs on the platform of the palmprint scanner are installed to serve as control points for the placement of the user's hands. In addition, a coordinate system is defined to align different palmprint images for feature measurement.

2.2 Level-2 Global Texture Feature: "Tuned" Mask Based Texture Energy

For the purpose of fast selection of a small set of similar palmprint patterns from the database, previously we applied four "tuned" masks to capture the global palmprint texture features which are more sensitive to horizontal lines, vertical lines, 45° lines and -45° lines respectively [2]. In this paper, we apply this approach to extract the global texture feature for Level-2 feature representation.

2.3 Level-3 Fuzzy "Interest" Lines

The so-called "interest" lines refer to the dominant feature lines such as principal lines, wrinkles, etc. in palmprint. The major steps of extracting dominant lines are described in [10].

2.4 Level-4 Local Directional Texture Energy

This level is the last level to make final decision so high accuracy of this level is expected. To achieve this goal, we generate a long feature vector from local directional

texture energy so as to facilitate more information. The description of this local feature is given in [10].

3 Multiple Similarity Measurement

3.1 Minimum Distance for Global Feature Matching

We have defined two global features, 1) geometry distance of key points and 2) global texture energy. Our goal is to search a palmprint in large database at high speed by using simple but effective distance measures for matching. The distance measure for the geometry feature vectors is

$$D_1 = |d_i - d_j| \tag{1}$$

where i and j represent two palmprint images and d_i (d_j) is the distance between key points of the i (j) palmprint images.

The global texture energy can be measured by the similar distance measure. Let the global texture energy vector of ith palmprint image be [v_{0i}, v_{1i}, v_{2i}, v_{3i}], similarly for j^{th} palmprint image. Their similarity can be measured by,

$$D_2 = \sum_{k=0}^{3} |v_{ki} - v_{kj}| \tag{2}$$
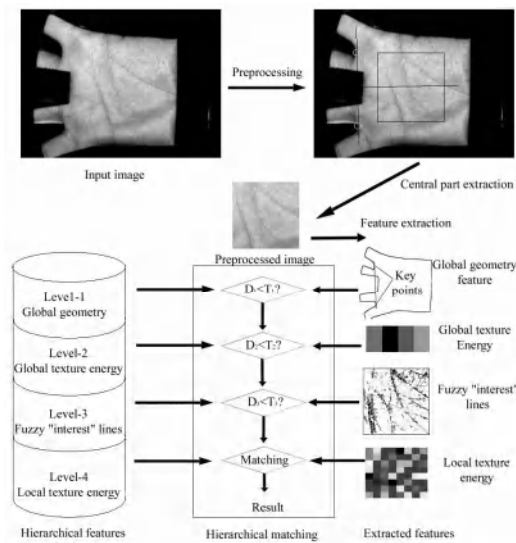
Fig. 1. System structure of the hierarchical palmprint system

(a)   (b)   (c)
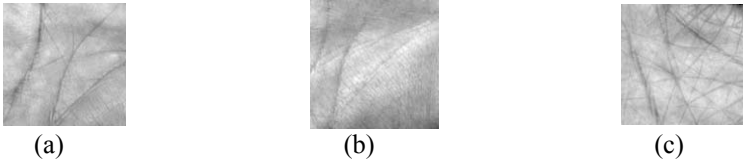
Fig. 2. Samples of different palmprint patterns with distinctive texture features. (a) strong principal lines, (b) less wrinkles and (c) strong wrinkles
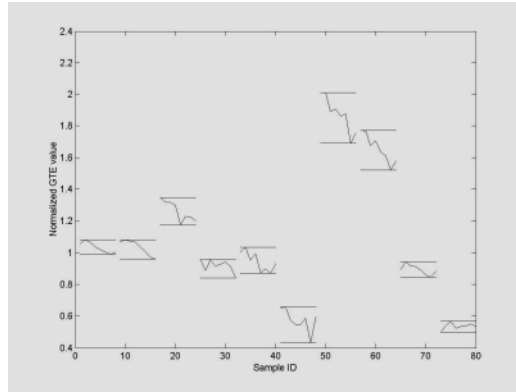
Fig. 3. Comparison of palmprint GTE distribution: inter-palm dispersion vs. inner-palm convergence

3.2 Minimum Distance for Local Feature Matching

In addition to the two global features, we also design two local features in our coding scheme, "interest" lines and local directional texture energy. The "interest" lines are represented by a 64 dimensions feature vector. At this level, we use an angular distance to evaluate the difference between two feature vectors. Let X and Y be two "interest" lines. The angular distance is defined as

$$D_3 = \frac{X^T Y}{\|X\|\|Y\|} \tag{3}$$

In the local directional texture energy, we use a local angular distance to evaluate the difference between two feature vectors. For simplicity, let the local directional texture energy vector of ith palmprint image be $[y_{0i}, y_{1i}, y_{2i}, y_{3i} ... y_{64i}]$, where $y_{ki} = [u_0, u_1, u_2, u_3]$. The local angular distance is defined as:

$$D_4 = \frac{1}{64} \sum_{k=1}^{64} \frac{y_{ki} y_{kj}^T}{\|y_{ki}\|\|y_{kj}\|} \tag{4}$$

4   Hierarchical Palmprint Matching

To avoid the blind search for the best fit between the template pattern and all of the sample patterns stored in an image database, a guided search strategy is essential to reduce computation burden. We begin initial search for the best similar palmprint matching sub-set with Level-1 global geometry feature. Our similarity measurement method is based on the comparison of key point distance with respect to its length of different samples. The candidates with small distance difference such as $D_1 < T_1$, will be considered for further coarse-level selection by global texture energy. The selected candidates will be subject to fine matching based on "interest" line and comparison of local directional texture energy.

4.1 Coarse Level Similar Palmprint Pattern Classification

This classification task can be viewed as a decision-making process that allocates an input palmprint sample to those palmprint images with the similar measurements in the database. The feature measurements of palmprint samples with distinct texture primitives should exhibit large variances while the measurements of the similar patterns should possess very small diversity. Thus, such a global feature is characterized with high convergence of inner-palm similarities and good dispersion of inter-palm discrimination. Fig. 2 shows three palmprint samples from the different individuals with distinctive texture features and Fig. 3 demonstrates the distribution of global palmprint texture energy measurements.

Although different individuals have different palmprint patterns, some of these patterns are so similar that it is very difficult, if not impossible, to classify them based on the global texture features only. To tackle such a problem, we propose a dynamic selection scheme to obtain a small set of the most similar candidates in the database for further identification by image matching at fine level. The idea behind this is to eliminate those candidates with large difference of Level-1 global geometry feature and generate a list of the very similar candidates with very small difference of their key point distance. The candidates in the list will undergo further selection in terms of its Level-2 global texture energy (GTE). Only those samples that remain very close GTEs (Global Texture Energy) will be considered for fine-level matching.

4.2 Fine Level Multi-step Image Matching

The proposed fine matching algorithm starts with simple distance measure for "interest" line. If the matching score D_3 is larger than the threshold T_3, controlling the false acceptance and false rejection rates at this level, the palmprint images will go through the final fine-matching in terms of their local directional texture energy. The final matching is based on the comparison of local feature vectors in terms of their local angular distance. The best match is the candidate with the least distance.

5   Experimental Results

The palmprint image samples used for the testing are the size of 384×284 with the resolution of 75 dpi and 256 grayscales. In our palmprint image database, 7,752 palmprint images from 386 different palms are stored. The dynamic selection of image features is demonstrated by multi-level palmprint feature extraction for personal identification and verification (see our work on palmprint verification [10]). The experiment is carried out in two stages. In stage one, the global palmprint features are extracted at coarse level and candidate samples are selected for further processing. In stage two, the regional palmprint features are detected and a hierarchical image matching is performed for the final retrieval.

   To evaluate the verification accuracy of our hierarchical palmprint system, we use three parameters, T_1, T_2 and T_3, which control the false acceptance and false rejection rate of the first three levels. Three sets of parameters, T_a, T_b and T_c are tested. Table 1 lists these parameters and the corresponding false rejection and correct rejection rates of the first three levels are given in Table 2.

   The proposed system is implemented by using Visual C++ 6.0 on an embedded Intel Pentium III processor (500MHz) PC. Table 3 lists the time required at each level and the total execution time is about 0.7 second, which is fast enough for real-time verification.

Table 1. The selection of parameters T_1, T_2, T_3 and T_a T_b, T_c

	T_a	T_b	T_c
T_1	20	23	25
T_2	0.0194	0.0216	0.0250
T_3	0.764	0.73	0.68

Table 2. Comparison of system performance

	False Rejection Rate	Correct Rejection Rate
T_a	6.13%	88.23%
T_b	3.89%	78.19%
T_c	1.98%	60.51%

Table 3. System execution time

Operations	Times (ms)
Preprocessing	538
Feature Extraction	215
Level 1 matching	4.7×10^{-5}
Level 2 matching	3.4×10^{-4}
Level 3 matching	0.009
Level 4 matching	0.059

6   Conclusions

Palmprint feature extraction and matching are two key issues in palmprint identification and verification. We conclude that the combination of four-level features including Level-1 global geometry feature, Level-2 global texture energy, Level-3 fuzzy "interest" line and Level-4 local texture feature possesses a large variance between different classes while remaining high compactness within the class. The experimental results provide the basis for the further development of a fully automated palmprint-based security system with high performance in terms of effectiveness, accuracy, robustness and efficiency.

Acknowledgement

References

[1]   Baltscheffsky, P. and Anderson, P., "The palmprint project: Automatic identity verification by hand geometry", Proc. 1986 International Carnahan Conference on Security Technology, Gothenburg, Sweden (1986) 228-235

[2]   Benke, K.K., Skinner, D.R. and Woodruff, C.J., "Convolution operators as a basis for objective correlates for texture perception", IEEE Trans. Syst. Man, Cybern.*, Vol. 18 (1988) 158-163

[3]   Duta, N., Jain, A.K. and Mardia, K.V., "Matching of palmprint", Pattern Recognition Letters, Vol. 23(4) (2001) 477-485

[4]   Han, C.C., Cheng, H.L., Fan, K.C. and Lin, C.L., "Personal authentication using palmprint features", Pattern Recognition, Vol. 36(2) (2002) 71-382

[5]   Jain, A., Bolle R. and Pankanti, S., Biometrics: Personal Identification in Networked Society, Kluwer Academic Publishers (1999)

[6]   Miller, B., "Vital signs of identity", IEEE Spectrum, Vol. 32(2) (1994) 22-30

[7]   Sanchez-Reillo, R., Sanchez-Avilla, C. and Gonzalez-Marcos, A., "Biometric identification through hand geometry measurements", IEEE Trans. Pattern Anal., Machine Intell., Vol. 22(10) (2000) 1168-1171

[8]   You, J. and Bhattacharya, P., "A Wavelet-based coarse-to-fine image matching scheme in a parallel virtual machine environment", IEEE Trans. Image Processing, Vol. 9(9) (2000) 1547-1559

[9]   You, J. and Cohen, H.A., "Classification and segmentation of rotated and scaled textured images using texture 'tuned' masks", Pattern Recognition, Vol. 26, pp. 245 - 258, 1993.

[10]   You, J., Li, W. and Zhang, D., "Hierarchical palmprint identification via multiple feature extraction", Pattern Recognition, Vol. 35(4) (2002) 847-859

[11]   Zhang, D., Automated Biometrics – Technologies and Systems, Kluwer Academic Publishers, (2000)

[12]   Zhang, D., Biometrics Resolutions for Authentication in An e-World, Kluwer Academic Publishers (2002)

[13]   Zhang, D., Kong, W.K., You, J. and Wong, M., "On-line palmprint identification", IEEE Trans. Pattern Anal. and Machine Intell. Vol. 25(9) (2003) 1041-1050

[14]   Zhang, D. and Shu, W., "Two novel characteristics in palmprint verification: datum point invariance and line feature matching", Pattern Recognition, Vol. 32(4) (1999) 691-702

Watermarking of MPEG-4 Videos

Abhinav Gupta and Phalguni Gupta

Department of Computer Science and Engineering
Indian Institute of Technology, Kanpur, India - 208016
{abhigupt,pg}@cse.iitk.ac.in

Abstract. A MPEG-4 compressed domain video watermarking method is proposed and its performance is studied at video bit rates ranging from 64 Kb/s to 900 Kb/s. The watermark is inserted by modifying Discrete Cosine Transformation (DCT) coefficients. The strength of watermark is changed according to local frame characteristics to reduce impact on visual quality. The algorithm's performance is also studied for watermarking bits in a frame ranging from 1 Kb/frame to 3 Kb/frame. The watermark is attack-free against attacks like scaling, rotation and cropping even if blind-techniques are used.

1 Introduction

Digital Watermarking is imperceptible insertion of information into multimedia data which act as a signature in the video. The classical approach to watermarking of a compressed video stream is to decompress the video and use a spatial-domain or transform-domain watermarking technique and then recompress the watermarked video. Some of the disadvantages of this approach are: noise addition due to recompression, bad quality due to no knowledge of compression parameters results and computational complexity. Another approach is to insert the watermark in compressed domain only. In this approach the watermark is inserted in some syntactical elements like DCT coefficients of a partially decoded video.

Hartung [1, 2] describes techniques to embed a spread-spectrum watermark into MPEG-2 [3] compressed videos as well as into uncompressed video. For compressed domain watermarking they decode the video to obtain the DCT coefficients of each frame and insert the watermark by modifying those DCT coefficients. Langelaar[4] describes a compressed domain watermarking technique called Differential Energy Watermark (DEW) in which the video is partitioned into groups of blocks each of which is further divided into two sets of equal size as determined by the watermark embedding key. By comparing the energy of selected DCT coefficients within two sets, a single payload bit is expressed. Jordan et al. [5] proposed a method for the watermarking of compressed video that embeds information in the motion vectors. Hsu and Wu present a method for watermarking [6] which modifies middle frequency DCT coefficients in relation to spatially and temporally neighbouring blocks. Nicholson [7] evaluated the watermark robustness and video quality after video is watermarked and compressed

with MPEG-4. However none of these techniques addressed direct watermarking of these MPEG-4 videos.

In this paper, a new compressed domain watermarking technique for MPEG-4 [8] video streams is presented. The approach is similar to [1] as watermark is inserted in DCT coefficients. However use of synchronization templates makes it robust and local gain method improves the quality.

In Sect. 2 an overview of MPEG-4 technique is presented. This is followed by the proposed method and results in Sect. 3 and Sect. 4 respectively. Section 5 proposes a protocol for subjective evaluation of the system and the paper concludes in Sect. 6.

2   Overview of MPEG-4

MPEG-4 [8] encodes the visual information as objects (natural, synthetic video and still textures). MPEG-4 encodes a description of the scene for proper rendering of all objects. An MPEG-4 visual scene may consist of one or more video objects. Each video object is characterized by temporal and spatial information in the form of shape, motion and texture and corresponds to a 2D object in the scene. A Video Object Plane(VOP) is a time sample of video object. VOP's can be encoded independent of each other or dependent on each other by motion compensation. A VOP contains the encoded video data in form of macroblocks. A macroblock contains a section of the luminance component and the spatially subsampled chrominance components. In MPEG-4 visual standard there is support for only one chrominance format for a macroblock, the 4:2:0 format. In this format, each macroblock contains 4 luminance blocks, and 2 chrominance blocks. Each block contains 8x8 pixels encoded using DCT transformation. The DCT coefficients are then adaptively quantized to achieve low bit rates.

3   Proposed Method

In the proposed method, a watermark signal is inserted directly into MPEG-4 compressed bit-stream while detection is performed using compressed bit-stream without any watermark signal. Section 3.1 discusses the formation of watermark signal from the original message signal and synchronization templates. Section 3.2 addresses the process by which the watermark signal is embedded in the MPEG-4 videos. Section 3.3 discusses the local adaptive gain method to increase quality of the videos.

