Lecture Notes in Computer Science 3781

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

Editorial Board

Stan Z. Li   Zhenan Sun   Tieniu Tan
Sharath Pankanti   Gérard Chollet
David Zhang (Eds.)

Advances in Biometric Person Authentication

International Workshop
on Biometric Recognition Systems, IWBRS 2005
Beijing, China, October 22-23, 2005
Proceedings

Springer

Volume Editors

Stan Z. Li
Zhenan Sun
Tieniu Tan
Chinese Academy of Sciences
Center for Biometrics and Security Research
National Laboratory of Pattern Recognition
Institute of Automation
P.O. Box 2728, Beijing, 100080, China
E-mail: {szli, znsun, tnt}@nlpr.ia.ac.cn

Sharath Pankanti
IBM T.J. Watson Research Center
Exploratory Computer Vision Group
Yorktown Heights, NY, 10598
E-mail: sharat@us.ibm.com

Gérard Chollet
École Nationale Supérieure des Télécommunications
46, rue Barrault, 75634 Paris Cedex 13, France
E-mail: chollet@tsi.enst.fr

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

Preface

Automatic and reliable authentication of individuals is becoming an essential part of the modern world, for security and convenience in our life, in our work and in society. Biometrics-based systems utilize physiological or behavioral characteristics of an individual including the face, iris, fingerprint, palmprint, hand, voice, signature, or a combination of them, for this task. We are now seeing increasing interest and practical deployment of biometric systems.

The International Workshop on Biometric Recognition Systems (IWBRS 2005) was held in conjunction with ICCV 2005, providing an interactive forum for leading biometrics researchers and system designers. A biometric authentication competition (BAC) was conducted by the workshop to track the state-of-the-art biometrics technologies.

This volume of workshop proceedings includes 32 papers carefully selected from a total of 130 submissions. The papers address the problems in face, iris, fingerprint, palmprint, speech, writing and other biometrics, and contribute new ideas to research and development of reliable and practical solutions for biometric authentication.

We would like to express our gratitude to all the contributors, reviewers, and Program Committee and Organizing Committee members who made this a very successful workshop. We also wish to acknowledge the Institute of Automation, Chinese Academy of Sciences, and Springer for sponsoring this workshop. Special thanks are due to Miao Hong, Xin Yang, Zhuoshi Wei, Yinghao Cai, Zhaofeng He, Cheng Zhong, Shiqi Yu and Xianchao Qiu for their hard work in workshop organization.

We hope you could benefit from the fruitful workshop to improve the performance of your biometric systems.

September 2005

Stan Z. Li
Zhenan Sun
Tieniu Tan
David Zhang
Sharath Pankanti
Gerard Chollet

Organization

General Chairs

Anil K. Jain (Michigan State University, USA)
Josef Kittler (University of Surrey, UK)
Tieniu Tan (Institute of Automation, Chinese Academy of Sciences, China)

Program Chairs

David Zhang (Hong Kong Polytechnic University, Hong Kong, China)
Sharath Pankanti (IBM, USA)
Gerard Chollet (Ecole Nationale Supérieure des Télécommunications, France)
Stan Z. Li (Institute of Automation, Chinese Academy of Sciences, China)

Program Committee

Simon Baker (Carnegie Mellon University, USA)
Pawan Sinha (MIT, USA)
Samy Bengio (IDIAP, Switzerland)
Vijayakumar Bhagavatula (CMU, USA)
Volker Blanz (Max Planck Institute, Germany)
Ruud M. Bolle (IBM, USA)
John Daugman (University of Cambridge, UK)
Wen Gao (Institute of Computing Technology, CAS, China)
Jufu Feng (Peking University, China)
Julian Fierrez-Aguilar (Universidad Politecnica de Madrid, Spain)
Sadaoki Furui (Tokyo Institute of Technology, Japan)
Jiahie Kim (Yonsei University, Korea)
Seong-Whan Lee (Korea University, Korea)
Ales Leonardis (University of Ljubljana, Slovenia)
Shihong Lao (Omron Corporation, Japan)
Davide Maltoni (University of Bologna, Italy)
Shree K. Nayar (Columbia University, USA)
Mark Nixon (University of Southampton, UK)
Jonathon Phillips (NIST, USA)
Jamie Sherrah (Safehouse International Ltd., Australia)
Arun Ross (West Virginia University, USA)
Yang Ni (INT, France)
Xiaoou Tang (Microsoft Research Asia, China)
Yunhong Wang (Beihang University, China)
GuangYou Xu (Tsinghua University, China)

Wei Yun Yau (Institute for Infocomm Research, Singapore)
PC Yuen (Hong Kong Baptist University, Hong Kong, China)
Changsui Zhang (Tsinghua University, China)
Young-Bin Kwon (Chung-Ang University, Korea)

Competition Coordinators

Yin Xie (Automatic Identification Manufacture Association of China)

Local Arrangements Chairs

Zhenan Sun (Institute of Automation, Chinese Academy of Sciences, China)
Xin Yang (Institute of Automation, Chinese Academy of Sciences, China)

Table of Contents

Face

Fingerprint

Iris

Speaker

Writing

Texture Features in Facial Image Analysis[*]

Matti Pietikäinen and Abdenour Hadid

Machine Vision Group
Infotech Oulu and Department of Electrical and Information Engineering
P.O. Box 4500, FI-90014 University of Oulu, Finland
{mkp,hadid}@ee.oulu.fi
http://www.ee.oulu.fi/mvg

Abstract. While features used for texture analysis have been successfully used in some biometric applications, only quite few works have considered them in facial image analysis. Texture-based region descriptors can be very useful in recognizing faces and facial expressions, detecting faces and different facial components, and in other face related tasks. This paper demonstrates this issue by considering the local binary pattern (LBP) as an example of texture-based approach and showing its efficiency in facial image analysis.

1 Introduction

Texture is an important characteristic for the analysis of many types of images. Among the traditional application areas of texture analysis are industrial inspection, biomedical image analysis, analysis of remotely sensed images, and content-based retrieval from image databases. A textured area in an image can be characterized by a non-uniform or varying spatial distribution of intensity. The intensity variation reflects some changes in the scene being imaged. The specific structure of the texture depends on the surface topography and albedo, the illumination of the surface, and the position and frequency response of the camera.

A wide variety of techniques for describing image texture have been proposed [1]. The methods can be divided into four categories: statistical, geometrical, model-based and signal processing. Among the most widely used approaches are statistical methods based on co-occurrence matrices of second order gray level statistics, signal processing methods based on multi-channel Gabor filtering or wavelets, and model based methods using Markov random fields.

Texture could play an important role in many biometric applications [2]. The most notable example of recent success is iris recognition, in which approaches based on multi-channel Gabor filtering have been highly successful. Multi-channel filtering has also been widely used to extract features e.g. in fingerprint and palmprint analyses.

---

[*] The financial support of the Academy of Finland and the National Technology Agency of Finland is gratefully acknowledged

Some works have also considered texture features in facial image analysis. For instance, the well-known Elastic Bunch Graph Matching (EBGM) method is using Gabor filter responses at certain fiducial points to recognize faces [3]. Gabor wavelets have also been used in facial expression recognition yielding in good results [4]. A problem with the Gabor-wavelet representations is their computational complexity. Therefore, simpler features like Haar wavelets have been considered in face detection resulting in a fast and efficient face detector [5].

Recently, the local binary pattern (LBP) texture method has provided excellent results in various applications. Perhaps the most important property of the LBP operator in real-world applications is its better tolerance against illumination changes than most of the other texture methods have. Another equally important property is its computational simplicity, which makes it possible to analyze images in challenging real-time settings [6–8]. In this paper, we will consider LBP features as examples to demonstrate the usefulness of texture-based approach in facial image analysis.