3.1   Spread Spectrum Watermark Signal

The watermark signal is often limited to a small value to ensure the imperceptibility and subject to interference from the host signal and additional noise arising from subsequent processing. A spread spectrum signal is vulnerable to synchronization errors occurring after scaling, cropping and rotation. A pair of

templates is imposed on the spread spectrum signal to combat synchronization losses. The first template restricts the watermark signal to have a regular periodic structure. In particular, the watermark signal $w(x, y)$ is constructed by repeating an elementary watermark tile $\hat{w}(x, y)$ in a non-overlapping fashion. If the tiling is done properly the peak always occur at the centre of each tile. If a linear transformation A is applied to watermarked VOP, the autocorrelation coefficients $h(x, y)$, thus the new peaks move to x' and y' according to

$$[x'\ y']^T = A[x\ y]^T \tag{1}$$

The second synchronization template forces $w(x, y)$ to contain a constellation of peaks in frequency domain. This requirement is met by making $\hat{w}(x, y)$ as combination of the message bearing symbol $m(x, y)$ and the synchronization signal $g(x, y)$. In frequency domain this $g(x, y)$ contains peaks in mid frequency band each peak occupying one frequency coefficient and having unity magnitude. After the geometrical transformation A applied to image the FFT coefficient $F(u, v)$ move to a new location (u', v') according to the equation

$$[u'\ v']^T = (A^T)^{-1}[u\ v]^T \tag{2}$$

Thus using the two equations above we can obtain the linear transformation A applied to the image and hence revert the transformation to perform the detection method. Figure 1(a) outlines the process of creation of watermark signal.

3.2   Watermark Embedding

This section describes embedding the watermark directly to the bit-stream generated in accordance with the Advanced Simple Profile (ASP) of the MPEG-4 standard. The watermark signal $w(x, y)$ is added to the luminance plane of the VOPs. Since the DCT is a linear transform adding the transformed watermark signal directly to DCT coefficients of the luminance blocks is equivalent to addition in spatial domain. An elementary bit-stream is parsed down to the block level and variable length coded motion vector and DCT coefficients are obtained. Motion vectors are reconstructed using VLC decoding and reversing the prediction steps wherever applicable. After the watermark signal is embedded, VLC codes are regenerated and bit-stream is reconstructed. Fig. 1(b) depicts the whole outline.

Since a 96x96 signal is embedded in a 192x192 image, a 8x8 block is embedded in 16x16 block. Hence, the total number of permutations possible are P_{64}^{256}. This property helps to introduce the Digital License Number. Each authorized user has a key that maps to one of the configuration of all possible permutations. Quadratic Chaining is a method where iteration over $(i + i^2)K mod 256$, where K is the license number, yields 64 places to insert watermark. Hash Table is another approach to do the same.
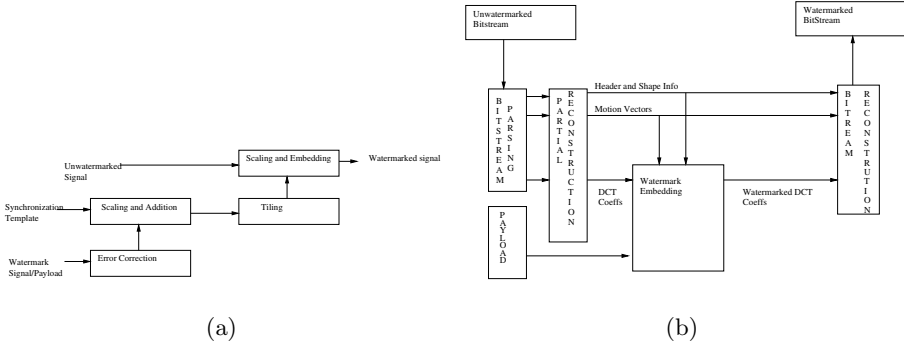
Fig. 1. a) Schematic Diagram for preparation of watermark signal. b)Outline for a watermark embedder. The embedder mimics the MPEG-4 decoder

3.3 Adaptive Local Gain Methodology

The adaptive local gain methodology improves the performance of the watermark. For relatively smooth regions of the video, where even a small amount of distortion may be visible , the local gain control reduces the watermark embedding power to minimize the watermark perceptibility. For relatively busy or textured regions of the image, the local gain control increases the embedding power for improved robustness. The local gain method uses a local activity measure to adjust the watermark power on a block by block basis, which is obtained directly from DCT coefficients for intra-blocks and predicted using motion vector information for predicted blocks. The gain model outputs a local gain $L(x, y)$. The watermark coefficients are then weighted by $L(x, y)$ to produce the watermark signal that will be embedded into video:

$$W^*(x, y) = \alpha L(x, y) W(x, y) \tag{3}$$

where W* is the watermark that will be embedded, α is user selected global gain and W is the watermark signal prior to gain adjustment. For each VOP, local gain weights are decided based on the estimated activity in the VOP. For Intracoded-VOP's the $L(x, y)$ is

$$L(x, y) = DCT(f(x), g(y))^2 \Sigma DCT(i, j)^2 \tag{4}$$

where $f(x)$ and $g(y)$ map the $(x, y)th$ pixel to $(f(x), g(y))$ DCT coefficient. For predicted VOPs we used the same formula but instead of Total Energy being $\Sigma DCT(i, j)^2$ it is taken to be

$$A_i = \frac{N1}{N} A_A + \frac{N2}{N} A_B + \frac{N3}{N} A_C + \frac{N4}{N} A_D \tag{5}$$

where N1,N2,N3 and N4 are the number of pixels that moved from Area A,B,C,D to the block currently being considered. The sketch for local adaptive gain is shown in Fig. 2(a).

(a)   (b)   (c)
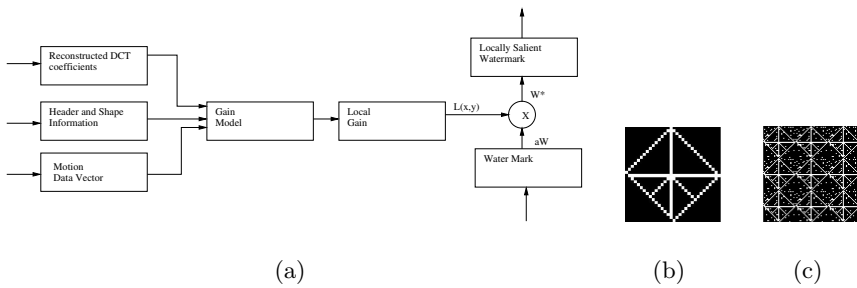
Fig. 2. a) The sketch for Local Adaptive Gain Method. b) The Original watermark signal that is to be inserted. c) The final watermark signal after tiling

(a) Video I(900 Kb/sec)   (b) Video II(64 Kb/sec)
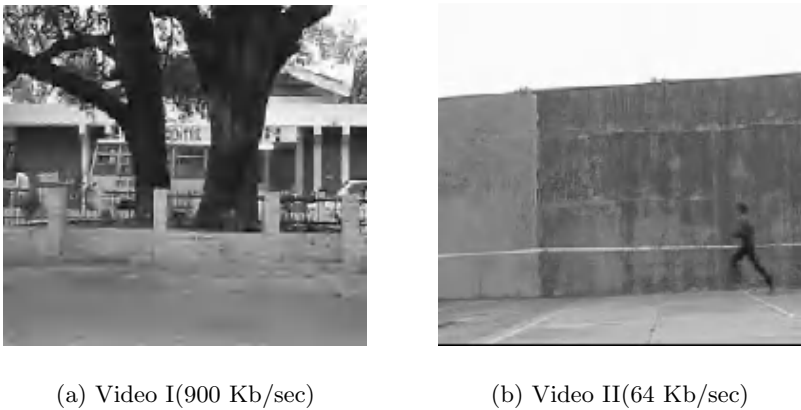
Fig. 3. a) Watermarked Video of bitrates 900 Kb/sec and payload bit-rate of 1 Kb/frame. b) Watermarked Video of bitrates 64 Kb/sec and payload bit-rate of 1 Kb/frame

4   Results

The algorithm has been tried with the two videos (one having lot of texture compared to other) taken in IIT Kanpur as there is no standard video dataset for watermarking approaches. The videos were compressed at different bit rates (64-900 Kb/sec) and the watermark signal bit rates were varied from 1 kb/frame to 3 Kb/frame. The frames were 192 x 192 in dimensions and a watermark of 32 x 32 was inserted. The preparation of watermark signal has been shown in figures. 2(b), 2(c). The results of watermark embedding at 1 Kb/frame and 900 kb/s and 64 kb/s videos are shown in Fig. 3. Figure 4 shows the performance of local adaptive gain module.

(a) No Local Adaptive Gain   (b) Local Adaptive Gain
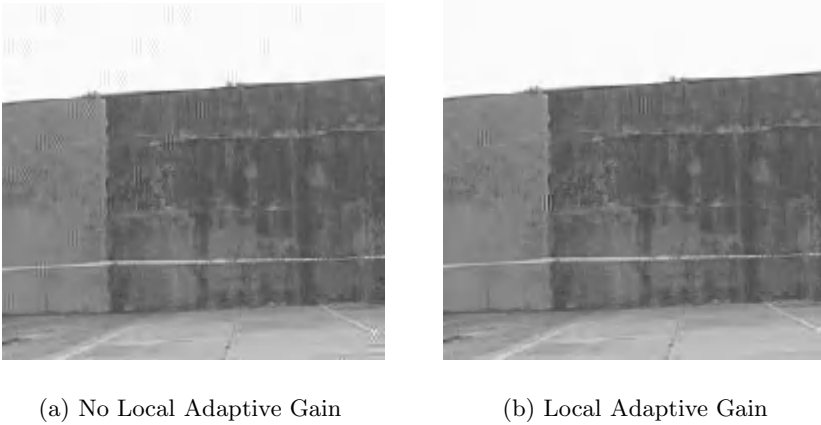
Fig. 4. a) Watermarked Video of bitrates 900 Kb/sec and Watermark payload rate of 3Kb/sec and without the Local adaptive gain module. b) Watermarked Video of bitrates 900 Kb/sec and Watermark payload rate of 3Kb/sec with Local adaptive gain applied. The difference is quality is quite visible in low texture regions like the sky and the wall

5   Subjective Evaluations

A protocol for evaluating the quality of watermarked video frames has been defined based on ITU-T recommendation P.910 [9] and ITU-R Rec. BT.500 [10]. These recommendations suggest different protocols based on the goal of the evaluations and availability of ground truth. In Absolute Category Rating (ACR) protocol, the images are shown and then the questions related to its quality are asked. The scale is discrete with values 1-5. In Degradation Category Rating (DCR)protocol, first the source is presented followed by system under test. The scale use is discrete impairment scale.

The ACR, DCR scales are not followed because they used discrete scales and hence take away the freedom to differentiate from the use. A protocol was designed in which the subject was shown the ground truth and system under test but the scale is continuous. In each test session the user was first made familiar with the interfaces. This was followed by a mock test session to check the familiarity. Then we had a test session which started with some stabilizing sequences. The scores were than normalized and mean scores were computed.

The subjective test was done on 5 subjects who were end-users and not experts in Image-processing. The scale on which the subjects voted was continuous. In the first session of the test the underlying video had a bit rate of 900 Kb/s. In second session the bit rates were changed and individual frames were shown. The watermark rate remained 1 Kb/frame. The following graphs (fig 5(a) and fig. 5(b)) show the performance of the two videos.

(a) Payload Bitrate Changed   (b) Video Bitrates Changed
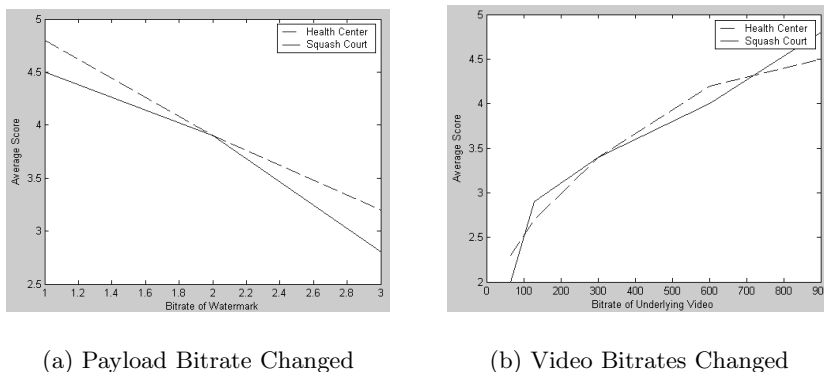
Fig. 5. a) The Subjective evaluations of the videos with payload bit-rate changed. b) The Subjective evaluations of the videos with video bitrates changed

6   Conclusion

A new technique for watermarking MPEG-4 bit-streams in compressed domain is proposed. The technique is not only robust to synchronization errors but also improves the quality of videos using local adaptive gain technique.

References

[1] Hartung, F., Girod, B.: Watermarking of uncompressed and compressed video. Signal Processing **66** (1998) 283–301
[2] Hartung, F.: Watermarking and fingerprinting of uncompressed and compressed video. Signal Processing **66** (1998) 283–301
[3] ISO: Information technology - generic coding of moving pictures and associated audio information. ISO/IEC 13818-2 (1994)
[4] Langelaar, G., R.Lagendijk: Optimal differential energy watermarking of dct encoded images and video. IEEE Transactions on image Processing **10** (2001) 148–158
[5] F. Jordan, M. K., Ebrahimi, T.: Proposal of a watermarking technique for hiding/retrieving data in compressed and decompressed video. ISO/IEC document JTC1/SC29/WG11 MPEG97/M2281 (1997)
[6] Hsu, C., Wu, J.: Hidden digital watermarks in images. IEEE Transactions on Image Processing **8** (1999) 58–68
[7] D. Nicholson, P. K., Delaigle, J.: Watermarking in mpeg4 context. European Conference on Multimedia Application Services and Techniques (1999) 472–492
[8] ISO: Information technology - coding of audio/video objects:video. ISO/IEC 14486-2 (October 1998)
[9] ITU-T: Methodology for subjective assesment of the quality of television pictures. Recommendation P.910 (1996)
[10] ITU-R: Subjective video quality assesment methods for multimedia applications. Recommendation BT. 500-11 (1995)

Hand Recognition Using Geometric Classifiers[*]

Yaroslav Bulatov[1], Sachin Jambawalikar[2], Piyush Kumar[3], and
Saurabh Sethia[1]

[1] Department of Computer Science, Oregon State University, Corvallis
OR 97331-3202, USA
{bulatov,saurabh}@cs.orst.edu
[2] Department of Biomedical Engineering
State University of New York at Stony Brook
NY 11794-8181, USA
sjambawa@ic.sunysb.edu
[3] Department of Computer Science, State University of New York at Stony Brook
NY 11794-8181, USA
piyush@acm.org

Abstract. We discuss the issues and challenges in the design of a hand outline based recognition system. Our system is easier to use, cheaper to build and more accurate than previous systems. Extensive tests on more than 700 images collected from 70 people are reported. Classification, verification and identification of the input images were done using two simple geometric classifiers. We describe a novel minimum enclosing ball classifier which performs well for hand recognition and could be of interest for other applications.

1 Introduction

Biometric recognition systems find applications in security systems of varying requirements. While finger printing and iris based systems work well for high security applications, they are not as suitable for medium and low security applications because of privacy concerns. Most users are not comfortable providing an identifying biometric signature for access to low risk facilities. Hand Geometry based verification systems find more acceptance because hand geometry is not considered distinctive enough to establish a positive identity. Yet hand geometry differs enough from person to person that it is sufficient for medium security applications.

Hand geometry recognition systems may provide three kinds of services. Verification, classification and identification. For verification the user provides his identity along with the hand geometry and the system verifies his identity. Verification can be used alone or in conjunction with other security systems where the user can enter his identity. For example, secure web-access can use hand geometry to verify a user [15]. For classification the user does not provide any

---

[*] A more complete version of this paper is available at
http://www.compgeom.com/hands/

identity information but is known to be legitimate. The system classifies him assuming his presence in the user database. This can be used in applications where the users can be trusted. For example, use of photocopying facilities in a university can be regulated using a hand based recognizer instead of a password. There is minimal risk of intruders because most such facilities are secured by other means. For identification the user does not provide any identity information other than the hand geometry and may be an intruder. The system tries to identify the individual or deny access. This is required in systems where a user will not be willing to go through the hassle of entering his identity. For example, a restricted dining facility or health club may install a hand geometry based identification system to keep track of its users.