2   The Local Binary Pattern Approach to Texture Analysis

The LBP texture analysis operator is defined as a gray-scale invariant texture measure, derived from a general definition of texture in a local neighborhood. For each pixel in an image, a binary code is produced by thresholding its value with the value of the center pixel. Fig. 1 shows an example of an LBP calculation.
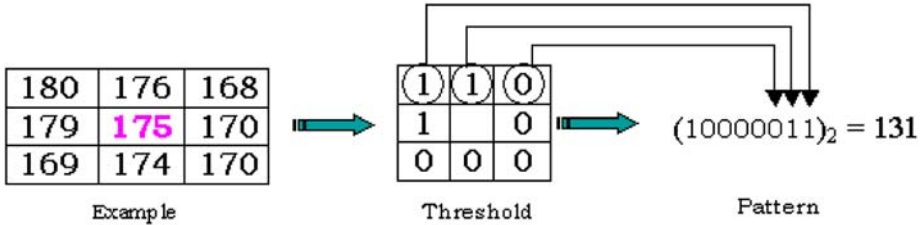
Fig. 1. Example of LBP calculation

A histogram is created to collect up the occurrences of different binary patterns. Each bin of the histogram (LBP code) can be regarded as a micro-texton. Local primitives which are codified by these bins include different types of curved edges, spots, flat areas etc. Fig. 2 shows some examples.

The basic version of the LBP operator considers only the eight neighbors of a pixel (Fig. 1)[6]. Later, the operator has been extended to consider different neighborhood sizes [7]. For example, the operator $LBP_{4,1}$ uses only 4 neighbors while $LBP_{16,2}$ considers the 16 neighbors on a circle of radius 2. In general, the operator $LBP_{P,R}$ refers to a neighborhood size of P equally spaced pixels on a circle of radius R that form a circularly symmetric neighbor set.
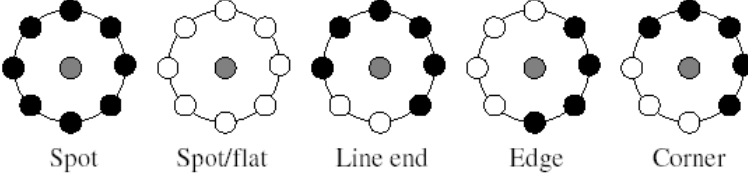
Fig. 2. Examples of texture primitives which can be detected by LBP (white circles represent ones and black circles zeros)

The LBP method has already been used in a large number of applications all over the world, including visual inspection, image retrieval, remote sensing, biomedical image analysis, face image analysis, motion analysis, environment modeling, and outdoor scene analysis. For a bibliography of LBP-related research, see [9] .

3 Texture Features in Facial Image Analysis: Case Studies

Applying a texture operator and considering only the response distributions as facial representation may yield in loss of spatial information. For efficiently representing the facial images, one should codify the texture information while retaining also their locations. One way to achieve that goal is to divide the face image into several regions and then extract (texture) features from each of them.

Using the local binary pattern (LBP) operator, for instance, the face image can be divided into several regions (or blocks) from which the local binary pattern histograms are computed and concatenated into a single histogram (see Fig. 3.a). In such a representation, the texture of facial regions is encoded by the LBP while the shape of the face is recovered by the concatenation of different local histograms. The idea of using LBP for face representation is motivated by the fact that face images can be seen as a composition of micro-patterns, such as those shown in Fig. 2, which can be well described by LBP.
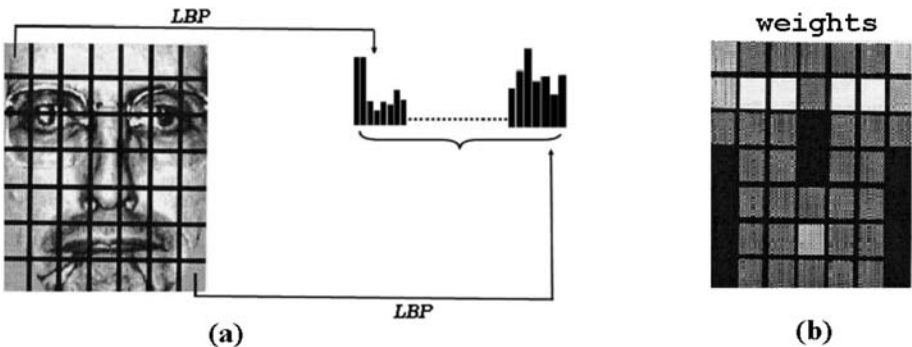
Fig. 3. (a) An LBP description. (b) The assigned weights: black squares indicate weight 0.0, dark grey 1.0, light grey 2.0 and white 4.0

3.1   Face Recognition

To analyze the performance of the texture based representation shown in Fig. 3.a, a face recognition system was built in [10]. The system uses a nearest neighbor classifier for recognition. A comparison against well known methods such as PCA, EBGM and Bayesian Intra/extrapersonal (BIC) was also done. To achieve a fair comparison, the FERET frontal face database and protocol were used.

In the LBP-based representation, each face image is represented by a texture feature histogram. In the nearest neighbor classifier, the χ^2 (Chi-square) dissimilarity metric is adopted for comparing a target face histogram S to a model histogram M: $\chi^2(S, M) = \sum_{i=0}^{l} \frac{(S_i - M_i)^2}{S_i + M_i}$, where l is the length of the feature vector used to represent the face image. Note that the choice of the χ^2 measure is motivated by the experimental findings [10] which showed that χ^2 gives better results than other dissimilarity measures such as the histogram intersection and Log-likelihood statistic.

When dividing the facial images into several regions, it can be expected that some of the regions contain more useful information than others in terms of distinguishing between faces. Therefore, one may use different weights, depending on the importance of the given regions in recognition. For instance, since the eye regions are important for recognition, a high weight can be attributed to the corresponding regions. Fig. 3.b shows the weights that were assigned to the different regions [10].

The extensive experiments clearly showed the superiority of the LBP-based approach over all considered methods (PCA, BIC and EBGM) on the FERET tests which include testing the robustness of the methods against different facial expressions, lighting and aging of the subjects. Additional experiments on the ORL face database (Olivetti Research Laboratory, Cambridge) have also showed a relative robustness with respect to alignment.

One may wonder whether the very good results are due to the division of the face images into local regions (instead of an holistic approach) or to the discriminative power of LBP features. This issue has been investigated in [11], by comparing the performance of four texture descriptors (gray-level difference histogram, homogeneous texture descriptor, texton histogram, and LBP) extracted from local regions. The performance of LBP was shown to be superior to the comparison algorithms. This confirms the validity of using LBP for face description. The main explanation for the better performance of the local binary pattern operator over other texture descriptors is its tolerance to monotonic gray-scale changes. Moreover, no gray-scale normalization is needed prior to applying the LBP operator to the face image.

Directions for future research include using learning algorithms (such as AdaBoost) for selecting the optimal subset of prominent LBP features extracted with different parameters. Some work on this issue has already been done in [12], in which the AdaBoost learning algorithm is used for finding optimal windows for recognition. Using this method promising recognition results are achieved with a smaller feature vector length. Recently, Zhang et al. proposed another direction consisting of combining texture operators. The approach [13] named

multi-resolution histograms of local variation patterns, first computes a multi-resolution and multi-orientation description of an image using Gabor decomposition. Then, LBP histograms are computed from the Gabor features for small non-overlapping regions and concatenated into a feature histogram. Excellent results are reported for the FERET database.

3.2 Face Detection

In [14], a variant of the LBP-based facial representation shown in Fig. 3.a is proposed for detecting frontal faces. A specific of this new representation is the use of overlapping regions and a 4-neighborhood LBP operator $LBP_{4,1}$ to avoid statistical unreliability due to long histograms computed over small regions. Additionally, the holistic description of a face is enhanced by including the global LBP histogram computed over the whole face image.

Investigating the performance of the new LBP representation in detecting frontal faces, training sets of face and non face images were first collected [14]. Then, the LBP facial representations were extracted from the training data and used to train an SVM classifier. After training, the system was run on several images from different sources to detect faces. Fig. 4 shows some detection results.

The experimental results on 80 test images from the MIT-CMU sets have showed that the LBP-based face detector compares favorably against the comparative approaches (Bayesian Discriminating Features, Schneiderman & Kanade method, and using normalized pixel values as inputs to an SVM classifier).
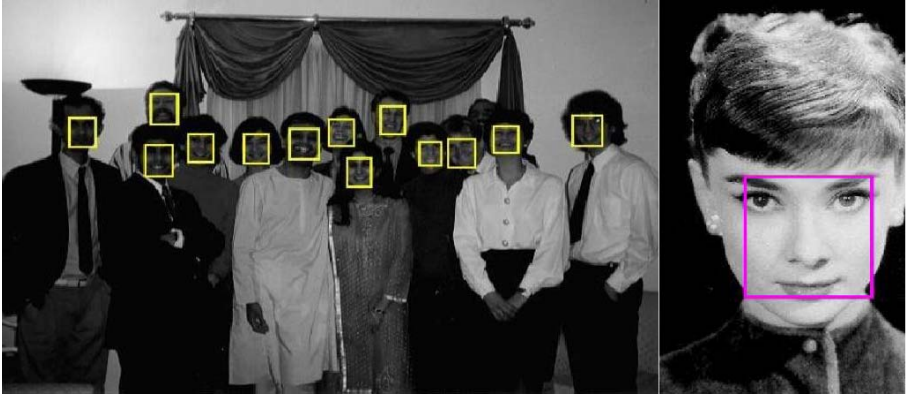
Fig. 4. Detection examples using LBP approach

Another variant of LBP-based facial representation, called "Improved LBP", has been used in [15]. Instead of using the extracted features as inputs to an SVM like in [14], the authors have considered a Bayesian framework for classifying the LBP representations. The face and non-face classes were modeled using multivariable Gaussian distributions while the Bayesian decision rule was used

to decide on the "faceness" of a given pattern. The reported results are very encouraging.

As an extension to the reported work, it would be of interest to consider the LBP features with learning algorithms, such as AdaBoost and its variants, in building a fast and robust multi-view face detector.

3.3   Facial Expression Recognition

In [16], an approach to facial expression recognition from static images was developed using LBP histograms computed over non-overlapping blocks for face description. The Linear Programming (LP) technique was adopted to classify seven facial expressions: anger, disgust, fear, happiness, sadness, surprise and neutral. During the training, the seven expression classes were decomposed into 21 expression pairs such as anger-fear, happiness-sadness etc. Thus, twenty-one classifiers are produced by the LP technique, each corresponding to one of the 21 expression pairs. A simple binary tree tournament scheme with pairwise comparisons is used for classifying unknown expressions. Good results (93.8%) were obtained for the Japanese Female Facial Expression (JAFFE) database used in the experiments. The database contains 213 images in which ten persons are expressing three or four times the seven basic expressions. Some sample images are shown in Fig. 5.

Fig. 5. Samples from original Japanese Female Facial Expression images

Another approach to facial expression recognition using LBP features is proposed in [17]. Instead of the LP approach, template matching with weighted Chi square statistic and SVM are adopted to classify the facial expressions using LBP features. Extensive experiments on the Cohn-Kanade database confirmed that LBP features are discriminative and more efficient than Gabor-based methods especially at low image resolutions.

3.4   Other Facial Image Analysis Tasks

The applicability of texture features in facial image analysis is not limited to the tasks presented above. A similar methodology could also be used, for instance, in detecting different facial components such eyes, mouth, nose etc. Fig. 6 shows an example of an LBP-based scheme for detecting the eyes. In addition, LBP features have also been considered for localizing and representing facial key points.
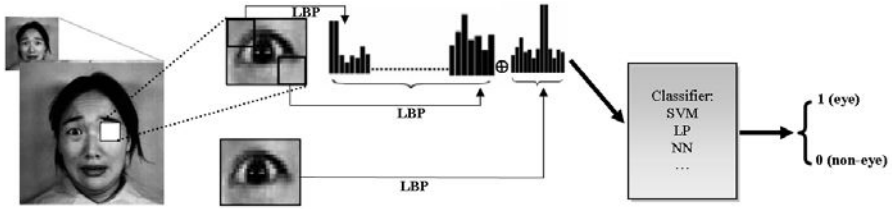
Fig. 6. Using LBP features for detecting facial components such as eyes

An accurate localization of such points of the face is crucial to many face analysis and synthesis problems such as face alignment. In [18], it was shown that using extended LBP features in the Active Shape Model (ASM) approach enhances the face alignment accuracy compared to the original method used in ASM.

4   Discussion

The results presented in this paper show that a texture-based approach can be very useful in various tasks of facial image analysis. Approaches based on Gabor filtering or wavelets measuring the frequency contents of facial image points or regions at different resolutions and orientations are often powerful, but computationally quite complex. The idea of using computationally simple LBP operator for face description is motivated by the fact that faces can be seen as a composition of micro-patterns (edges, lines, flat areas, spots etc.) which are well described by the operator. Combination of different texture approaches, like using Gabor filtering with LBP [13], could be one way to go ahead.

Using the dynamics of facial images could improve the accuracy of face and facial expression recognition. New approaches based on dynamic texture could be useful in this kind of problems.

Naturally texture-based approach has also limitations. Images taken at different times with different sensors in different illumination conditions are problematic, because under these changes the surface texture is likely to change. Proper methods for image preprocessing and features which are more robust against these kinds of transformations are needed.

References

1. Tuceryan, M., Jain, A.K.: Texture Analysis. In: Chen, C.H., Pau, L.F., Wang, P.S.P. (eds.): Handbook of Pattern Recognition and Computer Vision, 2nd edn. World Scientific (1999) 207-248
2. Wayman, J., Jain, A.K., Maltoni, D., Maio, D.: Biometric Systems: Technology, Design and Performance Evaluation. Springer (2005)
3. Wiskott, L., Fellous, J.-M., Kuiger, N., von der Malsburg, C.: Face Recognition by Elastic Bunch Graph Matching. IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (1997) 775-779