Previous Work: The idea of using hand geometry for verification is not new. Most of the early work is in the form of patents. Recently there has been some documented research in this area. Arun Ross developed a hand geometry based verification system [15] (see also [9, 10]) using hand images obtained from a digital camera and pegs to guide palcement of hand. Jain and Duta [8] and Raul Sanchez-Reillo et al. [16] report on experiments with another way of doing verification. They try to align finger contours and measure the mean alignment error between them. Öden et al. [13] report on a system for classification and verification using implicit polynomials. They fit fourth degree implicit polynomials in each finger and compare them using algebraic invariants. Recognition Systems Inc's HandKey II is a commercial biometric hand recognition system [14]. It is based on a patent by Sidlauskas [17] which describes a hand verification system based on a 3D image of the hand. Another patent by Faulkner [7] describes a biometric measuring apparatus for verifying a user based on measurements performed on the user's hand using a 3D view. Also related is work by Duta et al. [6] and Li et al. [12] on palm print based verification systems.

2   Data Collection

All previous experimental work on hand recognition has been on hand images obtained from a set up that includes a digital camera. We however collected data using a document scanner. Thus no special set up was required (For example pegs). Also the users were free to keep their hands anywhere on the scanner. The only instruction given to the users was to keep their fingers separated and have their hand roughly vertical on the scanner surface. Users were free to stretch their hands to whatever level they felt comfortable. In fact some of the users were encouraged to stretch their hands to different extents for different scans so as to generate difficult data. Our system is thus more tolerant of user inconsistencies. In all we took around 10 scans of the right hand of 70 people to obtain a total of 714 images. The images were scanned at 90 dpi using a HP Deskscan scanner. There were a number of images where the hand was not placed flat on the scanner or one of the fingers got cut off by the scanner edge. However we kept all such images as part of our experiments.

3   Feature Extraction

In this section we describe our feature extraction algorithms. We extract 30 different features form each image. The first step is to convert the colored image into a binary image. This is done as follows. The image is first converted to CIELAB color model and thresholded using a fixed value in the B channel. From these binary images the hand contours were extracted using the following algorithm. First the boundary pixels were identified as those which had less than four neighbors. Then we applied hit and miss transform to remove dead ends and cycles in the boundary. After that we found the right most point and traverse the resulting boundary in counter-clockwise order. Thirty different features were then extracted from each hand. The features are illustrated in figure **??**.

Our feature extraction algorithms relied on detecting tips of the fingers and centers of the valleys between the fingers correctly. Here's how this was accomplished. For each point q in the hand outline consider the points p and r at a certain distance (about half the typical finger length) along the outline clockwise and counterclockwise. Consider the angle pqr. This angle is the lowest at the tips and highest at the valleys. We plotted a graph of these values, one for every point on the outline, smoothed it to get rid of some noise and detected peaks and valleys. Peaks represented finger tips, and valleys represented dips between fingers. There were five well defined peaks roughly equally spaced and some other smaller peaks created by the noise at the bottom of the hand. We pick the five highest peaks and mark the first one to be the thumb, second to be the index finger, and so on.

Once we had positions of finger tips and valleys between fingers, computing features was simple. Length was measured to be the distance from the tip to the midpoint between two valleys around that finger. Width was measured as the distance between two valleys. Area was measured as the area of the finger after being cut off by the line connecting two surrounding valleys. Perimeter was the length traversed along the finger.

We used Euclidean Distance Transform (EDT) [4, 11] to find greatest inscribed circles. The greatest inscribed circles for the fingers were found by cutting the finger off, and running EDT on that finger separately. We cut four of the fingers further into two pieces (top half and bottom half), and fit the circle into each of those parts.

4   Nearest Box Classifier

In this section we describe a simple classifier that we use to implement hand geometry verification, classification, and identification. For each person we pick a small number (3 to 5) of hand images as training set. We find the bounding box of each training set in the 30 dimensional feature space. For a query vector, the distance to these bounding boxes in L_∞ metric is used as a measure similarity. The distance along each feature axis is normalized using the maximum difference variability observed for that feature in the training set. Thus a feature that

varies a lot gets its effect diminished. For verification, we use a experimentally determined threshold ϵ to decide whether a query feature vector is close enough to a given training set. For classification we simply classify the query point to the set with the nearest bounding box. Identification is implemented by classifying the query point assuming it to be legitimate followed by verification.

5   Minimum Enclosing Ball Classifier

In this section we describe a novel classifier which performs better than the nearest box classifier and is also of general interest for other classification problems.

Suppose we are given N d-dimensional features $x_1, x_2, ... x_N$ from c classes $D_1, D_2, ... D_c$. Given a new feature vector v we want to classify it in one of the classes.

Our classifier first maps all x_i to a higher dimensional space using a Gaussian kernel, employs a novel way to approximate minimum enclosing balls in the mapped space of each D_i and then uses the Voronoi diagram of the centers for classification. Note that instead of using the convex hulls of each D_i for separation (as is done in standard SVMs) we use the voronoi cell of the centers of minimum enclosing balls for classification purposes.

The first step in the classification is the use of the Gaussian kernel. All points in the feature space are implicitly mapped to a high dimensional space by using the Gaussian kernel, $K(x_i, x_j) = <\phi(x_i), \phi(x_j)> = e^{-\frac{1}{\theta} || x_i - x_j ||^2}$ where $\phi : \mathcal{R}^d \to \mathcal{F}$ is a nonlinear map that takes the data into a higher dimensional space $\mathcal{F}$, that is also a dot product space. Here $<,>$ denotes the usual dot product of two vectors. In the description of the whole algorithm we will not need to compute either ϕ or $\mathcal{F}$ explicitly. All we would need is the definition $K(x_i, x_j) = e^{-\frac{1}{\theta} || x_i - x_j ||^2}$ for our classifier which can be computed in $O(d)$ time in $\mathcal{R}^d$ [3]. In the absence of information about the probability distribution of data, there is little theoretical justification of one window width over another for the Gaussian kernel. Nevertheless the classifier seems to work well in practice for a large range of window widths (we selected $\theta = 125$ after normalizing each feature to standard normal distribution and then scaling the features to lie between 0 and 100[1]). The approximate method of computing minimum enclosing balls used in the classifier can also be used for support vector clustering, is much easier to code than the current Lagrange multiplier based methods [3, 18] and is conceptually simpler (see algorithm 1). One drawback of this method is that we don't know how to take care of outliers efficiently in practice, although theoretically this is known [5]. In practice, we observed that the classifier works for a few outliers.

In $\mathcal{F}$, we compute an implicit representation of the approximate center of the minimum enclosing ball for each class D_i. We now classify a point p to a class if $\phi(p)$ lies in the Voronoi cell of the approximate center calculated in the space $\mathcal{F}$.

---

[1] For scaled features, there is a wide choice of θ for which the classifier performs well.

We first look at the new approximation algorithm we designed for computing the minimum enclosing ball (MEB) of a set of points in $\mathcal{F}$ (See algorithm 2). Algorithm 2 is an extension of algorithm 1 for kernel spaces. Conceptually the algorithm is very simple. It maintains a vector $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, ...]$, in which it implicitly maintains the center $c_i = \sum_{j=1}^{\chi} \lambda_j \phi(p_j)$. Here χ is the core set size [2]. In each iteration it finds the farthest point from the current center and updates the $\lambda_i's$ to reflect the movement of the center. The center in algorithm 2 moves exactly the way it moves in algorithm 1.

It was proven in [1] that for Euclidean spaces, if $iter = \frac{1}{\epsilon^2}$ then algorithm 1 gives a $(1 + \epsilon)$ approximate MEB. Since we need the MEB algorithm for kernel spaces we extended the algorithm to take into account the new mapping function ϕ. In algorithm 2, the function $\texttt{FarthestPoint}(\boldsymbol{\lambda}, P)$ calculates the farthest point from the current center $c = \sum_{j=1}^{n} \lambda_j \phi(p_j)$ to the mapped points of P, without explicitly using $\phi()$.

If the dimension of the mapped space is finite, one can prove that algorithm 1 with $iter = \frac{1}{\epsilon^2}$ gives a $1 + \epsilon$ approximate minimum enclosing ball [1]. For the purposes of classification, we found that setting $iter = 100$ in practice works quite well for Gaussian kernels.

Once the centers of the mapped points are computed, we use a Voronoi diagram in the mapped space to do the classification. A point is classified according to $\min_{i=1..n} || c_i - p ||$ which can be again evaluated using kernel functions without mapping the points explicitly.

For verification purposes in our application we also needed to answer the question if a person is in the set of people that the database contains. The classifier classifies any person to a class, irrespective of whether the person is in the group of people who are in the database or not. For verification, we used a verifier that answers the question if a feature vector v belongs to one of the classes D_i where i is output of the classifier when v is fed into it. This can be turned into an outliers detection question. Is v an outlier for D_i? For this we tried to use a standard outlier detection method, the *Mahalanobis distance*. A point is a γ^2 outlier if its Mahalanobis distance is greater than γ. If μ_{D_i} and Σ_{D_i} are the mean vector and covariance matrix for the class D_i, then v is considered an outlier if $|| v - \mu_{D_i} ||_M^2 = (v - \mu_{D_i})' \Sigma_i^{-1}(v - \mu_{D_i}) \geq \gamma^2$ where $\Sigma_i = \frac{1}{2}(\Sigma_{D_i} + \Sigma^{pooled})$ and $\Sigma^{pooled} = \frac{1}{N-c} \sum_{i=1}^{c}(|D_i| - 1)\Sigma_{D_i}$. We found out that perturbing Σ^{pooled} matrix with Σ_{D_i} gave us better results than just using Σ^{pooled} for our verifier. The reason we did this was because the covariance matrix does not seem to be the same for all classes in our data.

6   Experimental Results

We tested the nearest box and the MEB classifier for training set sizes of 3, 4, and 5. Experiments were repeated for randomly chosen training sets and results averaged. For verification each set was queried by all the images. Figure **??** (a) shows the average plot of false acceptance rate and false rejection rate for different values of threshold. Figures **??** (b) shows the corresponding plots for

identification. False acceptance rate (FAR) is defined as the percentage of images that were accepted incorrectly. In case of identification, acceptance with wrong identification is considered false acceptance. False rejection rate (FRR) is defined as the percentage of images that were rejected incorrectly.

Using training set size of 5, we achieve the following results keeping FAR below 1%. For verification we achieve an FRR below 3% using the nearest box classifier and below 2% using the MEB classifier. For identification we achieve an FRR below 6% using the nearest box classifier and below 5.5% using the MEB classifier.

Similarly we tested classification using random training sets. We also compared our classifiers to SVM using the same kernel function and found out that we were competitive to standard SVM at least for this application. Table 1 reports the average misclassification rates.

The average timings for both our MEB and Nearest Box classifiers were negligible (Less than 1 millisecond for verification/classification/identification on our test PC with 1.3 Ghz Pentium IV and 512MB RAM).

7   Summary and Conclusions

Our experiments clearly show that hand based recognition systems can be used for medium security applications. Stronger claims about such a system can only be made after conducting experiments at a larger scale. Our novel minimum enclosing ball classifier works well for hand recognition. It may also be useful for other applications.

References

[1] Mihai Badoiu and Kenneth L. Clarkson. Smaller core-sets for balls. In *Proceedings of 14th ACM-SIAM Symposium on Discrete Algorithms*, pages 801–802, July 2003. 757

[2] Mihai Badoiu, Sariel Har-Peled, and Piotr Indyk. Approximate clustering via core-sets. In *Proceedings of 34th Annual ACM Symposium on Theory of Computing.*, pages 250–257, 2002. 757

[3] Asa Ben-Hur, David Horn, Hava T. Siegelmann, and Vladimir Vapnik. Support vector clustering. *Journal of Machine Learning*, revised version Jan 2002. 756

[4] O. Cuisenaire and B. Macq. Fast and exact signed euclidean distance transformation with linear complexity. In *Proc. of the IEEE Intl. Conf. on Acous., Speech and Sig. Processing*, volume 6, pages 3293–3296, March 1999. 755

[5] John Dunagan and Santosh Vempala. Optimal outlier removal in high-dimensional spaces. In *Proceedings of 33rd Annual ACM Symposium on Theory of Computing.*, pages 627 – 636, 2001. 756

[6] Nicolae Duta, A. K. Jain, and Kanti V. Mardia. Matching of palmprint. *Pattern Recognition Letters*, 23(4):477–485, 2002. 754

[7] K. W. Faulkner. Apparatus and method for biometric identification, August 1994. US Patent No. 5,335,288. 754

[8] A. K. Jain and Nicolae Duta. Deformable matching of hand shapes for verification. In *Proceedings of International Conference on Image Processing*, October 1999. 754

[9] A. K. Jain, S. Prabhakar, and Arun Ross. Biometrics-based web access. Technical Report MSU-CPS-98-33, Department of Computer Science, Michigan State University, East Lansing, Michigan, November 1998. 754

[10] A. K. Jain, A. Ross, and S. Pankanti. A prototype hand geometry-based verification system. In *Proc. of 2nd Int'l Conf. on Audio- and Video-based Biometric Person Authentication*, pages 166–171, March 1999. 754

[11] C. R. Maurer Jr., V. Raghavan, and R. Qi. A linear time algorithm for computing the euclidean distance transform in arbitrary dimensions. In *Proc. of the 17th International Conference on Information Processing in Medical Imaging*, volume 2082 of *LNCS*, pages 358–364, June 2001. 755

[12] Wenxin Li, David Zhang, and Zhuoqun Xu. Palmprint identification by fourier transform. *International Journal of Pattern Recognition and Artificial Intelligence*, 16(4):417–432, 2002. 754

[13] Cenker Öden, Aytül Erçil, Vedat Taylan Yildiz, Hikmet Kirmizitas, and Burak Büke. Hand recognition using implicit polynomials and geometric features. In *Proc. of the 3rd Intl. Conf. on Audio- and Video-Based Biometric Person Authentication*, volume 2091 of *LNCS*, pages 336–341. Springer, June 2001. 754

[14] Recognition Systems' HandKey II product. `http://www.recogsys.com/products/hk/ac_handkey_2.htm`. 754

[15] Arun Ross. A prototype hand geometry-based verification system. M.S. project report, Computer Science & Engineering, Michigan State University, East Lansing, MI 48824, USA, 1999. 753, 754

[16] Raul Sanchez-Reillo, Carmen Sanchez-Avila, and Ana Gonzalez-Marcos. Biometric identification through hand geometry measurements. *IEEE Tran. on Pattern Analysis and Mach. Intl.*, 22(10):1168–1171, October 2000. 754

[17] D.P. Sidlauskas. 3D hand profile identification apparatus, April 1988. US Patent No. 4,736,203. 754

[18] D. M. J. Tax. *One-class classification ; Concept-learning in the absence of counter-examples.* Ph.D. thesis, Delft University of Technology, 65, Delft, June 2001. 756

Feature-Level Fusion
for Effective Palmprint Authentication

Adams Wai-Kin Kong[1, 2] and David Zhang[1]

[1] Biometric Research Center, Department of Computing
The Hong Kong Polytechnic University, Kowloon, Hong Kong
[2] Department of Systems Design Engineering
University of Waterloo, Ontario, Canada N2L 3G1
adamskong@ieee.org
csdzhang@comp.polyu.edu.hk

Abstract. A feature-level fusion approach is proposed for improving the efficiency of palmprint identification. Multiple Gabor filters are employed to extract the phase information on a palmprint image, which is then merged according to a fusion rule to produce a single feature called the Fusion Code. The similarity of two Fusion Codes is measured by their normalized hamming distance. A database containing 7,752 palmprint images from 386 different palms is used to validate the performance of the proposed method. Empirically comparing our previous non-fusion approach and the proposed method, improvement in verification is ensured

1   Introduction

Biometric personal identification/verification has long been a widely studied topic. Various technologies, including iris, fingerprint, hand geometry, voice, face, signature and retina identification/verification [1-2], have been proposed and developed. Each of the these technologies has its own strengths and weaknesses. Currently, hand-based biometric technologies such as fingerprint verification and hand geometry verification are the most appealing approaches for the biometric identification market. They constitute a total of 60% of total market share as of 2001 [3].

   Automatic fingerprint verification is the most mature biometric technology which has been investigated and studied for more than 25 years. Although various scanning technologies, preprocessing, feature extraction and matching algorithms have been proposed for fingerprint verification, there are yet some problems waiting to be resolved. For example, based on the current fingerprint scanning technologies, approximately 1% of people have fingerprints that are almost impossible to be obtained, and 9% of the fingerprints are difficult to capture [4].

   Another hand-based biometric technology is hand geometry [3]. It uses geometric information on our hands for personal verification. Based on the simple features of our hands, hand geometry only provides limited accuracy and its ability to distinguish individuality is still an open question [1, 5]. To overcome the problems of the current

hand-based biometric technologies, we proposed to use another hand-based biometric, palmprint for personal identification/verification several years ago.
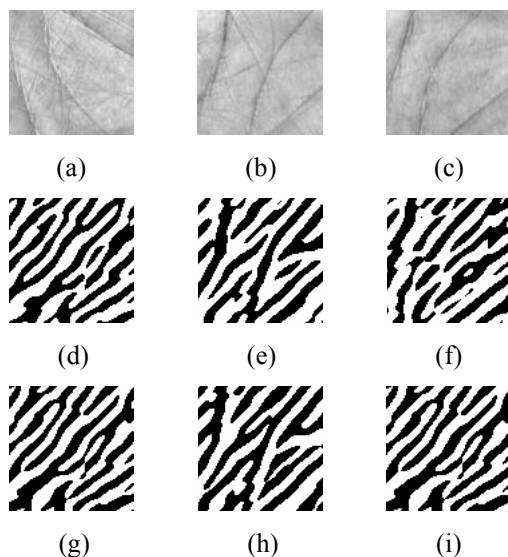
Fig. 1. Three typical samples of PalmCodes: (a)-(c) original images, (d)-(f) real parts of PalmCodes, (g)-(i) imaginary parts of PalmCode

Palmprint, a large inner surface on our hand, contains many line features, for example, principal lines, wrinkles, and ridges. Because of the large surface and the rich line features, we expect palmprints to be robust to noise and to have high individuality. The most promising results developed by us are obtained from a texture-based approach published in [11], which applies Daugman's iris coding scheme [13] to palmprint images. The extracted feature is called PalmCode. In this paper, we propose to use a fusion technique to further improve the coding scheme for palmprint identification.

1.1 Motivation

Since the proposed method is developed with reference to PalmCode, we begin our work by a short review about the concept:

1. An adjusted 2-D Gabor filter is applied to the preprocessed palmprint images [11].
2. The signs of the filtered images are coded as a feature vector.
3. Two PalmCodes are measured by the normalized hamming distance.

The detailed implementation of PalmCode and preprocessed palmprint image is mentioned in [11]. Figs. 1(d)-(i) are three PalmCodes derived from the three different palms in Figs. 1(a)-(c). We can observe that the PalmCodes from the different palmprints are similar, which are constituted by many 45° streaks. Intuitively, these

structural similarities among PalmCodes from different palms reduce the individuality of PalmCode and the performance of the palmprint identification system.

(a)

(b)

(c)

(d)

(e)

(f)
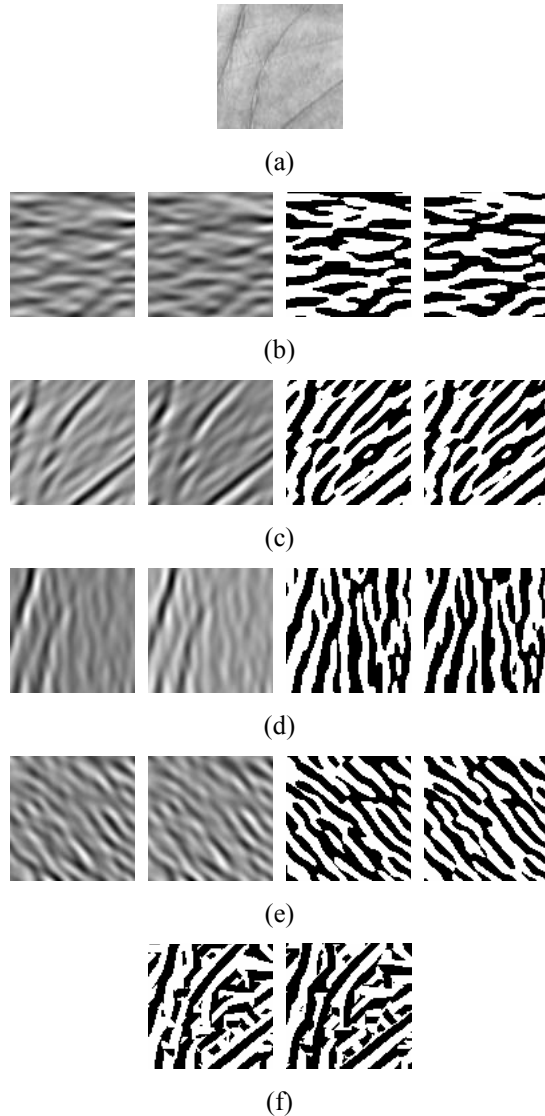
Fig. 2. Procedure of how the Fusion Code is generated: (a) original palmprint image, (b)-(e) real parts (Column 1) and imaginary parts (Column 2) of the filtered images, and real parts (Column 3) and imaginary parts (Column 4) of PalmCodes and (f) Fusion Code

In this paper, multiple Gabor filters are applied to palmprints and a feature-level fusion technique is introduced to merge the filtered images so as to:

1. destruct the structural similarity among different palmprint features from different palms;
2. increase the individuality among palmprint features from different palms; and
3. increase the performance of our palmprint identification system.

This paper is organized in the following way. Section 2 and Section 3 present the step by step implementation of Fusion Codes and illustrate the comparison of the two Fusion Codes, respectively. Experimental results of the proposed method are given in Section 4. Finally, Section 5 summarizes the main results of this paper and offers concluding remarks.

2   Implementation of Fusion Code

First, the preprocessed palmprint image is passed to a circular Gabor filter bank. The filter bank contains four circular Gabor filters, which have the following general formula:

$$G(x, y, \theta, u, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left\{-\frac{x^2 + y^2}{2\sigma^2}\right\} \exp\{2\pi i(ux \cos\theta + uy \sin\theta)\} \tag{1}$$

where, u is the frequency of the sinusoidal wave, θ controls the orientation of the function, and σ is the standard deviation of the Gaussian envelope. Same as the implementation of PalmCode, the Gabor filters are adjusted to zero DC (direct current). The parameter θ for the four Gabor filters are 0, $\pi/4$, $\pi/2$ and $3\pi/4$. The parameters u and σ for the four Gabor filters are 0.0916 and 5.6179, respectively. In fact, the PalmCode reported in [11] only uses the Gabor filter with the parameters $\theta = \pi/4$, u=0.0916 and σ =5.6179. Figs. 2(b)-(e) show the filtered palmprint images and the corresponding PalmCodes. For convenience sake, we use G_j, where j=1,2,3,4 to represent the four Gabor filters.

2.1 Fusion Rule Design and Feature Coding

The filtered images contain two kinds of information: magnitude M_j and phase P_j, which are defined as

$$M_j(x, y) = \sqrt{G_j * I(x, y) \times \overline{G_j * I(x, y)}} \tag{2}$$

and

$$P_j(x, y) = \tan^{-1}\left(\frac{i(\overline{G_j * I(x, y)} - G_j * I(x, y))}{G_j * I(x, y) + \overline{G_j * I(x, y)}}\right), 3 \tag{3}$$

where "——" represents complex conjugate, "*" is an operator of convolution and I is a preprocessed palmprint image. Because of the zero DC Gabor filters, both of them are independent of the DC of the image. DC replies on the brightness of the capturing environment. In addition to the DC, phase is also independent of the contrast of the image but it is not true for the magnitude. As a result, since the PalmCode only uses the phase information, it is stable for two properties: variations of the contrast, and *DC* of palmprint images. To design a fusion coding scheme inheriting these two

properties, we employ the magnitude for fusion and the phase for the final feature. Thus, we propose a fusion rule:

$$k=arg\ max_j\ (M_j(x,y))\quad(4)$$

and coding equations:

$$
\begin{array}{llll}
(h_r, h_i)=(1, 1) & if & 0 \leq P_k(x, y) < \pi/2, & (5)\\
(h_r, h_i)=(0, 1) & if & \pi/2 \leq P_k(x, y) < \pi, & (6)\\
(h_r, h_i)=(0, 0) & if & \pi \leq P_k(x, y) < 3\pi/2, & (7)\\
(h_r, h_i)=(1, 0) & if & 3\pi/2 \leq P_k(x, y) < 2\pi, & (8)
\end{array}
$$

where h_r and h_i are bits in the real and the imaginary parts of the Fusion Code. A Fusion Code is illustrated in Fig. 2.(f).

3   Similarity Measurement of Fusion Codes

In terms of the feature format, the proposed Fusion Code is exactly the same as that of the PalmCode. Thus, the normalized hamming distance for the PalmCode is still useful for the Fusion Code. If we are given two data sets, a matching algorithm would determine the degree of similarity between them. To describe the matching process clearly, we use a feature vector to represent image data that consists of two feature matrices, the real one and the imaginary one. A normalized hamming distance is adopted to determine the similarity measurement for palmprint matching. Let P and Q be two palmprint feature vectors. The normalized hamming distance can be described as:

$$
D_o = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} P_M(i,j) \cap Q_M(i,j) \cap ((P_R(i,j) \otimes Q_R(i,j) + P_I(i,j) \otimes Q_I(i,j)))}{2\sum_{i=1}^{N}\sum_{j=1}^{N} P_M(i,j) \cap Q_M(i,j)}\quad(9)
$$

where $P_R(Q_R)$, $P_I(Q_I)$ and $P_M(Q_M)$ are the real part, the imaginary part and the mask of $P(Q)$, respectively. The mask is used for denoting the non-palmprint pixels as described in [11]. The result of the Boolean operator ($\otimes$) is equal to zero, if and only if the two bits, $P_{R(I)}(i,j)$, are equal to $Q_{R(I)}(i,j)$. The symbol $\cap$ represents the AND operator, and the size of the feature matrices is $N\times N$. It is noted that D_o is between 1 and 0. For the best matching, the normalized hamming should be zero. Because of imperfect preprocessing, we need to translate vertically and horizontally one of the features and match again. The ranges of the vertical and the horizontal translations are defined from –2 to 2. The minimum D_0 value obtained from the translated matching is considered to be the final matching score.

4   Experimental Results

We collected palmprint images from 193 individuals using our palmprint capture device described in [11]. The subjects are mainly students and staff volunteers from the Hong Kong Polytechnic University. In this dataset, 131 people are male, and the age distribution of the subjects is: about 86% are younger than 30, about 3% are older than 50, and about 11% are aged between 30 and 50. In addition, we collected the palmprint images on two separate occasions, at an interval of around two months. On

each occasion, the subject was asked to provide about 10 images each of the left palm and the right palm. Therefore, each person provided around 40 images, resulting in a total number of 7,752 images from 386 different palms in our database. In addition, we changed the light source and adjusted the focus of the CCD camera so that the images collected on the first and second occasions could be regarded as being captured by two different palmprint devices. The average time interval between the first and second occasions was 69 days. The maximum and the minimum time intervals were 162 days and 4 days, respectively.

4.1 Verification

To obtain the verification accuracy of the proposed method, each palmprint image was matched with all of the palmprint images in the database. A matching is counted as a correct matching if two palmprint images are from the same palm. The total number of matching is 30,042,876. None of the hamming distances is zero. The number of comparisons that have correct matching is 74,086 and the rest are incorrect matching. Fig. 3 depicts the corresponding Receiver Operating Characteristic (ROC) curve, which is a plot of the genuine acceptance rate against the false acceptance rate for all possible operating points. In Fig. 3, we can see that our method can operate at a genuine acceptance rate of 98% and a false acceptance rate of 0.01%, with a corresponding threshold 0.36. We also plot the ROC curve of PalmCode for comparison. According to the ROC curves, the performance of Fusion Code is better than that of PalmCode. The verification accuracy of Fusion Code is comparable with previous palmprint approaches [6-12]

5   Conclusion

We have presented a feature-level coding scheme for improving the performance of PalmCode [11], which was the best palmprint identification approach developed by our group. The proposed Fusion Code applies four Gabor filters to the preprocessed palmprint images to compute four PalmCodes. According to the fusion rule, the four PalmCodes are merged to construct Fusion Code. We have proved that Fusion Code is independent of the contrast and the brightness of the palmprint images. The total size of Fusion Code and its mask is 384 bytes, same as that of PalmCode.

In our testing database containing 7,752 palmprint images from 386 different palms, Fusion Code can achieve high genuine (98%) and low false acceptance (0.01%) verification rates, which is comparable with all other palmprint recognition approaches [6-12]. The execution time for the whole process, including preprocessing, feature extraction and final matching, is between 1 and 1.2 seconds on a PC embedded Intel Pentium III processor (500MHz).
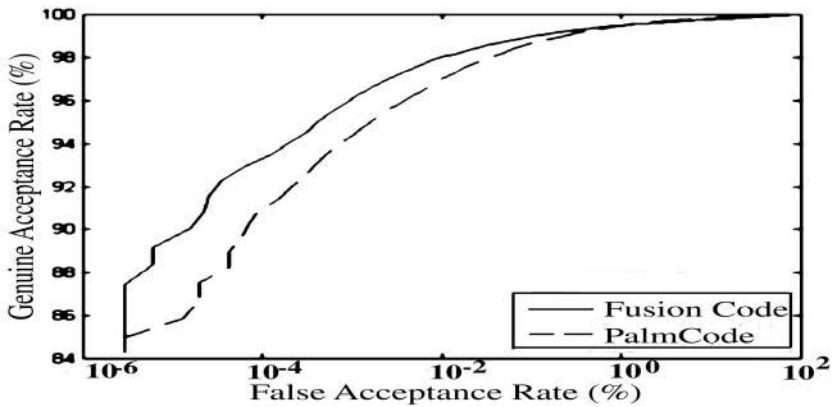
Fig. 3. Verification test results. (a) Genuine and imposter distributions and (b) the receiver operator characteristic curves of Fusion Code and PalmCode

References

[1] Jain, R. Bolle and S. Pankanti (eds.), *Biometrics: Personal Identification in Networked Society*, Boston, Mass: Kluwer Academic Publishers, 1999.

[2] 2. D. Zhang, *Automated Biometrics – Technologies and Systems*, Boston: Kluwer Academic Publishers, 2000.

[3] International Biometric Group's Biometric Market Report 2000-2005: http://www.biometricgroup.com/e/biometric_market_report.htm

[4] G. Lawton, "Biometrics: a new era in security", *Computer*, vol. 31, no. 8, pp. 16-18, 1998.

[5] S. Pankanti, R.M. Bolle and A. Jain, "Biometrics: the future of identification", *Computer*, vol. 33, no. 2, pp. 46-49, 2000.

[6] D. Zhang and W. Shu, "Two novel characteristics in palmprint verification: datum point invariance and line feature matching," *Pattern Recognition*, vol. 32, no. 4, pp. 691-702, 1999.

[7] N. Duta, A.K. Jain, and K.V. Mardia, "Matching of Palmprint", *Pattern Recognition Letters*, vol. 23, no. 4, pp. 477-485, 2001.

[8] J. You, W. Li and D. Zhang, "Hierarchical palmprint identification via multiple feature extraction", *Pattern Recognition*, vol. 35, no. 4, pp. 847-859, 2002.

[9] W. Li, D. Zhang and Z. Xu, "Palmprint identification by Fourier Transform", *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 16, no. 4, pp. 417-432, 2002.

[10] C.C. Han, H.L. Cheng, K.C. Fan and C.L. Lin, "Personal authentication using palmprint features", *Pattern Recognition*, Special Issue: Biometrics, vol. 36, no 2, pp. 371-381, 2003.

[11] D. Zhang, W.K. Kong, J. You and M. Wong, "On-line palmprint identification," *IEEE Trans. PAMI*, vol. 25, no. 9, pp. 1041-1050, 2003.

[12] G.M. Lu, D. Zhang and K.Q. Wang, "Palmprint recognition using eignpalms features", *Pattern Recognition Letter*s, vol. 24, pp. 1463-1467, 2003.

[13] J.G. Daugman, "High confidence visual recognition of persons by a test of statistical independence", *IEEE Trans. PAMI*, vol. 15, no. 11, pp. 1148-1161, 1993.

A Palmprint Acquisition Device with Time-Sharing Light Source Used in Personal Verification

Weinan Zhao[1], Wenxin Li [1,2], Tao Wang[1], and Zhuoqun Xu[1]

[1]Department of Computer Science and Technology, Peking University
[2] Department of Computing, The HongKong Polytechnic University
wnzhao@water.pku.edu.cn
{lwx,zqxu}@pku.edu.cn
won_tar@ailab.pku.edu.cn

Abstract. This paper proposes a hardware-based line feature enhancement approach that uses a palmprint acquisition device fitted with a time-sharing light source. When capturing images, two lamps of the light source can be lit in turn, producing two, differently lit images. As the two lamps light the inner surface of the palm from different directions, they capture images with different lines enhanced. By fusing the two images with a nonlinear method, we can produce a palmprint with much clearer line features. Our research will focus on improving this device and integrating it into a functional palmprint verification system.