4. Tian, Y.-L., Kanade, T., Cohn, J.F.: Facial Expression Analysis. In. Li, S.Z., Jain, A.K. (eds.) Handbook of Face Recognition. Springer (2004) 247-275
5. Viola, P.A., Jones, M.J.: Robust Real-time Face Detection. International Journal of Computer Vision 57 (2004) 137-154
6. Ojala, T., Pietikäinen, M., Harwood, D.: A Comparative Study of Texture Measures with Classification Based on Feature Distributions. Pattern Recognition 29 (1996) 51-59
7. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2002) 971-987
8. Mäenpää, T., Pietikäinen, M.: Texture Analysis with Local Binary Patterns. In: Chen, C.H., Wang, P.S.P. (eds.): Handbook of Pattern Recognition and Computer Vision, 3rd edn. World Scientific (2005) 197-216
9. http://www.ee.oulu.fi/research/imag/texture/
10. Ahonen, T., Hadid, A., Pietikäinen, M.: Face Recognition with Local Binary Patterns. In: Computer Vision, ECCV 2004 Proceedings, Lecture Notes in Computer Science 3021 (2004) 469-481
11. Ahonen, T., Pietikäinen, M., Hadid, A., Mäenpää, T.: Face Recognition Based on the Appearance of Local Regions. In: 17th International Conference on Pattern Recognition (2004), Cambridge, UK, 3:153-156
12. Zhang, G., Huang, X., Li, S.Z., Wang, Y., Wu, X.: Boosting Local Binary Pattern (LBP)-Based Face Recognition. In: Advances in Biometric Person Authentication, SINOBIOMETRICS 2004 Proceedings, Lecture Notes in Computer Science 3338 (2004), 179-186
13. Zhang, W., Shan, S., Zhang, H., Gao, W., Chen, X.: Multi-resolution Histograms of Local Variation Patterns (MHLVP) for Robust Face Recognition. In: Audio- and Video-Based Biometric Person Authentication, AVBPA 2005 Proceedings, Lecture Notes in Computer Science 3546 (2005), 937-944
14. Hadid, A., Pietikäinen, M., Ahonen, T.: A Discriminative Feature Space for Detecting and Recognizing Faces. In: IEEE Conference on Computer Vision and Pattern Recognition (2004) II: 797-804
15. Jin, H., Liu, Q., Lu, H., Tong, X.: Face Detection Using Improved LBP Under Bayesian Framework. In: Third International Conference on Image and Graphics (ICIG 04), Hong Kong, China, 18-20 Dec 2004, 306-309
16. Feng, X., Pietikäinen, M., Hadid, A.: Facial Expression Recognition with Local Binary Patterns and Linear Programming. Pattern Recognition and Image Analysis 15 (2005) 550-552
17. Shan, C., Gong, S., McOwan, P.W.: Robust Facial Expression Recognition using Local Binary Patterns. In: IEEE International Conference on Image Processing (2005)
18. Huang, X., Li, S.Z., Wang, Y.: Shape Localization Based on Statistical Method Using Extended Local Binary Pattern. In: Third International Conference on Image and Graphics (ICIG 04), Hong Kong, China, 18-20 Dec 2004, 184-187

Enhance ASMs Based
on AdaBoost-Based Salient Landmarks Localization
and Confidence-Constraint Shape Modeling

Zhiheng Niu[1], Shiguang Shan[2], Xilin Chen[1,2], Bingpeng Ma[2,3], and Wen Gao[1,2,3]

[1] Department of Computer Science and Engineering,
Harbin Institute of Technology, Harbin, 150001, China
`zhniu@jdl.ac.cn`
[2] Institute of Computing Technology,
Chinese Academy of Sciences, Beijing, 100080, China
`{sgshan,xlchen,bpma,wgao}@jdl.ac.cn`
[3] Graduate School of the Chinese Academy of Sciences, Beijing, 100039, China

Abstract. Active Shape Model (ASM) has been recognized as one of the typical methods for image understanding. Simply speaking, it iterates two steps: profile-based landmarks local searching, and statistics-based global shape modeling. We argue that the simple 1D profile matching may not localize landmarks accurately enough, and the unreliable localized landmarks will mislead the following shape matching. Considering these two problems, we propose to enhance ASM from two aspects: (1) in the landmarks local searching step, we introduce more efficient AdaBoost method to localize some salient landmarks instead of the relatively simple profile matching as in the traditional ASMs; (2) in the global shape modeling step, the confidences of the landmark localization are exploited to constrain the shape modeling and reconstruction procedure by not using those unreliably located landmarks to eliminate their negative effects. We experimentally show that the proposed strategies can impressively improve the accuracy of the traditional ASMs.

1 Introduction

In most pattern recognition and computer vision tasks, the localization and alignment of target object from an image is a task of great importance. To deal with the problem, many methods have been proposed in recent years including active contour models (snake) [1], deformable template [2], elastic bunch graph matching [3], Gabor wavelet networks [4], Active Shape Models (ASMs) [5] and Active Appearance Models (AAMs) [6] etc. Among all these methods, ASMs and AAMs, both based on statistical models, has been recognized to be efficient and effective for image interpretation. Simply speaking, ASMs iterate two steps: profile-based landmarks local searching, and statistics-based global shape modeling. In the first step, local texture on the direction perpendicular to the contour, so-called profile, is exploited to model the local texture of each landmark and search for the landmarks locally. The global shape models are then applied in the second step to "correct" the local search result according to the statistical shape model.

The profile-based searching in ASMs provides an efficient way to localize landmarks, which has made ASMs computationally fast. However, due to the variations in pose, lighting, and expressions, local texture of the same landmark may vary dramati-

cally. So, the simple 1D profile model may not localize landmarks accurately enough for further process. Therefore, we argue that some elaborate local texture matching and searching method should improve the final performance of the ASMs, especially for those most significant landmarks. In this paper, AdaBoost based on Haar wavelets which has achieved great success in face detection [7] is introduced to localize some salient landmarks.

Furthermore, in the shape matching step of ASMs, all the landmarks are equivalently treated. However, one can intuitively imagine that some landmarks inherently cannot be located accurately, e.g., some contour points, while others may inherently be easy to localize, e.g., the eyes. It is easy to understand that the imprecisely located landmarks will mislead the following shape matching process. Aiming at this problem, this paper proposes to exploit the confidence of the local matching and reconstruct the shape model according to those reliably localized landmarks with high confidence. Thus, the possibly incorrect landmarks are not involved in the shape matching process.

2   Enhanced ASMs (EASMs)

The system overview of the proposed Enhanced ASMs is illustrated in Figure 1, in which two main contributions can be highlighted: one is the AdaBoost-based salient landmarks location, the other is the shape fitting based on only "reliable" landmarks. In the following two sections, we will take facial landmarks localization for example to describe these two contributions in detail. Readers can easily generalize it to the analysis of other objects.
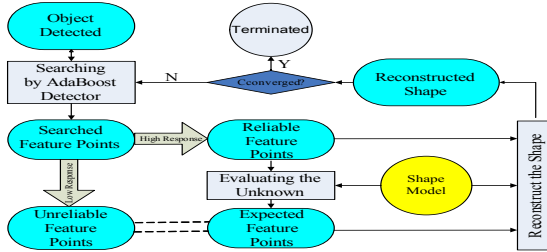
Fig. 1. System overview

2.1   AdaBoost-Based Salient Landmarks Localization

The Adaboost we used for salient landmarks localization is quite similar to that for face detection proposed by Viola and Jones [7]. The main difference lies in the adoption of more constraints when preparing the training data, especially the negative (non-landmark) examples, which are all derived from the neighborhood region of the specific facial landmark. This is reasonable considering that we need only to search these landmarks in the face region detected by a previous face detector, such as Viola and Jones' AdaBoost method. For the same reason, in AdaBoost searching, the candidate search region for each landmark is also restricted elaborately. Besides, all the AdaBoost-based landmark detectors can share the same integral image, which can further speed up the procedure.

For face case, we define totally 44 landmarks, 13 contour points and 31 inner points respectively. Since the contour points are intuitively more appropriate to localize by using profile-based local search, only the 31 inner points are localized using the abovementioned AdaBoost-based method.

Note that, in the iteration of the ASM searching, each AdaBoost-based landmark detector naturally provides a detection confidence indicating the "accuracy" of the detection, which will in turn determine whether the corresponding landmark can be used for sequential shape fitting.

Cristinacce and Cootes [8], [9] has also used boosted classifiers similarly, with a Shape Constrained manner, and achieve good experimental result.