1   Introduction

Palmprint image acquisition is the first issue in a palmprint verification system. Previous acquisition methods are based on the requirement of "to make the captured image as clear as possible"[1]. This requirement calls for a uniform light source, since it doesn't emphasize specific palmprint features.

However, Palmprint verification systems commonly make use of the principal lines of the palm. In 1999, Wei Shu and David Zhang introduced a recognition algorithm based on palmprint line extraction and matching [2]. In 2002, Nicolae Duta and Anil K. Jain proposed an algorithm based on extraction and matching of the point on the major lines on the palm surface [3]; in 1998, J. You and W. Li presented a Hierarchical method based on feature points in principal lines and line patterns [4]; in 1998, Jun-ichi Funada proposed an algorithm to extract ridge structure features by removing the principal lines [5].

What all these methods have in common is a software approach to line feature extraction. In this paper, our proposed approach in contrast seeks to enhance line features with a new palmprint acquisition device. The rest of this paper is organized as follows. Section 2 explores the principal line imaging mechanism and describes the principal line imaging model. Section 3 presents the design of the palmprint acquisition device. Section 4 introduces our image fusion algorithm. Section 5

provides the experimental results. Section 6 concludes the proposed method and gives the future research directions.

2 The Principal Line Imaging Model

Principal lines on the palm surface are shown in Fig. 1. We name 1, 2 and 3 as Line 1, Line 2 and Line 3 respectively. To model them in an optical imaging system, we need the knowledge of their exact geometry properties. The anatomical structure of a principal line is shown in Fig. 2; besides, according to our observations, the most depressed part of the palm surface is generally located between Line 2 and Line 3 (area C in Fig. 1). That means for Line 3, the side near area A is higher than the other; while for Line 1 and Line 2, the side near area B is higher. These properties can be integrated into the principal line imaging models shown in Fig. 3 and Fig. 4.

To increase the contrast between the groove and its two sides, we need to illuminate the latter while at the same time reduce the amount of light that enters the former as much as possible. In the models shown in Fig. 3. and Fig. 4, this means the incoming ray should go along the direction from the higher side to the other side (direction 2 in Fig. 3 and Fig. 4), and the angle between the incoming ray and the palm surface (θ in Fig. 3 and Fig. 4) should as small as possible as long as it is bigger than some threshold that ensure the two sides sufficient illumination. Obviously, the three lines cannot be enhanced simultaneously using a single lamp. We use two lamps in the device. One lamp L_1 is used to enhance Line 3 and another lamp L_2 suffices to enhance Lines 1 and Line 2, since their directions are similar.

The effect of the enhancement relies on the relative location between the lamp and the principal lines. Thus two factors are important: palm positioning and individuality in the geometry of the palm surface. Our positioning method is shown in Fig. 5. a support saddle is installed to position the wrist and a support pole to position the fingers. When capturing, the subject is required to put his wrist on the support saddle, fingers on the support pole with the positioning wheel between middle finger and ring finger. In order to dealing with the individuality in the geometry of the palm surface, we experiment on a sample set of 30 samples (each sample includes two images) from 10 persons. Finally the location of the light source is fixed, as can be seen in Fig. 5.

Fig. 1. Principal lines on the palm surface **Fig. 2.** Anatomical structure of a principal line

Fig. 3. Imaging model of Line 3   **Fig. 4.** Imaging model of Line 1 and Line 2
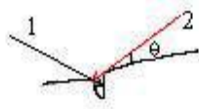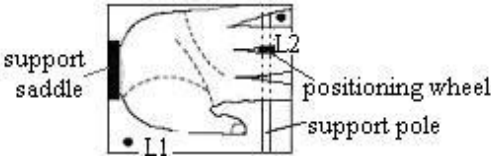
Fig. 5. Palm positioning and the location of the time-sharing light source

3   The Palmprint Acquisition Device with Time-Sharing Light Source (TSLS)

In this section we will present the design of our device. Its appearance is shown in Fig. 6.

When L_1 and L_2 are lit at the same time, they will interfere with each other. As a result, the enhancement effect will be weakened: for Line 3, the incoming rays from L_2 will increase the amount of the rays entering the groove so that the enhancement effect achieved by L_1 will be weakened. Similarly, the rays from L_1 will also weaken the enhancement effect achieved by L_2. To address this problem, we introduce a time-sharing light source that employs a time-sharing light source controller (TSLSC) to control the two lamps: light up L_1 when capturing image with Line 3 enhanced, while light up L_2 when capturing image with Line 1 and Line 2 enhanced. To avoid hand moving in the interval between two capture, we desire the interval to be as small as possible. Our palmprint acquisition system is shown in Fig. 7. In this system, TSLSC, controlling the quartz lamps, is connected to PC through the parallel port, and a camera is connected to PC through the usb port. We use software approach to synchronize lamp lighting with image capture. The minimum interval between two capture achieved by this system is 1s. It is possible for subjects to keep their hand still in such a small interval.

Fig. 6. Outlook of the palmprint acquisition device
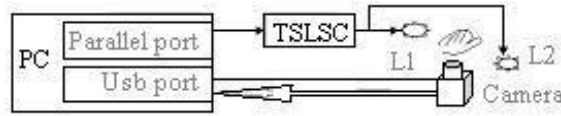
Fig. 7. Palmprint acquisition system with time-sharing light source

a. Image under uniform   b. Image under L1   c. Image under L2
light source

Fig. 8. Image under uniform light source and under the time-sharing light source

4   Image Fusion Algorithm

4.1 Objective

An example of captured images using our approach is shown in Fig. 8. (For contrast, we also give the image captured under a uniform light source). Examine the three images, we can find that Line 2 is blurry under the uniform light source but is successfully enhanced under L2. At the same time, we can find obvious enhancement to Line 1 (when contrast a and c) and Line 3 (when contrast a and b). For convenience of discussion, we refer to the image captured under uniform light source as I0, the image captured under L1 as I1, and the image captured under L2 as I2. Our objective is to fuse I1 and I2 into one image named I, in which all enhanced lines will be reserved as possible with least other information lost. Since the subjects are required to keep their hands still during capturing process, image registration is unnecessary. Then we can focus on the fusion method. Image fusion methods have been discussed extensively in the literature, but few of them apply to this special case. An example of these methods [6] is shown in Fig. 10 (a).

4.2 Nonlinear Fusion

Our nonlinear fusion method is based on the following observations (we refer to the pixels in the image that correspond to the vertical projection point of L_1 and L_2 on the image plane as LP_1 and LP_2 respectively): in I_1, the information closer to LP_1 than to

LP_2 is more desired to be reserved in I; in I_2, the information closer to LP_2 than to LP_1 is more desired to be reserved in I. It can be formulated as follows:

For each pixel P in I, its value,

$$p=p_1 \times w(d_1,d_2)+p_2 \times(1-w(d_1,d_2)) \tag{1}$$

where, p_1 and p_2 are pixel values in I_1 and I_2 respectively; d_1 and d_2 are the distances from P to LP_1 and to LP_2; $w(d_1, d_2)$ is the weight function with the form:

$$w(d_1,d_2)=u(N(d_1,d_2)) \tag{2}$$

$$N(d_1,d_2)=\frac{wod \times d_1}{wod \times d_1+(2-wod) \times d_2} , \ 0<wod<2 \tag{3}$$

$$u(x)=\begin{cases} -\left(\frac{1}{2}\right)^{1-pow}\left(x-\frac{1}{2}\right)^{pow}+\frac{1}{2} , & x>\frac{1}{2} \\ \frac{1}{2} , & x=\frac{1}{2} \\ \left(\frac{1}{2}\right)^{1-pow}\left(\frac{1}{2}-x\right)^{pow}+\frac{1}{2} , & x<\frac{1}{2} \end{cases} \tag{4}$$

where, $N(d_1, d_2)$ together with $u(x)$ decide weights for p_1 and p_2 that measure their importance in I according to their location. $N(d_1, d_2)$ calculates the initial weight according to d_1 and d_2. It indicates the relative location of P between LP_1 and LP_2 in sense of weighted distance: the more is the weighted distance of d_1 greater than that of d_2, the greater $N(d_1, d_2)$ will be. When $N(d_1, d_2)=1/2$, the corresponding locations form a line (named balance line) on which p_1 and p_2 are of equal importance. wod controls the location of the balance line. If $wod<1$, the balance line will bias to LP_2 and I_1 will hold more portion in I. And vice versa.

$u(x)$ enlarges the difference between the initial weights for p_1 and p_2 using a nonlinear transform. It is a descending function since we consider p_1 more important when its initial weight $N(d_1, d_2)$ is smaller. Fig. 9 shows the curve of $u(x)$. It has two desired properties: 1) the point $x=0.5$ corresponds to the balance line. When $x=0.5$, $u(x)>0.5$, that means in the area between LP_1 and the balance line, I_1 will be treated more important than I_2, and vice versa; 2) in the area around the balance line, it provides a smooth transition. pow value controls the convex degree of the $u(x)$ curve (see Fig. 9) and thus affects the two properties: as pow value decreases, the difference between I_1 and I_2 in their importance will be more enlarged; however, the transition will be less smooth.

Fig. 9. Curve of $u(x)$

a. Li et al's method

b. Nonlinear method
(pow=1, wod=1)

c. Nonlinear method
(pow=0.2, wod=1)

d. Nonlinear method
(pow=0.4, wod=1)

e. Nonlinear method
(pow=0.4, wod=0.8)

f. Nonlinear method
(pow=0.4, wod=1.2)
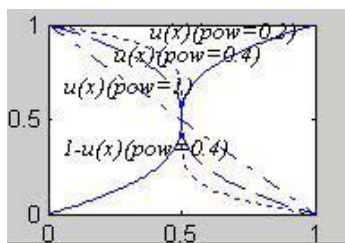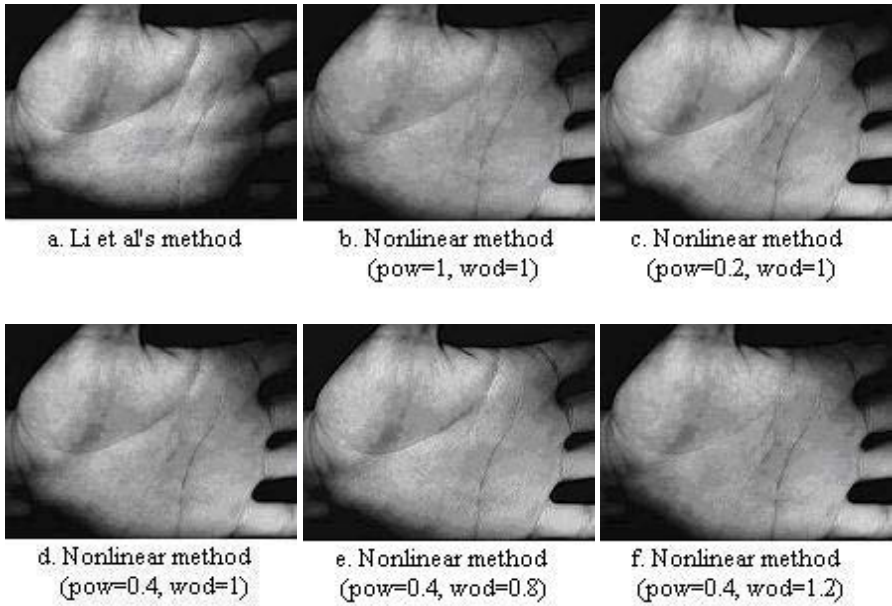
Fig. 10. Image fusion result

5   Experimental Results and Discussion

5.1 Results of Different *pow* Value

As *pow* value decrease, the fused image becomes sharper, but the balance line becomes obvious. Since the former is desired for line feature extraction and the latter is not, we have to made tradeoff between them. Experiments show the optimal value is 0.4. Typical results are shown in Fig. 1.0 (b, c, d).

5.2 Results of Different *wod* Value

wod value controls the location of the balance line. Since in the area around the balance line, the image will become less sharp, we need to find the optimal *wod* value. Through experiments we found the optimal value to get quite sharpness in most cases is 1. Typical results are shown in Fig. 1.0 (d, e, f).

5.3 Discussion

Using the nonlinear fusion method, we experiment on a sample set of 80 samples (each sample includes two images those are captured under L_1 and L_2 respectively) from 10 persons. Principal lines in 77 groups are enhanced in different degrees. This

result is encouraging. However, the 3 exceptions show we need more precise positioning method; besides, in our fusion algorithm, due to the variation in the relative locations between the principal lines and LP_1 and LP_2, the sharpness of the principal lines near the balance line will vary a little. To improve that requires an adaptive mechanism.

6   Conclusion and Future Work

The palmprint acquisition device with time-sharing light source demonstrates a good ability to enhance the principal lines in palmprint in contrast with software enhancement method. When adopting more precise positioning method, we can use this device in large-scale sampling. Our future work will focus on two issues: 1) to further reduce the interval between two capture in order to provide more user-friendly interface; 2) to make the nonlinear image fusion method more adaptive.

References

[1]   Michael Wong and David Zhang, "Design and Implementation of a High Quality Palmprint Acquisition Device", The 4th ACM Hong Kong Postgraduate Research Conference, 2003

[2]   D. Zhang and W. Shu, "Two novel characteristics in palmprint verification: datum point invariance and line feature matching", Pattern Recognition, 1999, 32(4): 691-702

[3]   Nicolae Duta and Anil K. Jain and Kanti V. Mardia, "Matching of palmprints" Pattern Recognition Letters, 2002,23(4): 477-485

[4]   J. You, W. Li, and D. Zhang, "Hierarchical Palmprint Identification via Multiple Feature Extraction", Pattern Recognition, vol. 35, no. 4, pp. 847-859, 2002.

[5]   Jun-ichi Funada, "Feature Extraction Method for Palmprint Considering Elimination of Creases", 14th International Conference on Pattern Recognition, 1998, Vol SA45(2): 1849-1853.

[6]   H. Hui Li, B. S. Manjunath, and S. K. Mitra, "Multi-Sensor Image Fusion Using the Wave let Transform," Proc. ICIP'94, I: 51-55, IEEE Computer Society Press, Los Alamitos, CA, 1994

HMMs Based Palmprint Identification

Xiangqian Wu[1], Kuanquan Wang[1], and David Zhang[2]

[1] School of Computer Science and Technology
Harbin Institute of Technology (HIT), Harbin 150001, China
{xqwu,wangkq}@hit.edu.cn
http://biometrics.hit.edu.cn
[2] Biometric Research Centre, Department of Computing
Hong Kong Polytechnic University, Kowloon, Hong Kong
csdzhang@comp.polyu.edu.hk

Abstract. This paper presents a novel approach of palmprint identification with Hidden Markov Models (HMMs). Palmprint is first aligned and normalized by using the boundary of the fingers. Then the continuous HMMs are used to identify palmprints. The palmprint features are extracted by using Sobel operators and projecting technique. It shows that HMMs with six states and two Gaussian mixtures can obtain the highest identification rate, 97.80%, in one-to-320 matching test. Experimental results demonstrate the feasibility of HMMs on the palmprint identification task.

1 Introduction

As relatively new biometric features, palmprints have several advantages compared with other currently available features [1]: palmprints contain more information than fingerprints, so they are more distinctive; palmprint capture devices are much cheaper than iris devices; palmprints also contain additional distinctive features such as principal lines and wrinkles, which can be extracted from low-resolution images; a highly accurate biometrics system can be built by combining all features of palms, such as palm geometry, ridge and valley features, and principal lines and wrinkles, etc. Many algorithms have been developed for palmprint recognition in the last several years [2, 3, 4, 5]. All of these palmprint recognition algorithms were focused on palmprint feature extraction. In this paper, we investigate palmprint identification from the point of view of classifiers. Hidden Markov models (HMMs) classifiers, as statistical models with a profound mathematical basis, have been very successfully used in speech recognition [6] and optical Chinese character recognition [7]. HMMs have also been effectively employed in some biometrics fields, such as speaker recognition [8], etc. To use HMM in pattern recognition, a high-correlated feature sequence should be extracted from the patterns. The gradient magnitudes and directions of the neighboring points on palm-lines are high-correlated. This fact motivates us to investigate the feasibility of applying HMMs to palmprint identification.
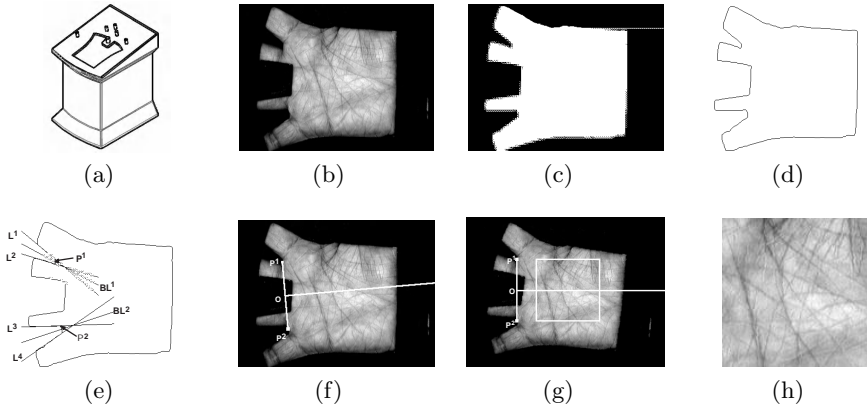
Fig. 1. The main steps of preprocessing. (a) Palmprint capture device, (b) original image, (c) binary image, (d) boundary tracking, (e) line fitting, bisectors and intersections, (f) the palmprint coordinate system, (g) central part extraction and (h) the preprocessed result

2   Palmprint Preprocessing

Palmprint images should be orientated and normalized before feature extraction and matching. In our CCD based palmprint capture device [3], there are some pegs between fingers to limit the palm's stretching, translation and rotation (Fig. 1(a)). These pegs separate the fingers, which enables us to use the points on the fingers' boundary to align and normalize palmprints. An original palmprint captured by the device is shown in Fig. 1(b). There are six main steps in the preprocessing:

1. Smooth the original image by a low-pass filter and use a threshold to convert it to a binary image (see Fig. 1(c));
2. Trace the palm's boundary (see Fig. 1(d));
3. Use straight lines (L^1, L^2, L^3 and L^4) to fit the segments of the boundary of the first finger, middle finger, third finger and little finger (see Fig. 1(e));
4. Computing the bisectors (BL^1 and BL^2) of the angles formed by L^1 and L^2, L^3 and L^4 and the intersection of BL^1 and the boundary, BL^2 and the boundary are P^1 and P^2, respectively (see Fig. 1(e)).
5. Line up point P^1 and P^2, and make a palmprint coordinate system in which y-axis is line P^1P^2 and the original point is the midpoint of line segment P^1P^2 (see Fig. 1(f)).
6. Crop a sub-image with fixed size from the center of the image (see Fig. 1(g) and Fig. 1(h)).

In this paper, all of the original palmprints are 384×284. The central part of the palmprint image, whose size is 128×128, is cropped to represent the whole palmprint. For feature extraction, the sub-images with size 128×128 are resized to 64×64 by using the bilinear interpolation.

3   Theory of Hidden Markov Models

A discrete HMM λ is defined as a five-tuple:

$$\lambda = (N, M, A, B, \pi) \tag{1}$$

where N and M are the number of states (denoted as $q_1, q_2, \ldots, q_N$) and the number of distinct observation symbols per state (denoted as $v_1, v_2, \ldots, v_M$), respectively; $A = \{a_{ij}\}_{1 \leq i,j \leq N}$ is the state transition probability distribution matrix in which $0 \leq a_{ij} \leq 1$ is the transition probability from state q_i to q_j state and, for each i, $\sum_j a_{ij} = 1$; $B = \{b_j(k)\}_{1 \leq j \leq N, 1 \leq k \leq M}$ is the observation symbol probability distribution matrix in which $0 \leq b_j(k) \leq 1$ is the probability of observation symbol v_k at state q_j and, for each j, $\sum_k b_j(k) = 1$; $\pi = \{\pi_i\}_{1 \leq i \leq N}$ is the initial state distribution vector in which $0 \leq \pi_i \leq 1$ is the initial probability of state q_i and $\sum_i \pi_i = 1$.

Discrete HMMs cannot deal with the case that the observations are continuous signals (or vectors). In this case, we replace $b_j(k)$ in the observation symbol probability distribution matrix B in Eq.(1) with the continuous model probability density function (pdf) $b_j(O)$, where O is the observed vector. In general, the pdf is represented by a finite mixture of the form:

$$b_j(O) = \sum_{m=1}^{M} c_{jm} G[O, \mu_{jm}, U_{jm}], \qquad 1 \leq j \leq N \tag{2}$$

where c_{jm} is the mixture coefficient for the mth mixture in state q_j and G is a Gaussian density with mean vector μ_{jm} and covariance matrix U_{jm} for the mth mixture component in state q_j. The mixture coefficients c_{jm} satisfy the stochastic constraint:

$$c_{jm} \geq 0, \qquad 1 \leq j \leq N, \ 1 \leq m \leq M \tag{3}$$

$$\sum_{m=1}^{M} c_{jm} = 1, \qquad 1 \leq j \leq N \tag{4}$$

There are three basic problems for HMMs that must be solved. Problem 1 is the evaluation problem: Given the observation sequence $O = O_1 O_2 \ldots O_T$ and a model $\lambda = (N, M, A, B, \pi)$, how to efficiently compute $P(O|\lambda)$, i.e. the probability of the observation sequence, given the model? Problem 2 is the decoding problem: Given the observation sequence $O = O_1 O_2 \ldots O_T$ and a model $\lambda = (N, M, A, B, \pi)$, how to get the optimal state sequence in some meaningful sense? Problem 3 is learning problem: how to adjust the model parameters $\lambda = (N, M, A, B, \pi)$ to maximize $P(O|\lambda)$? These problems have been resolved by the forward-backward procedure, Viterbi algorithm and Baum-Welch method (EM method), respectively [6].

1	2	1
0	0	0
-1	-2	-1

(a) S_{0^0}

0	1	2
-1	0	1
-2	-1	0

(b) S_{45^0}

1	0	-1
2	0	-2
1	0	-1

(c) S_{90^0}

2	1	0
1	0	-1
0	-1	-2

(d) S_{135^0}

Fig. 2.   Four directional Sobel operators

(a)   (b)   (c)   (d)   (e)
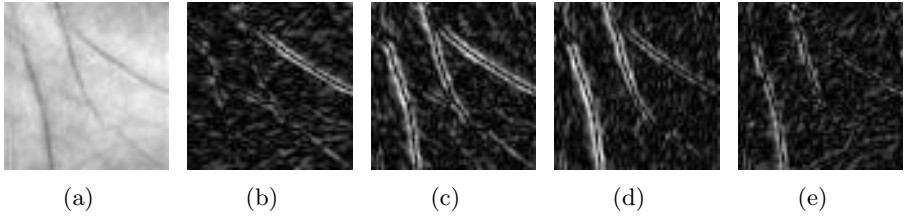
Fig. 3.   An example of the directional magnitude images: (a) is the original palmprint; (b), (c), (d) and (e) are the magnitude image obtained by opera-tor: $S_{0^0}, S_{45^0}, S_{90^0}$ and S_{135^0}

4   Feature Extraction

An essential issue of using a HMM classifier in pattern recognition is to extract the pattern features and rearrange them in a high-correlated se-quence. The main purpose of this paper is to investigate the feasibility of the HMM classifier on palmprint identification, therefore, we only use a very simple method for palmprint feature extraction. Four directional Sobel opera-tors $S_{0^0}, S_{45^0}, S_{90^0}$ and S_{135^0} are designed as Fig. 2. The palmprint I is convolved with these operators to obtain four directional magnitude images (See Fig. 3):

$$M_\alpha = |(I * S_\alpha)| \qquad \alpha = 0^0, 45^0, 90^0 \text{ and } 135^0 \tag{5}$$

where $*$ represents the convolve operation and $|\cdot|$ is the operation to get a new matrix whose components are the absolute value of the corresponding component in the original matrix.

If we project $M_\alpha(\alpha = 0^0, 45^0, 90^0$ and $135^0)$ along both x and y directions, we can obtain 64 histogram values along the axis x and another 64 histogram values along the axis y. These two histograms are concatenated to form 128 histogram values $O_\alpha(j), \ j = 1, 2, \ldots, 128, \alpha = 0^0, 45^0, 90^0$ and 135^0. $O_\alpha(j)$ represents the total line points' magnitude in α direction in the jth row $(j \leq 64)$ or in the $(j - 64)^{th}$ column $(j > 64)$. Now we can obtain a sequence containing 128
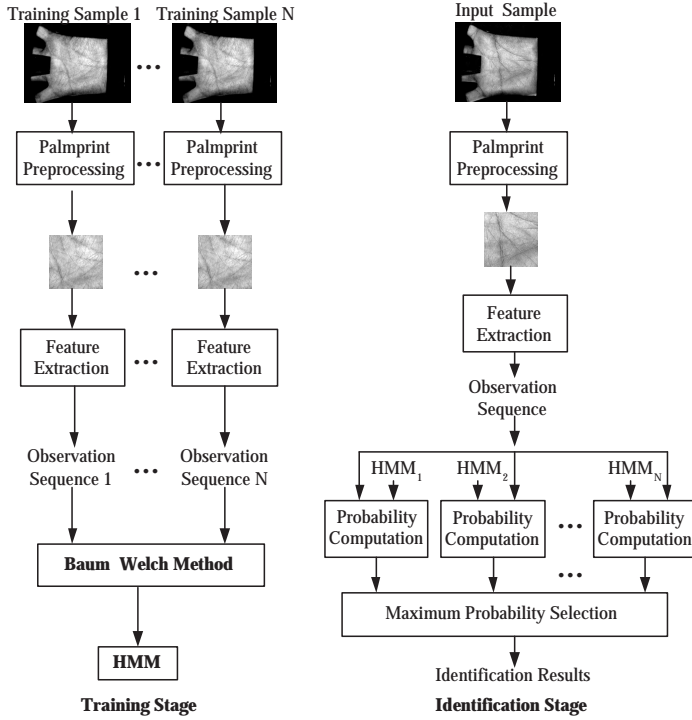
Fig. 4. The block diagrams of the training and identification stages

observations, in which each observation is a four-dimensional vector:

$$O(j) = (O_{0^o}(j), O_{45^o}(j), O_{90^o}(j), O_{135^o}(j))^T, \qquad j = 1, 2, \ldots, 128 \qquad (6)$$

The observation $O(j)$ reflects the lines' magnitudes in different directions in a row or a column of a palmprint. This sequence is used as the feature fed into HMM for training and identification.

Palmprint identification with HMMs involves a training stage and an identification stage. In the training stage, we train a HMM for each palmprint class. First we compute the observation sequences of the training palmprints of a class C_i . Then we feed these observation sequences into Baum-Welch method [6] to train a HMM λ_i for this class. Finally we store the trained HMM for palmprint identification. In identification stage, the observation sequence of the input image is first extracted. Then, given each stored HMM, the probability of this sequence is computed. Finally, the label of the HMM corresponding to the maximum probability is taken as the identification result. Fig. 4 shows block diagrams of the training and identification stages.

Fig. 5. Some typical samples used in the experiments
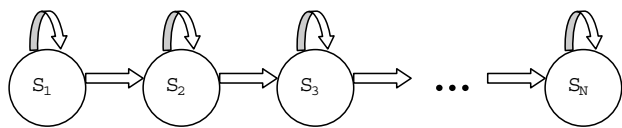
Fig. 6. State transition of HMMs for palmprint identification

5   Experimental Results

In our experiments, we made use of a database of 3,200 palmprint images from 320 palms. Ten images were captured from each palm. Six of these were used to train the HMM for the palmprint class and the remaining four samples were employed in the tests. Fig. 5 shows some samples used in the experiments. The HMM for each class is chosen as a left-to-right model whose state will transit to itself or its next adjacent state (Fig. 6). The identification accuracies of the proposed approach with different number of states and Gaussian mixtures are shown in Fig. 7. From this figure, identification rates of HMMs with 6 states and 2 Gaussian mixtures, with 7 states and 2 Gaussian mixtures, and with 8 states and 1 Gaussian mixture exceed 97%. Especially, when the number of the states and the number of the Gaussian mixtures in the HMMs are chosen as 6 and 2 respectively, the highest accuracy (97.80%) can be obtained.

6   Conclusion and Future Work

This paper has investigated the feasibility of HMMs on palmprint identification. The highest identification rate (97.8%) has been obtained when continuous HMMs with 6 states and 2 Gaussian mixtures were used. The good identification result encourages us to further investigate the proposed approach.

In the future work, the proposed approach will be tested using a larger palmprint database and some complex feature extraction methods will be developed for HMMs based palmprint identification.

Acknowledgements
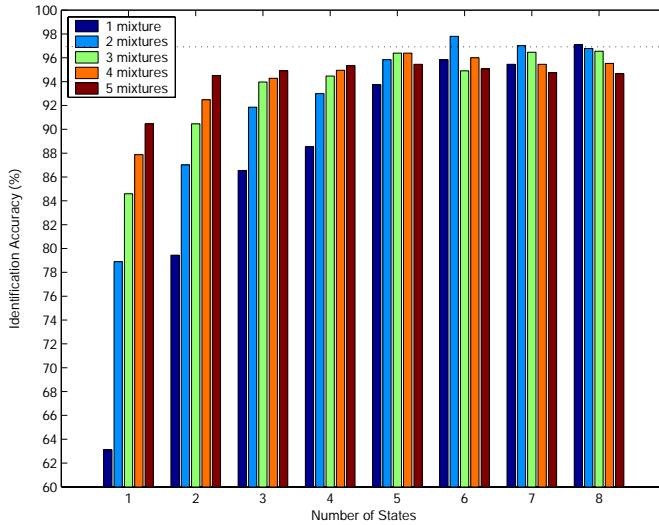
Fig. 7. The identification accuracies of the proposed approach with different number of states and different number of Gaussian mixtures

References

[1] Jain, A., Ross, A., Prabhakar, S.:An introduction to biometric recognition. IEEE Transaction on Circuit and System for Video Technology, Vol. 14, No. 1. (2004) 4–20

[2] Duta, N., Jain, A., Mardia, K.: Matching of palmprint. Pattern Recognition Letters, Vol. 23, No. 4. (2001) 477–485.

[3] Zhang, D., Kong, W., You, J., Wong, M.: Online palmprint identification. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 9. (2003) 1041–1050.

[4] Han, C., Chen, H., Lin, C., Fan, K.: Personal authentication using palm-print features. Pattern Recognition, Vol. 36, No. 2. (2003) 371–381.

[5] Kumar, A., Wong, D., Shen, H., Jain, A.: Personal Verification using Palmprint and Hand Geometry Biometric. Lecture Notes in Computer Science, Vol. 2688. (2003) 668–678.

[6] Rabiner, L.: A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, Vol. 77, No. 2.(1989) 257–286.

[7] Jeng, B., Chang, M., Sun, S., Shih, C., Wu, T.: Optical Chinese recognition with a hidden Markov model classifier. Electronics Letters, Vol. 26, No. 18. (1990) 1530–1531.

[8] Yu, K., Mason, J., Oglesby, J.: Speaker recognition using hidden Markov models, dynamic time warping and vector quantisation. IEE Proceedings on Vision, Image and Signal Processing, Vol. 5. (1995) 313–318.

Personal Identification and Verification: Fusion of Palmprint Representations

Carmen Poon, David C. M. Wong, and Helen C. Shen

Department of Computer Science, Hong Kong University of Science and Technology
Clear Water Bay, Hong Kong.
{carmenp,csdavid,helens}@cs.ust.hk

Abstract. This paper aims to study the accuracy and robustness of personal identification or verification systems where palmprint is the only modality available or utilized. Three different representations of palmprint are fused at the score-level by the sum rule, and at the decision-level by weighed or majority votes. Results showed that fusion at the score-level is easier to formulate and justify, and performs better than fusing at the decision-level. On a database of 340 subjects (10 samples/class), 10-fold and 2-fold cross-validation is accurate to 99.8% and 99.2% respectively. When operating as a verification system, it can achieve a false acceptance rate of 0.68% while maintaining a false rejection rate of 5%.