2.2   Confidences-Constraint Shape Reconstruction

In traditional ASMs, once the landmarks are locally located in the local search step, a statistical model, pre-trained from a training set based on PCA, is acted on **all** the candidate landmarks to find a model best matching with the current candidate shape. This procedure can be formulated as follows:

$$\mathbf{x} = \overline{\mathbf{x}} + \mathbf{Pb} \qquad (1)$$

where $\mathbf{x} = \left[x_0, y_0, x_1, y_1, \ldots, x_{n-1}, y_{n-1} \right]^T$ is the shape vector concatenate the coordinates of all the landmarks, $\overline{\mathbf{x}}$ is the mean shape vector, P is the matrix containing the principle components of the shape vector, and b is the shape parameters of $\mathbf{x}$ when projected to the shape subspace.

In this paper, we argue that candidate landmarks with very low detection confidence may decrease the accuracy of the sequential model fitting. So, we propose to utilize only those landmarks with high enough confidence for model fitting.

Firstly, in the current iteration, all the candidate landmarks are categorized into two sets, reliable candidates set (RCS) and unreliable candidates set (UCS), by the following criteria:

$$\begin{aligned} L_i &\in RCS, \text{if } C_i > \theta_i \\ L_i &\in UCS, \text{if } C_i \leq \theta_i \end{aligned} \qquad (2)$$

where L_i denotes the ith landmark, C_i is the detection confidence of L_i, and θ_i is the threshold for L_i. And we assume the number of landmarks in RCS and UCS are r and u respectively. Then, we try to perform PCA by using only the r reliable landmarks, considering the u unreliable landmarks as missing data. Thus, in shape vector $\mathbf{x}$, there are $m = 2r$ elements are known and $k = 2u$ elements missing.

We have the transform matrix $\mathbf{P}$ that will transform the covariance matrix of $\mathbf{x}$ to diagonal matrix $\mathbf{\Lambda}$ which contains its eigenvalues, and $\mathbf{Q} = \mathbf{P}\mathbf{\Lambda}^{-1/2}$ that will transform it to identity matrix. Therefore the coordinates in variance-normalized space is denoted as $\mathbf{z} = \mathbf{Q}^T(\mathbf{x} - \overline{\mathbf{x}})$. There exists a row transform matrix $\mathbf{R}$, where $\mathbf{R}^T\mathbf{R} = \mathbf{R}\mathbf{R}^T = \mathbf{I}$, that will transform $\mathbf{x}$ to $\mathbf{x}' = \mathbf{R}^T(\mathbf{x} - \overline{\mathbf{x}})$ whose first m elements are known and the last k elements are unknown. The objective function we try to minimize is the variance-normalized distance from shape x to the mean shape $\overline{\mathbf{x}}$, it comes from the fact that the shorter the distance is the shapes are more like to be.

$$z^T z = \left[Q^T (x - \bar{x}) \right]^T Q^T (x - \bar{x}) = {x'}^T R^T Q Q^T R {x'}^T \tag{3}$$

It is actually a least square problem. We partition them as $x' = R^T (x - \bar{x}) = \begin{bmatrix} \alpha_m \\ \alpha_k \end{bmatrix}$

and $C = R^T Q Q^T R = \begin{bmatrix} C_{mm} & C_{mk} \\ C_{km} & C_{kk} \end{bmatrix}$, Where C_{ij} is $i \times j$ matrix and α_i is i-dimensional

column vector. α_k is the only independent variable of objective function. Finally we represent the unknown elements (unreliable points) by the know elements (reliable points) and the previous established shape model.

$$\hat{\alpha}_k = -C_{kk}^{-1} C_{km} \alpha_m \tag{4}$$

The predicted shape vector is:

$$\hat{x} = R\hat{x}' + \bar{x} = R \begin{bmatrix} \alpha_m \\ \hat{\alpha}_k \end{bmatrix} + \bar{x} \tag{5}$$

Radically our reconstructing of shape by a part of landmarks, reliable points, is a kind of PCA with missing data. We develop this simple but efficient method to solve the problem instead of complicated EM algorithm and Probabilistic Principal Component Analysis [10], [11].

3   Experiments and Analysis

We test the algorithm on two data sets. The first is the face database built by our lab. The images are 240*320, and with 103 manually labeled landmarks. We compare the method introduced in this paper and the conventional ASM on this data set. The second is a publicly available image set known as the BioID database with 20 manually labeled landmarks, used by D. Cristinacce and T. Cootes [9]. We have a comparison with their approach. Our algorithm relies on a global face detector to detect a rectangle of upright front face which is part of the algorithm's input. We use the face detector proposed by Viola and Jones [7].

3.1   Tests on Our Own Database

We test 400 images contained a frontal face on the first data set. We measure the positional error of each feature and express it as a proportion of the inter-ocular distance. The x-axis is the No. of the facial feature points, and the y-axis is the error distance. The mean error of our method is 0.047, while the conventional ASM is 0.084. Our method achieves a good accuracy.

We tests three different methods: the EASM with AdaBoost-based Searching and Confidence-Constraint Shape (ASM_{A+C}), ASM and ASM with AdaBoost-based Searching (ASM_A). Figure 2 shows our method is significantly accuracy and demonstrates that using reliable points to reconstruct shape make the algorithm more robust. This is because the mechanism of reconstructing shape decreases the effect of the improperly searched points to the final shape.
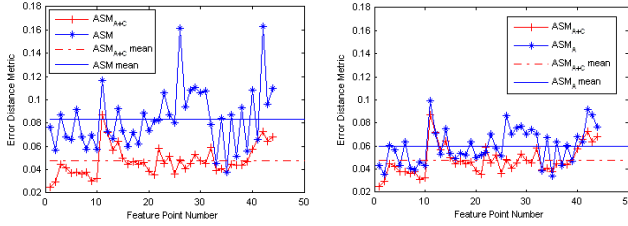
Fig. 2. Average positional error of the 44 feature points comparing with ASM and ASM_A

3.2   Tests on BioID Face Database

The BioID images consist of 1521 images of frontal faces with a set of 20 manually labeled feature points. The images are taken in uncontrolled conditions using a web camera within an office environment. We test our method under the same Testing Criteria described by D. Cristinacce and T. Cootes [9]. The comparison of EASM and SOS is shown in Table 1, where m_{e17} is the mean error distance of the 17 points [9]. Note that the proportion of successful searches of our EASM should be relatively higher if our face detection failures are the same 74 as in SOS, considering that most of the detection failures are low quality images much harder to analyze.

Table 1. Comparison of EASM and SOS [9] using the same 17 feature points

Methods	Global face detection failures	Proportion of successful searches	
		$m_{e17}<0.10$	$m_{e17}<0.15$
EASM	51	84%	97%
SOS[9]	74	85%	96%

Because of the eyes localization is much important in face recognition and some other applications we pay much concentration in eyes searching and localization. Our method achieves high accuracy in eyes localization with 95% of the tested images on BioID data set whose errors are less than 0.1 and in this condition the conventional ASM achieves only 22% (shown in right side of Figure 3).
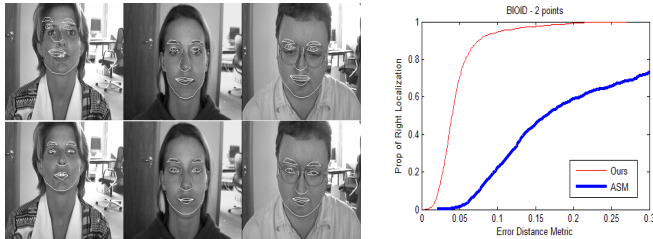
Fig. 3. Some examples of searching results of ASM (up) and EASM (down), and the right side shows average positional error of the eyes, comparing EASM and ASM, on BioID

4   Conclusions and Future Work

In this paper, we present our Enhanced Active Shape Models (EASMs) by using the more accurate AdaBoost-based landmark detector instead of the relatively simple

profile matching, and exploiting the confidences of the landmark localization to constrain the shape modeling and reconstruction in order to avoid the negative effects of those unreliably located landmarks. We empirically show that these two improvements can impressively improve the accuracy of the ASMs. Especially, the technique that only the reliable points are used to reconstruct the shape has many advantages: firstly we have decreased the effect of the improperly searched points to the final shape; secondly we can use only part of the landmark points to set the initial shape instead of specific feature points (e.g. eyes), if these feature points are sheltered, it does not necessarily change the final result; thirdly we can easily change the searching strategy and keep the shape model unchanged just by setting the unused feature points invalid.