1   Introduction

Amongst the various biometrics used for personal identification or verification, palmprints are considered to have high uniqueness, fast processing time, and low cost [1, 2, 3]. Inkless palmprint images can be captured by CCD cameras [3, 4] or scanning technologies [5]. Typical methods in processing them are: directional lines detection [1, 5, 6]; Fourier transform [4]; Karhunen-Loeve transform [7]; wavelets transform [8, 9]; Gabor filters [3]; and feature space dimensionality reduction by Fisher's linear discriminant [2]. However, the palmprint capturing devices currently available require users to lay their hands on a contact surface against pegs for fixing the rotation and the stretching of the palm. The surface required frequent cleaning for hygienic, performance and security reasons. To overcome the shortfalls, we propose a contact-less palmprint capturing system that works without pegs (see Fig. 1).

In spite of the success of many single modality biometric systems, fusion of multiple representations, matchers, or modalities to improve accuracy or flexibility are inevitably the trend of modern security systems [10]. Fusion can be carried out at different stages of an identification or verification process, which typically comprises the steps of: 1) *feature extraction*, 2) *matching score calculation*, and 3) *decision making*. Fusion at each stage are, correspondingly: 1) at the *representation-level*, where feature vectors are concatenated before putting forward to a classier; 2) at the *score-level*, where scores resulted from each independently-classified feature vector are combined to derive the final decision; and 3) at the *decision-level*, where decisions are made based on each feature vector, from which the final conclusion is drawn.

Fig. 1. Palmprint Capturing System   **Fig. 2**. Schematic diagram of image alignment. Gaps between the fingers are used to minimize rotation and translation errors
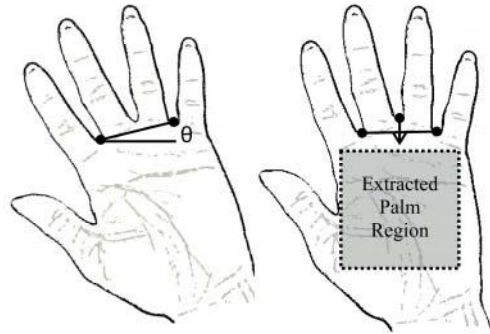
In this paper, fusion at the score-level and decision-level of three representations of palmprint are studied. Fusing at the representation-level is not applicable in our case since the varying length of our feature vectors made it difficult to scale each feature fairly at the representation-level. By fusing different representations of palmprint, it is hoped that the performance of biometric systems that uses palmprint as one of their biometric modals can be improved as a result.

2   Methodology

The three representations utilize: 1) a 2D Gabor filter, 2) a set of oriented line detectors, and 3) Haar wavelet transform. To extract features from a local spatial domain, palm images are either divided into fixed number of overlapping rectangular blocks, or fixed size, non-overlapping sectors of elliptical half-rings.

The image of the hand captured on a uniform, dark background is identified by a threshold determined statistically. With reference to the gaps-between-fingers, the palm is duly rotated and the maximum palm area is selected [11] (see Fig. 2). A shade model is constructed by applying a lowpass filter to the image and is subtracted from the original palm image to minimize the effects of non-uniform illumination.

2.1 2D Gabor Filter

The 2D Gabor filter family is inherited from the receptive field profiles encountered experimentally in cortical simple cells [12]. The general functional form of it is:

$$G(x, y) = e^{\left(-\pi \left[\frac{(x-x_o)^2}{\alpha^2} + \frac{(y-y_o)^2}{\beta^2} \right] \right)} \cdot e^{\left(-2\pi i \left[u_o(x-x_o) + v_o(y-y_o) \right] \right)} \tag{1}$$

Previous studies suggest that a Gabor filter orientated at 135° is sufficient and best for capturing features of a left-hand palmprint [3]. In this study, the spatial frequency $f_o = \sqrt{u_o + v_o}$ is set to best respond to lines of 10 pixels wide, which corresponds to

that of the majority palm lines found in our database. (α, β) specify the radial and angular bandwidths in the frequency domain, and are chosen to be [13]:

$$\alpha = \beta = \frac{3\sqrt{2\ln(2)}}{\sqrt{2}\pi f_o} \tag{2}$$

After applying a 15x15 (in pixels) Gabor filter, the palm image is converted into a binary image using an adaptive threshold estimated from the histogram. The resultant image is divided into 11x11 overlapping rectangular blocks such that each rectangular block overlaps half of its adjacent blocks. The number of pixels detected as a line in each rectangular block forms the feature vector.

2.2 Detection of Directional Lines

A set of line detectors oriented from 0° to 165°, in steps of 15°, is used to extract features from palmprints. For each pixel, the algorithm selects the line detector that matches most with the structural orientation in the local vicinity of the pixel-of-interest and stores the resultant magnitude in a separate map. Pixels with magnitude below an adaptive threshold that is determined from the histogram of the magnitude map are excluded and only representative features are retained.

The palm image is divided into non-overlapping elliptical half rings, which centered at the same point and with the two axes of each ring doubles that of its immediate inner one [11] (see Fig. 3). The innermost ellipse is divided into 3 sectors, while each outer ring will have 2 more sectors than its inner layer.

For each sector, the percentage of pixels in each of the four orientations, i.e. oriented between 0°-45°, … , 135°-180°, are used as features to represent the palmprint. Arrangement of the features is such that those obtained from an inner layer precede those obtained from an outer layer (see Fig. 4). This ensures that when two feature vectors of unequal length are compared to each other, point-wise comparison of them is actually comparing the same spatial region of two different palm images.
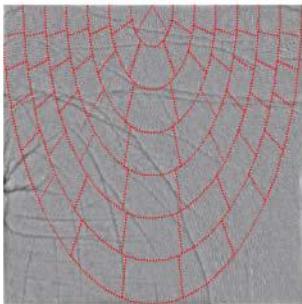
Fig. 3. Palm image is divided into sectors of elliptical half-ring
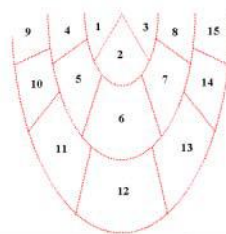
Fig. 4. Arrangement of feature vector

2.3 Haar Wavelet Transform

Palm images are decomposed by the Haar wavelet for a single-level. Smoothing masks are applied before and after wavelet transform. Since it was found that most of the low frequency components are attributable to the redness underneath the skin and should preferably be excluded from features for identification, magnitude of pixels within one standard deviation are set to zero. Values of the rest of the pixels are projected onto a logarithmic scale in order to minimize variations between two images:

$$I(x_i, y_i) = \begin{cases} 0, & if \ | I(x_i, y_i) | \le std(I(x, y)) \\ \ln(\ | I(x_i, y_i) | - std(I(x, y)) + 1 \ ), & if \ | I(x_i, y_i) | > std(I(x, y)) \end{cases}, \tag{3}$$

where $I(x,y)$ is the intensity of a detailed image.

Each of the detailed images is divided into sectors of non-overlapping elliptical half-rings (see Sect. 2.2). Mean energy level of each sector forms the feature vector, which is arranged in such a way that those extracted from an inner layer precede those from an outer layer (see Fig. 4).

2.4 Distance Measures

For each representation, the score between two feature vectors is calculated as the mean of the absolute difference between them to accommodate for the varying dimensionality. If $featureV_k$ represents a feature vector of N_k elements for the kth image, the score between the ith and the jth images is given as:

$$Score(i, j) = \frac{\sum_{n=1}^{min(N_i, N_j)} | featureV_i(n) - featureV_j(n) |}{min(N_i, N_j)} \tag{4}$$

3   Experiments

3.1 Analysis of Identification Systems

Fusion of the three methods is examined at the score-level and at the decision-level. Since the number of classes outstands the number of samples per class, a classifier that is not statistically based is chosen, i.e. the nearest-neighbor (1-NN). The system is tested with 10-fold and 2-fold cross-validation methods.

The *sum rule* is used when fusing at the score-level. Scores resulted from the three representations are first rescaled to a comparable scale by normalizing the genuine score distribution to zero mean and unit variance based on the distribution estimated from the training set. The final score is the sum of the normalized scores.

The *weighed vote* scheme is examined when fusing at the decision-level. The classes are ranked in ascending order of scores and the first three classes with maximum confidence will be given weights w_1, w_2, and w_3, respectively. The weight W is chosen randomly, while w_2 and w_3 are selected accordingly to satisfy:

$$w_1 = W$$
$$\frac{1}{2}W < w_2 < W$$
$$\frac{1}{3}W < w_3 < W \text{ , and, } \frac{1}{2}w_2 < w_3 < w_2$$

$$,\qquad(5)$$

We have chosen the three weights .9, .5, and .31. A slight variation of the weighed vote scheme, where each representation method is allowed to cast three third-picks (i.e. 5 votes/representation), is also examined.

3.2 Analysis of Verification Systems

The sum rule is used to fuse the three representations at the score-level. The sum of the normalized scores of the three representations is analyzed.

To fuse at the decision-level, nine images per class are used for training while the remaining one is resided for testing. The score at equal error rate (EER) is obtained from the training set for each representation. A representation method accepts a test sample only if the resultant score is below the threshold. The overall system will accept or reject the test sample based on the majority vote.

4   Results

Hand images are captured with resolution of 1280x960 (in pixels) and 8-bit colors. Ten images each of the left and right hand of the 170 individuals were captured to form a database of 340 subjects (right hands are flipped around the vertical axis and stored as another subject).

4.1 Performance of Identification Systems

It is found that representing palmprint by either one of the three representations results in accuracy differs by less than 1% (see Table 1). On the other hand, classification rates can be improved by as much as 2.2% in 2-fold cross-validation with information fusion (see Table 1).

85% of the incorrectly identified images are the same regardless of the fusion-level, suggesting that the two fusion rules presented in this study are correlated.

4.2 Performance of Verification Systems

The Haar wavelet method performs notably better than the other two methodologies when false rejection rate (FRR) is to be kept below 3% (genuine acceptance rate > 97%) (see Fig. 10). It is the only method that can maintain 90% genuine acceptance rate when false acceptance rate (FAR) is at least 0.68% (see Table 2).

Table 1. Classifiation rate of the three algorithms

	Individual Representation			Fusion of Different Representations		
	2D Gab. Filter	Dir. Lines	Haar Wavelet	Score-Level (Sum Rule)	Decision-Level (3 votes/Rep.)	Decision-Level (5 votes/Rep.)
10-fold	99.2%	99.3%	99.6%	99.8%	99.7%	99.8%
2-fold	97.0%	97.3%	97.9%	99.2%	98.8%	99.0%
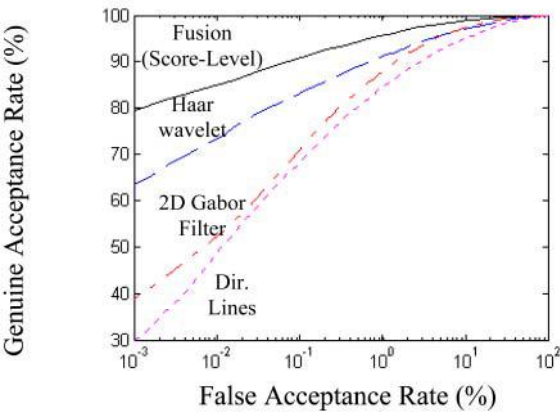
Fig. 5. Receiver operating characteristic (ROC) curves

Table 2. EER, FAR, and FAR of the three algorithms

	Individual Representation			Fusion at Score-Level (Sum Rule)	Fusion at Decision-Level
	2D Gab. Filter	Dir. Lines	Haar Wavelet		
EER (threshold)	4.7% (80.2)	6.5% (0.990)	4.6% (1.91)	2.6% (4.22)	3%-3.5%
FAR @ FRR=3%	8.6% (86.5)	19% (1.13)	9.8% (2.04)	2.0% (4.18)	–
FAR @ FRR=5%	4.3% (79.3)	10% (1.04)	4.0% (1.89)	0.72% (9.05)	–
FRR @ FAR=1.8%	8.8% (72.0)	12% (0.877)	7.1% (1.79)	3.2% (9.16)	4.5%
FRR @ FAR=0.68%	15% (65.2)	18% (0.806)	10% (1.69)	5.1% (9.04)	–

Fusion at the score-level greatly improves the verification accuracy. EER is reduced sharply to 2.6%. This corresponds to 95% genuine acceptance rate when FAR is 0.68% (see Fig. 10). To fuse at the decision-level, majority vote of the three representations results in 4.5% FRR and 1.8% FAR. EER is estimated to be around 3% to 3.5%. It must be noted that the estimation relies on three independent thresholds and therefore it is hard to obtain the true minimum EER. It is therefore suspected that fusion at the decision-level can hardly be justified and most likely it cannot perform better than fusing at the score-level.

5   Conclusion

In this paper, we have shown that fusing different representations of palmprint can result in a more accurate identification or verification system.

Fusion at the score-level (by the sum rule) and fusion at the decision-level (by weighed or majority votes) agree with each other, but the former is easier to formulate and justify, and has a better performance. When tested the fused system with a database of 340 subjects (10 samples per subject), 10-fold and 2-fold cross validation method results in classification rates of 99.8% and 99.2% respectively. When used as a verification system, it can operate at a false acceptance rate of 0.68% with an acceptable genuine acceptance rate of 95%. Equal error rate of the system is 2.6%.

For future development, we shall explore systems that combine the fusion of multiple representations of palmprints with other biometrics modalities so that optimal accuracy can be obtained while a certain degree of flexibility to users can also be retained.

Acknowledgement

References

[1]   Kumar, A., Wong, D.C.M., Shen, H.C., Jain, A.K.: Personal Verification using Palmprint and Hand Geometry Biometric. Proc. Int. Conf. AVBPA. Guildford, UK (2003) 668-678

[2]   Wu, X.Q., Zhang, D., Wang, K.Q.: Fisherpalms based palmprint recognition. Pattern Recognition Letters, Vol. 24(15). (2003) 2829-2838

[3]   Zhang, D., Kong, W.K., You, J., Wong, M.: Online Palmprint Identification. IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 25(9). (2003) 1041-1050

[4]   Li, W., Zhang, D., Xu, Z.: Palmprint identification by Fourier transform. Int. J. Pattern Recognition and Artificial Intelligence, Vol. 16(4). (2002) 417-432

[5]   Han, C.C., Cheng, H.L., Lin, C.L., Fan, K.C.: Personal authentication using palm-print features. Pattern Recognition, Vol. 36(2). (2003) 371-381

[6]   Zhang, D., Shu, W.: Two novel characteristics in palmprint verification: datum point invariance and line feature matching. Pattern Recognition, Vol. 32. (1999) 691-702

[7]   Lu, G., Zhang, D., Wang, K.: Palmprint recognition using eigenpalms features. Pattern Recognition Letters, Vol. 24(10). (2003) 1463-1467

[8]   Kumar, A., Shen, H.C.: Recognition of palmprints using wavelet-based features. Proc. Int. Conf. System and Cybernetics, SCI-2002, Orlando, Florida, Jul. (2002)

[9]   Wu, X.Q., Wang, K.Q., Zhang, D.: Wavelet Based Palm print Recognition. Proc. 1st Int. Conf. Machine Learning and Cybernetics, Vol. 3. (2002) 1253-1257

[10]   Ross, A., Jain, A.K.: Information fusion in biometrics. Pattern Recognition Letters, Vol. 24. (2003) 2115-2125

[11]   Poon, C., Wong, D.C.M., Shen, H.C.: A New Method in Locating and Segmenting Palmprint into Region-of-Interest. 17th Int. Conf. Pattern Recognition, (submitted, Jan. 2004).

[12]   Daugman, J.G.: Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression. IEEE. Trans. Acoustics, Speech, and Signal Processing, Vol. 36(7). (1988) 1169-1179

[13]   Kumar, A., Pang, G.: Fabric defect segmentation using multichannel blob detectors. Optical Engineering, Vol. 39(12). (2000) 3176-3190

Towards an Automated Dental Identification System (ADIS)

Gamal Fahmy[1], Diaa Nassar[1], Eyad Haj-Said[1], Hong Chen[2],
Omaima Nomir[3], Jindan Zhou[3], Robert Howell[1], Hany H. Ammar[1],
Mohamed Abdel-Mottaleb[3] and Anil K. Jain[2]

[1] West Virginia University
[2] Michigan State University
[3] University of Miami

Abstract. This paper addresses the problem of developing an automated system for postmortem identification using dental records. The Automated Dental Identification System (ADIS) can be used by law enforcement agencies to locate missing persons using databases of dental x-rays. Currently, this search and identification process is carried out manually, which makes it very time-consuming and unreliable. In this paper, we propose architecture for ADIS, we define the functionality of its components, and we briefly describe some of the techniques used in realizing these components.