Currently, the landmarks are directly divided two distinct groups according to their localization confidence, and only the so-called "reliable" landmarks are involved in the shape modeling. However, considering that the confidences are real continuous values, in the future, we will try confidences-weighted shape modeling methods to make better use of the information. How to further expand the proposed methods to AAMs will also be considered in the future.

Acknowledgments

References

1. M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. International Journal of Computer Vision, pp. 321–331, 1988.
2. A.L. Yuille. Deformable templates for face detection, J. Cogn. neurosci. 3, pp.59–70, 1991.
3. L. Wiskott, J. M. Fellous, N. Kruger, C. Malsburg, Face Recogniton by Elastic Bunch Graph Matching, IEEE TPAMI, Vol.19, No. 7, pp. 775–779, 1997.
4. V. Kruger and G. Sommer. Gabor wavelet networks for object representation. Technical Report 2002, Institute of Computer Science, University of Kiel, 2000.
5. T.F. Cootes, C. J. Taylor, D.H. Cooper, and J. Graham. Active shape models - their training and application. CVIU, 61(1), pp. 38–59, January 1995.
6. T.F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In 5th ECCV, volume 2, pp.484–498. Springer, Berlin, 1998.
7. P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. CVPR 2001, volume 1, pp.511–518, Kauai, Hawaii, 2001.
8. D. Cristinacce and T. F. Cootes. Facial feature detection using AdaBoost with shape constraints. In 14th BMVC, pp. 231–240, 2003.
9. D. Cristinacce and T. F. Cootes. A Comparison of Shape Constrained Facial Feature Detectors. Proceedings of the Sixth IEEE FGR, Seoul, Korea, 2004.
10. Haifeng Chen. Principal Component Analysis with Missing Data and Outliers. Technical Report 2002, Electrical and Computer Engineering Department Rutgers University, Piscataway, NJ, 08854.
11. M.E. Tipping and C. M. Bishop. Probabilistic principal component analysis. Journal of the Royal Statistical Society, Series B, 61, Part 3, pp. 611–622, 1999.

Face Authentication
Using One-Class Support Vector Machines

Manuele Bicego[1,★], Enrico Grosso[1], and Massimo Tistarelli[2]

[1] DEIR, University of Sassari, via Torre Tonda 34, 07100 Sassari, Italy
Phone +39 079 2017321
bicego@uniss.it
[2] DAP, University of Sassari, piazza Duomo 6, 07041 Alghero (SS), Italy

Abstract. This paper proposes a new method for personal identity verification based the analysis of face images applying One Class Support Vector Machines. This is a recently introduced kernel method to build a unary classifier to be trained by using only positive examples, avoiding the sensible choice of the impostor set typical of standard binary Support Vector Machines. The features of this classifier and the application to face-based identity verification are described and an implementation presented. Several experiments have been performed on both standard and proprietary databases. The tests performed, also in comparison with a standard classifier built on Support Vector Machines, clearly show the potential of the proposed approach.

1 Introduction

Face analysis is undoubtedly an interesting research area for challenging methodological issues and potential practical applications. In the past, several methods have been proposed for face authentication (personal identity verification from one face sample) and recognition (subject identification from a face database), ranging from geometrical approaches to statistical or local analysis [1]: almost all techniques explored by the Pattern Recognition and Machine Learning research have been tested in the face analysis application context. Among others, Support Vector Machines [2] have had a great success in this field, for several reasons: fast training [3], accurate classification and, at the same time, high generalization capability (i.e. the ability of learning the trend and the regularity of observed data).

Support Vector Machines have been employed for face analysis, recognition [4, 5] and authentication [6, 7], showing promising results. Support Vector Machines are intrinsically a binary classifier: the SVM determines an hyperplane that optimally separates two sets, i.e. positive and negative examples. This is a limitation for identity verification because only positive examples should be used for training. The most widely adopted solution is to train a binary SVM with instances of the target face as positive examples and gathering a set of different faces from another database, as negative examples (representing "The rest of the

---

★ Contact author

world"). Consequently, the set of negative samples must be general enough to faithfully represent any individual but the target subject. Since the choice of a proper or complete set of negative examples is crucial, an improper or incomplete set could heavily bias the authentication result. In the experimental section of the paper, it will be shown that different choices of the negative set could lead to different authentication rates. Therefore, a desirable solution is represented by a modelling tool, which maintains the good classification capabilities of the SVM, but also to be trained by positive examples only. Such tool exists, and is represented by the recently introduced One Class Support Vector Machines [9–11]: this method determines the optimal hypersphere enclosing the training data in the feature space. An interesting application to face detection has been proposed by Jin and others [12]

This paper investigates the use of one class SVM for identity verification from face images. Image feature vectors are used to train a One-Class SVM. In the authentication phase, the face vector is fed to the trained classifier and the distance from the center of the hypersphere is computed. This measure represents the inverse of the matching score. The claimed identity is verified by comparing the score with a pre-determined threshold. In the experimental section, OC-SVM and standard binary SVM are compared using the ORL database. Further experiments on the OC-SVM based system have been also reported.

2   One Class Support Vector Machines

One Class SVM (OC-SVM) represents a recently introduced kernel method [9–11] able to characterize the support of a high dimensional distribution. The standard SVM is a binary classifier, able to distinguish between two classes by using a separating hyperplane. Due to its binary nature, this technique is not adequate to model one class problems, i.e problems in which only positive examples are present. On the other side, the OC-SVM is a unary classifier, able to derive a support vector description of a data set formed by positive examples only. In order to do that, OC-SVM computes the smallest hypersphere in the feature space enclosing the data. Also in this case, all the advantages of the kernel trick could be mantained.

More formally, let $D = \{x_i\}$ be the data set of ℓ examples to be modelled. In the linear case (no kernel is employed), the goal is to find the smallest sphere that contains the data, i.e:

$$||x_i - a||^2 \leq R^2 \qquad i = 1 \cdots \ell \tag{1}$$

where R is the radius, a is the center and $|| \cdot ||$ is the Euclidean norm.

These constraints can be relaxed by introducing the slack variables $\xi_i > 0$:

$$||x_i - a||^2 \leq R^2 + \xi_i \qquad i = 1 \cdots \ell \tag{2}$$

The problem of finding the smallest hyperplane is solved by introducing the Lagrangian $\mathcal{L}$:

$$\mathcal{L}(R, a, \xi_1, ..., \xi_\ell) = R^2 - \sum_i (R^2 + \xi_i - ||x_i - a||^2)\alpha_i - \sum_i \xi_i \beta_i + C \sum_i \xi_i \tag{3}$$

where $\alpha_i, \beta_i \geq 0 \; \forall i$ are the lagrange multipliers associated to the constrain (2), C is a trade-off parameter, and $\sum_i \xi_i$ is a penalty term accounting for the presence of the outliers. Setting to zero the partial derivative of $\mathcal{L}$ with respect to R, a, ξ_i and solving, we obtain

$$\sum_i \alpha_i = 1 \qquad a = \sum_i \alpha_i x_i \qquad \alpha_i = C - \beta_i \qquad \forall i \qquad (4)$$

Using these results, the costrained minimization of the Lagrangian in (3) can be rewritten as the maximization of the Wolfe dual form $\mathcal{L}$:

$$\mathcal{L} = \sum_i \alpha_i x_i \cdot x_i - \sum_i \sum_j \alpha_i \alpha_j x_i x_j \qquad (5)$$

$$0 \leq \alpha_i \leq C, \qquad \sum_i \alpha_i = 1 \qquad \forall i \qquad (6)$$

Finally, the Karush-Kuhn-Tucker conditions give

$$\xi_i \beta_i = 0 \qquad (R^2 + \xi_i - ||x_i - a||^2)\alpha_i = 0 \qquad (7)$$

The simple linear case could be quite easily extended to the non linear case. By looking to the dual form in (5) one could observe that the inner product between input points could be replaced by a Kernel function $K(x, x)$, exactly as in the standard SVM theory. The idea is to find the closing hypersphere in the space induced by the chosen kernel, resulting in a non linear boundary in the original space.

The only problem of this non linear case is that, if the mapping induced by the kernel is unknown (as in the Gaussian case), the center of the hypersphere could not be explicitly computed as a linear combination of the image of the training points. Nevertheless, the distance between a point x and the center of the hypersphere A can be computed in the feature point as:

$$d^2(x) = K(x, x) - 2 \sum_i \alpha_i K(x_i, x) + \sum_i \sum_j \alpha_i \alpha_j K(x_i, x_j) \qquad (8)$$

3 The Proposed Approach

Every authentication system is composed of two distinct modules: off-line enrollment and on-line verification. Both modules rely on the extraction of one feature vector from each image, as from the following steps:

1. The Viola face detector [13] is applied to detect face patterns on the image. It is based on a cascade of weak classifiers combined with the AdaBoost scheme, working in a multiscale fashion. If the face detector is unable to properly localize the face, the image is discarded from the subsequent analysis.
2. The gray level face image is normalized against global illumination changes.

3. The normalized face image is subsampled to obtain a fixed size image ($60 \times 60 pixels$). This procedure allows an approximate scale invariance.
4. The image is vectorized to obtain a feature vector. The direct use of raw gray level information as features for the Support Vector Machines is not new in the literature [14]).

One OC-SVM is trained for each subject, representing the subject template. A video stream is used for gathering all the training images and 50 images are automatically chosen for training. The choice may be performed randomly or driven by a knowledge-based criterion to choose the *most representative set of images*.

Also for the authentication phase several face images are acquired from a video sequence. Subsequently, all images are processed to obtain the feature vectors, which are fed to the OC-SVM corresponding to the subject of the claimed identity. Each image submitted to the OC-SVM produces a matching score, which is the distance from the center of the trained hypersphere. If this distance is below a given threshold the claimed subject's identity is confirmed. The choice of the authentication threshold is obviously critical. Two possibilities are considered: either to choose one threshold for all subjects or to set a different threshold for each subject. Both these options have been tested in the experimental section.

4   Experimental Results

This section reports a preliminary experimental evaluation of the proposed approach. In particular, in the first test the OC-SVM tool has been compared with standard SVM methods, using the ORL database. Subsequently, the system has been tested on two proprietary databases with varying facial expression, illumination and scale.

4.1   Binary Versus One-Class SVM

In order to perform a direct comparison between OC-SVM and standard binary SVM, the same data set, protocol and features are used to perform both tests. The ORL database, containing 40 subjects, with 10 images for each subject, has been used. For each subject only 5 images have been used for training. Both complete client and impostor tests were performed. In the first case the authenticator is tested using images of the same subject (but not the same used for training) to estimate the false rejection rate (FRR). In the impostor test, other subjects images are fed into the system, to measure the false acceptance rate (FAR). The equal error rate (EER), i.e. the rate for which FAR=FRR, is picked as an estimate of the discrimination capability of the classifier.

In standard applications of binary SVM the authentication phase is performed by computing the distance of the test image from the separating hyperplane. The sign of the distance determines the class to which the subject belongs. In the current implementation the distance from the hyperplane is used

as matching measure, with the option of setting a threshold (which is 0 in standard SVM).

In binary SVM there is the need of defining the impostor set, i.e. the negative examples. In order to highlight the fact that the choice of the impostor set is crucial, four different training sessions were performed, varying the impostor set (for all 10 randomly chosen subjects, with 2 images each): from the Bern database[1], the Stirling database[2], the Yale database[3], and from a mixture of the Stirling and the Yale database.

Please note that all SVM parameters remained unchanged but only the impostor set was changed. In particular, for both SVM and OC-SVM, the Radial Basis Function kernel was used with σ and C parameters fixed. From the results displayed in Table 1 it could be noted that the choice of the impostor set, for binary SVM, is crucial. In fact, the classification results change when varying the impostor data set.

Table 1. Comparative EERs

Method	Impostor Set	EER
Binary SVM	Bern	6.00%
Binary SVM	Stirling	5.50%
Binary SVM	Yale	6.00%
Binary SVM	Mixed	6.19%
One class SVM	-	5.50%

The OC-SVM, instead, does not require any impostor set, and the performances are stable. In particular, the OC-SVM Equal Error Rate is equal to the best SVM case: OC-SVM maintains all the good classification capabilities of the SVM, still removing the awkward problem of needing an appropriate impostor set.

4.2 Proprietary Databases

The proposed system was thoroughly tested using two proprietary databases. The first database includes 25 subjects with an image sequence of 95 to 195 color images for each subject, with several changes in facial expression and scale (see fig. 1). The sampled images are 640x480 pixels. For the face classification the images have been reduced to gray level with 8 bits per pixel.

50 images have been chosen for training. The RBF kernel was used ($\sigma = 1800$), and the regularization constant C was set to 0.3. These parameters have been selected after a preliminary evaluation on the ORL database, varying the kernel and parameters' configurations.

---

[1] Downloadable from ftp://iamftp.unibe.ch/pub/Images/FaceImages
[2] Downloadable from http://pics.psych.stir.ac.uk
[3] Downloadable from http://pics.psych.stir.ac.uk

Fig. 1. Some variability in the first proprietary database

In table 2 the EER obtained for two different configurations is shown, i.e. by setting the same threshold for all subjects, or setting different thresholds for each subject. In the latter case, the reported EER is the average between all EERs resulting from the ROC curves. In fig. 2(a) the ROC curve for the single threshold selection is presented.

Table 2. EER from the proposed system based on One Class SVM, applied to the first proprietary database

Method	number of client tests	number of impostor tests	EER
One threshold for all subj	2275	122520	3.38%
One threshold for each subj	2275	122520	0.56%
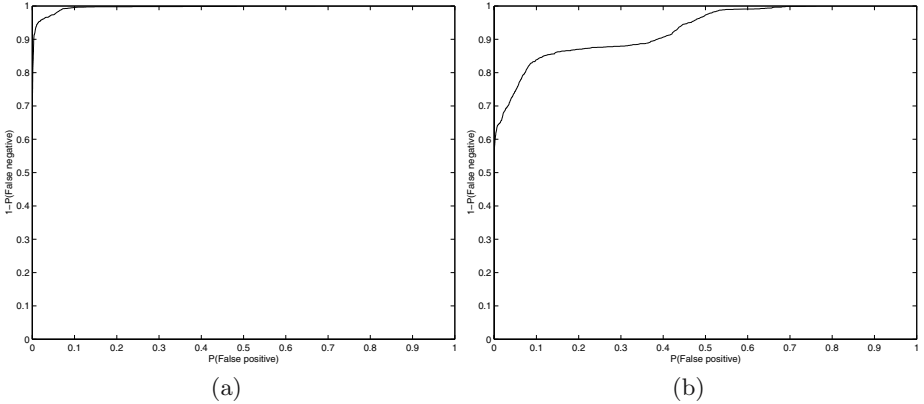
Fig. 2. ROC curves obtained from the identity verification test using: (a) the first database and (b) the second database

In order to better understand the potential of the proposed method a further experiment has been performed. The face image database has been augmented by adding more images to the existing set. For five out of the total 25 subjects, an image set acquired in a different session (at least two weeks apart) has been added. The training has been performed using only images from one of the two acquisition sessions. Some example images are presented in Fig. 3. The corresponding EERs are shown in table 3 for both single and multiple threshold selection. In fig. 2(b) the corresponding ROC curve for the single threshold selection is presented.