1   Introduction

Law enforcement agencies have been exploiting biometric identifiers for decades as key tools in forensic identification. With the evolution in information technology and the huge volume of cases that need to be investigated by forensic specialists, automation of forensic identification became inevitable. Forensic identification may take place prior to death and is referred to as *Antemortem* (AM) identification. Identification may as well be carried out after death and is called *Postmortem* (PM) identification. While *behavioral* characteristics (e.g. speech) are not suitable for PM identification, most of the *physiological* characteristics are not appropriate for PM identification as well, especially under severe circumstances encountered in mass disasters (e.g. airplane crashers) or when identification is being attempted more than a couple of weeks postmortem. Therefore, a postmortem biometric identifier has to survive such severe conditions and resist early decay that affects body tissues. Dental features are considered the best candidates for PM identification. This is due to their survivability and diversity. *Forensic odontology* is the branch of forensics concerned with identifying human individuals based on their dental features. Traditionally, *forensic odontologists* relied on the morphology of dental restorations (fillings, crowns, .. etc.) to identify victims. However, modern materials used in restorations and fillings have poor radiographic characteristics. Hence, it is becoming important to make identification decisions based on inherent dental features like root and crown morphologies, teeth size, rotations, spacing between teeth and sinus patterns.

Based on the information provided by experts from the Criminal Justice Information Services Division (CJIS) of the FBI, there are over 100,000 unsolved

Missing Person cases in the National Crime Information Center at any given point in time, 60 percent of which have remained in the computer system for 90 days or longer, [12]. It is worth mentioning that the computing systems developed and maintained by CJIS are used by more than 94,000 agencies. CJIS includes in its strategic plan the creation of an Automated Dental Identification System (ADIS), with similar goals and objectives to its Automated Fingerprint Identification System (AFIS) but using dental/teeth characteristics instead of fingerprints.

The ADIS will provide automated search and matching capabilities for digitized x-ray and photographic images. Cavities in today's children and their children will be virtually unknown due to the advances in dentistry [1], [2], [3]; e.g. pit and fissure sealants, and non-radiographic crowns.

2   Background

There are several computer-aided PM identification systems. *CAPMI* [4] and *WinID* [5] are the most famous among these systems. However, these systems do not provide high level of automation, as feature extraction, coding, and image comparison are still carried-out manually. Moreover, the dental codes used in these systems only capture artificial dental work. ADIS is a process automation tool, for postmortem identification, that is being designed to achieve accurate and timely identification results with minimum amount of human intervention. To this end, ADIS will not only automate some of the steps taken by forensic experts to examine missing and unidentified persons (MUP) cases, it will also be intelligently analyzing radiographs to utilize underlying image structures that are often difficult to be assessed merely by visual examination.

3   The Proposed System

3.1   Functionality

At a high level of abstraction, the functionality of the ADIS prototype can be described by one of two uses:

1) Identification of a subject, or
2) Maintaining the system

3.1.1   Identify Subject

The process of identifying a subject is initiated when a user submits the subjects' record (containing image and non-image information). The result of Identification should be a "*short*" list of candidates whose reference records are retrieved from the dental image Repository (DIR) and presented to the forensic expert who takes a final decision about the identity of the subject Missing or Unidentified Person (MUP).

3.1.2   Maintaining the System

The process of maintaining the system primarily includes: Updating the repositories (adding/purging) reference records, Evaluating the performance of the system, Updating the matching techniques (retraining, changing system parameters, changing component realizations), Reporting problems, or substandard performance.

3.2   System Scenarios

In this Demo, different components of the ADIS architecture (shown in Fig. 1.) will be demonstrated.

   In ADIS, we will use the dental codes that currently exist in the National Crime Information Center (NCIC) to enhance the retrieval and matching speed and performance, these codes are provided by CJIS. We note here that these NCIC codes are manually extracted from MUP. Although ADIS is designed to be an automated system, it will also allow the use of the NCIC database for results refinements. In order to populate the DIR, high-level features are extracted from all the reference images and are stored in the DIR.

   When the user submits a subject record, that is typically an unidentified PM record, two processes occurs simultaneously

i)   NCIC codes are extracted from the subject image (this step is not mandatory in ADIS, these codes can also be extracted automatically through ADIS techniques)
ii)   High level features are extracted from the images, these features are extracted through the potential matching component as shown next

   The NCIC extracted codes can help in narrowing down the search list from the order of hundreds of thousands to the order of hundreds, however this is an optional step. High-level features are used to retrieve a list potential match from the DIR. This process happens through the potential search component presented next. This list of potential matches (candidate images) will be in order of 100-150 records.

   The potential search matching component is based on feature extraction and archival retrieval techniques, in which a set of features are extracted from the images, encoded as a feature template, then used as a query for searching the dental feature database for the records with the most similar features, [6]. The dental features database will be created from the dental image database by extracting dental features from the dental images, coding them into feature templates, then storing them onto the dental features database.

   Once the list of candidate images are created, they are fed to the image comparison matching stage. This stage processes the query image through enhancement and segmentation stages. Then this processed query image is matched against the list of candidate images that are provided from the potential matching component. This comparison matching stage aims at reducing the list of candidate images from the order of hundreds to the order of 30-35 images to be delivered to the forensic expert for a final identification decision. Once a positive identification is made, for a subject image, then its corresponding reference image if removed from the DIR. If a positive identification can't be made for a subject image, then this image is added to the DIR as a new missing or unidentified image. In our system, typically if the subject record

is AM, then the reference records are PM and vice versa. However we will also allow both subject and reference images to be AM, this is to avoid database redundancy if a AM record of a missing person was submitted multiple times.

4   The Proposed Architecture of the System

In this section we will briefly describe the main three components in ADIS, the DIR, the Potential Search Matching component and the image comparison component, as in figure 1. The upper parts of the architecture is used to archive the DIR with reference records. The lower part of the architecture is used for identifying a subject record. This lower part consists of the potential search matching component that retrieves a candidate list from the DIR and an image comparison that identifies the short match list to the forensic expert.

4.1   The DIR

The Digital Image Repository (DIR) is a central system, which allows the voluntary submission, storage, and retrieval of digital images relating to missing and unidentified person investigations.
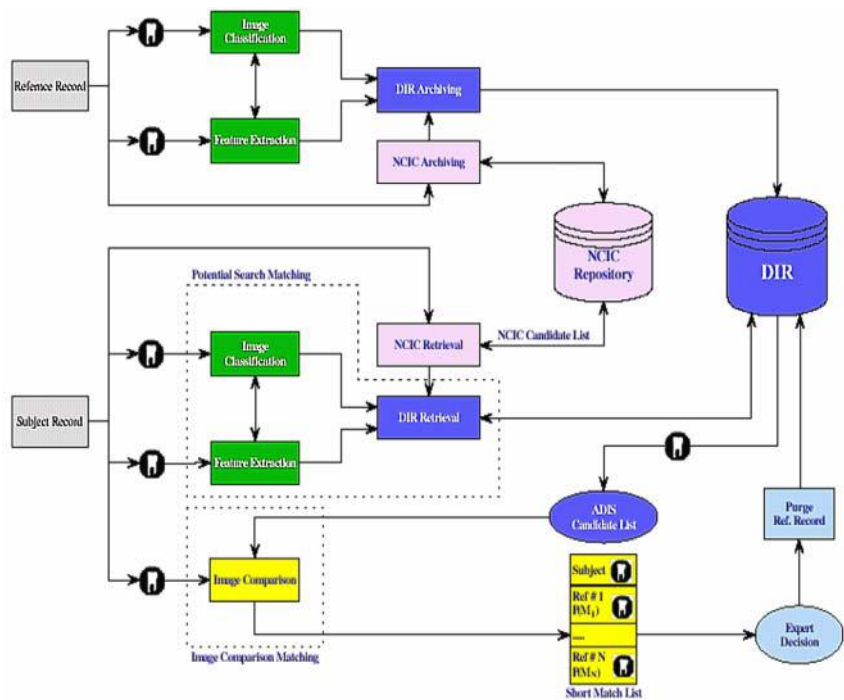
Fig. 1. Block Diagram of ADIS

The DIR will serve as an electronic centralized location where law enforcement agencies can submit digital images of dental x-rays, and other information regarding missing and unidentified person cases. It will eliminate the need to obtain x-rays from other law enforcement agencies for identification purposes. In addition it will help the forensic scientist to access the DIR in order to study the available images for quickly determining exact match. The repository will contain image information as well as non-image information for each missing/unidentified person. The procedure of DIR archiving is carried out after extracting dental features from each case's dental images. Here we list some of the high level features that will be extracted from each tooth

1. The shape and size (width, length) of each tooth.
2. The root features: number, curvature, and length of each one.
3. Existing/Extracted tooth.
4. Teeth spacing, the spacing between two adjacent teeth.

As all of these features are relative to the image size, they must be normalized. In addition to the dental features the DIR will contain information about each image and non-image information about each case such as case ID, if this record is for a *Postmortem* or *Antemortem*, date last update.

4.2   Potential Search Matching Component

This component extracts the high-level dental features and stores it in the DIR, then it retrieves a list of candidate images for every subject image. As noted before, this list is in the order of 100-150 records. In extracting the high level features, there are two methodologies adopted in ADIS, the first one focuses more on extracting the contour of roots and crowns that are usually important in periapical images, fig 3, [4]. The second methodology focuses on extracting the contour of each individual tooth in bite-wing images, fig 2, [4]. In this section we will briefly describe each methodology

4.2.1   Root and Contour Extraction for Periapical Images

This methodology involves three stages radiograph segmentation, Contour Extraction and Shape matching as described in detail in [7]. The radiograph segmentation stage utilizes the integral projection [9] histogram in horizontal directions to find the gap between the mandibular and maxillary teeth (Fig. 4) and the intergral projection histogram in vertical directions to find the gap between neighboring teeth (Fig. 5). The contour extraction stage has two sub-stages: crown shape extraction and root shape extraction. More details can be found in [7].
   In the Shape Matching stage, the shapes extracted from the query image must be matched to the shapes extracted from the database images. One of the main difficulties in matching AM and PM images is due to the fact that they were taken at different times often as long as several years. Thus the viewpoints are usually slightly different, which causes an *affine transformation* between the two images that must be considered prior to shape fitting. We confine ourselves with a subclass of the affine transformation. In particular, we do not consider the shear transformation, because the shear is negligible in AM and PM images. More details about shape matching can be found in [7]
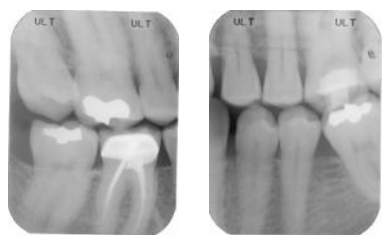
Fig. 2. Bitewing dental images
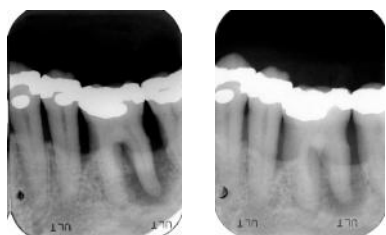
Fig. 3. Perapical dental images

4.2.2   Contour Extraction for Bitewing Images

In this methodology, Bitewing radiograph images are automatically enhanced and segmented. This methodology involves two main stages, Bitewing enhancement and bitewing segmentation.

In the bitewing enhancement stage, dental radiographs contain three distinctive regions: background, teeth, and bones. Usually the teeth regions have the highest intensity, the bone regions have high intensity that sometimes is close to that of the teeth, and the background has a distinctively low intensity. It is easy to separate the background by threshold-based methods, but these methods usually fail to discriminate teeth from bones, especially in cases of uneven exposure. To overcome this problem, the first step we use is to enhance the image's contrast. Top-hat and bottom-hat filters can be used to extract light objects (or, conversely, dark ones) on a dark (or light) but slowly changing background [8]. We use both the top-hat and the bottom-hat filters on the original image, and combine the results by adding to the original image the result of the top-hat filter, and subtracting the result of the bottom-hat filter, so that the teeth areas can be enhanced and the bone and background areas can be suppressed as well. To determine the gum lines we need to separate the bones of the upper and the lower jaws and identify the tips of the bones. From the detected bones images, the upper and lower jaws are separated using integral projection [9]. More details about this stage can be found in [11].
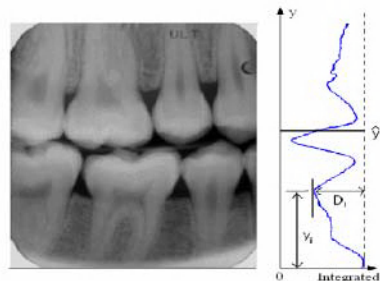
Fig. 4. Integral projection on the y axis
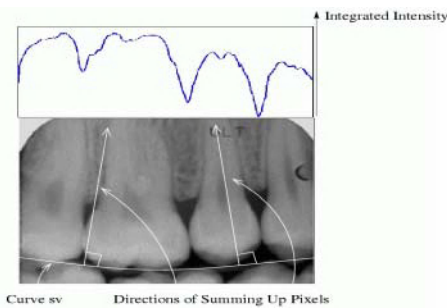
Fig. 5. Integral projection of pixels of the upper teeth along the lines perpendicular to the curve of the valley

4.3   Image Comparison Matching Component

In this component the radiograph images is processed through four main stages, enhancement, segmentation, alignment and matching. While the enhancement stage removes noise and enhances the quality of the radiograph images, the segmentation stage automatically segments the dental image into rectangular regions, where each region contains each tooth with its surroundings. The alignment stage registers/aligns different regions independent of their orientation, scale and translation. Theses regions are then matched through a neural network based matcher as shown next. We here give a brief description of all stages except the alignment stages that is still evolving.

4.3.1   Segmentation

In this stage, we only consider segmenting radiograph images into regions, where each region contains each individual tooth in it, rather than extracting the exact contour as in previous sections. This is due to the nature of the matcher that deals with regions and matches them with no need for the exact root or crown contour. We utilized the morphological filtering approach in this segmentation stages as described in [8].

4.3.2   Matcher

In this stage, we consider the decision-making stage of the prototype ADIS. The input to the decision-making stage is a pair of preprocessed representative segments from a candidate and a reference radiographs. The decision-making stage extracts embedded features from the input pair, and then determines their matching probability based on the measured differences between extracted features. As we previously mentioned, this stage is composed of two layers: the feature extraction and the decision. More details about the matching stages can be found in [10].

Acknowledgement

References

[1]   The Canadian Dental Association, Communique. May/June 1997.
[2]   P. Stimson & C. Mertz, Forensic Dentistry. CRC Press 1997.
[3]   Gustafson & Ghosta, Forensic Odontology. American Elsevier Pub. Co. 1996.
[4]   United States Army Institute of Dental Research, " Computer Assisted Post Mortem Identification via Dental and other Characteristics" – USAIDR Information Bulletin, vol. 5, No.1.
[5]   Jim McGivney et al., WinID2 software. (http://www.winid.com).

Gamal Fahmy et al.

[6]   A.K. Jain, R. Bolle and S. Pankanti, Biometrics: Personal Identification in a Networked Society, Kluwer, 1999.
[7]   K. Jain and H. Chen, "Matching of Dental X-ray images for Human identification", to appear in journal Pattern Recognition.
[8]   L.Vincent: Morphological Grayscale Reconstruction in Image Analysis: Application and Efficient Algorithms. IEEE Transactions on Image Processing, 2(2): 176-201, April 1993
[9]   R. Brunelli and T. Poggio, Face Recognition: Features versus Templates, IEEE Trans on PAMI, (15)10 : 1042-1052, 1993.
[10]   Hany H. Ammar, and Diaa M. Nassar, "A Neural Network based System for Dental Radiograph Comparison", *ISSPIT02*, 2002.
[11]   Omaima Nomir, Mohamed Abdel-Mottaleb, Diaa Eldin Nassar, Gamal Fahmy, and Hany H. Ammar, "Challenges of Developing an Automated Dental Identification System", accepted for publication in the IEEE mid-west symposium for circuits and systems in Cairo, Egypt, Dec 2003.
[12]   Robert Howell, "Dental Remains", Dental Task Force Technical Report by the CJIS of the FBI on Forensic Identification.

Author Index