Fig. 3. Some example images from the two different sessions of the second proprietary database. Pictures from the same acquisition session are presented on the top and bottom row respectively

Table 3. EER from the proposed system based on One Class SVM, applied to the second proprietary database

Method	number of client tests	number of impostor tests	EER
One threshold for all subj	1760	10684	14.31%
One threshold for each	1760	10684	6.53%

5   Conclusions

In this paper a new face authentication technique has been proposed, based on One Class Support Vector Machines. This tool maintains all the attractive features of Kernel tools like standard Support Vector Machines, such as high generalization capabilities, but without the need for an impostor set. In the experimental evaluations it has been shown that, in the standard binary SVM case, the choice of the impostor set is crucial and may lead to different authentication performances.

A further improvement of the system is to combine, using a majority vote scheme, the results obtained by processing a whole video sequence. In other words, each image of a video sequence is processed, resulting in a yes/no answer. If the majority of the images produce a positive answer then the subject is authenticated, otherwise the authentication is rejected.

References

1. Zhao, W., Chellappa, R., Phillips, P., Rosenfeld, A.: Face recognition: A literature survey. ACM Computing Surveys **35** (2003) 399–458
2. Vapnik, V.: The Nature of Statistical Learning Theory. Springer-Verlag (1995)
3. Platt, J.: Fast training of support vector machines using sequential minimal optimization. In: Advances in Kernel Methods - Support Vector Learning. MIT Press (1999) 185–208
4. Guo, G., Li, S.Z., Kapluk, C.: Face recognition by support vector machines. Image and Vision Computing **19** (2001) 631–638

5. Heisele, B., Ho, P., Poggio, T.: Face recognition with support vector machines: Global versus component-based approach. In: Proc. Of IEEE Int. Conf. on Computer Vision. Volume 2. (2001) 688–694
6. Smeraldi, F., Capdevielle, N., Bigun, J.: Face authentication by retinotopic sampling of the gabor decomposition and support vector machines. In: Proc. of Int. Conf. on Audio and Video Based Biometric Person Authentication. (1999) 125–129
7. Jonsson, K., Kittler, J., Li, Y., Matas, J.: Support vector machines for face authentication. Image Vision Computing **20** (2002) 269–275
8. Tefas, A., Kotropoulos, C., Pitas, I.: Using support vector machines for face authentication based on elastic graph matching. In: Proc. of IEEE Int. Conf. on Image Processing. Volume 1. (2000) 29–32
9. Ben-Hur, A., Horn, D., Siegelmann, H., , Vapnik, V.: Support vector clustering. Journal of Machine Learning Research **2** (2001) 125–137
10. Schölkopf, B., Williamson, R., Smola, A., Shawe-Taylor, J., Platt, J.: Support vector method for novelty detection. In: Advances in Neural Information Processing Systems. Volume 12. (1999) 526–532
11. Tax, D., Duin, R.: Support vector domain description. Pattern Recognition Letters **20** (1999) 1191–1199
12. Jin, H., Liu, Q., Lu, H.: Face Detection Using One-Class-Based Support Vectors. In: Proc. of Sixth IEEE Int. Conf. on Automatic Face and Gesture Recognition (2004) 457–463
13. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: IEEE Proc. Int. Conf. on Computer Vision and Pattern Recognition. Volume 1. (2001) 511–518
14. Pontil, M., Verri, A.: Support vector machines for 3d object recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence **20** (1998) 637–646

A Novel Illumination Normalization Method for Face Recognition

Yucong Guo, Xingming Zhang, Huangyuan Zhan, and Jing Song

Department of Computer Engineering and Science,
South China University of Technology, GuangZhou, China, 510640
{yucong_guo,netsoftzhy}@hotmail.com

Abstract. The problem of illumination makes the face recognition still an un-solved problem. A new normalization method was presented which is illumination invariant in face recognition. The histogram equalization was applied to improve the performances of the AT (Affine Transform) algorithm. A novel illumination normalization method was introduced. Using the information of the distribution of the histogram, the method makes the combination of the AT algorithm and the ICR (Illumination Compensation based on Multiple Regression Model) algorithm smoothly. Experiments reveals that our proposed algorithm is illumination invariant and achieves better preprocess result while the recognition rate has been evidently improved.

Keywords: Illumination normalization; face recognition; illumination invariant; degeneration model

1 Introduction

Face recognition has being one of the central issues in the field of image process and pattern recognition in recent years. Most of the researches concerning face recognition have point out that the most challenging problems in face recognition are pose, illumination and expression. Especially, the problem of illumination is much more difficult to solve. It has been proved that, the difference for two images, which belong to one individual in different illumination conditions, has overrun the difference for two images belong to two individuals with the same illumination conditions [1]. Therefore, how to solve the illumination problem for face images and achieve the illumination invariant preprocess effect, is one of the most significant issues in face recognition.

To deal with the illumination problem, usually, the image enhancement and image recover techniques, such as histogram equalization, histogram normalization are applied [2]-[4]. However, experiments show that all the above methods cannot achieve good preprocessing effect. In recent years, new solutions to achieve illumination invariant preprocessing effect for face images have been put forward, such as illumination cone, quotient images etc [5]-[8]. While these solutions do work with the illumination problem, they either has too much time cost, or cannot be widely applied for their restrictions.

In this paper, we present a new face normalization algorithm for face image that is convenient to apply. The histogram equalization was applied to improve the performances of the AT (Affine Transform) algorithm [9]. Using the information of the distribution of the histogram, a method was advanced to combine the AT algorithm and ICR (Illumination Compensation based on Multiple Repression Model) algorithm [10]

smoothly. Experiments demonstrate that our proposed method is illumination invariant and achieves better preprocess effect in variable illumination environments and complicated backgrounds while the face recognition rate has been evidently improved.

2   Methods

2.1   Illumination Corrections Using Affine Transform

As for the mathematics model presents the degenerations of face image caused by the illumination problem, many investigators have given their own statements [1], [5], [8]. Among them, one degeneration model that is mostly accepted is established based on a mass of experiences.

$$P'(x, y) = A(x, y)P(x, y) + B(x, y) \tag{1}$$

In Eq. (1), P (x, y) defines the original face image, A (x, y) and B (x, y) define the genes that cased the degenerations of the image. P′ (x, y) is the modified image as a result of local lighting. Based on the above experiential mode, the key of the illumination normalization is to get a precise estimation P′ (x, y)for P (x, y). We should estimate the A (x, y) and B (x, y) firstly. When the estimations of $\hat{A}(x, y)$ and $\hat{B}(x, y)$ are gained, we could recover the image by estimating the P′ (x, y) using Eq. (1). As a whole, we have applied the AT algorithm [9].

$$\overline{P}(x, y) = \frac{k}{\hat{A}(x, y)}(P'(x, y) - \hat{B}(x, y)) \tag{2}$$

In Eq. (2), k is a stable coefficient.

Based on the above degenerate model, if we can estimate A (x, y) and B (x, y) accurately, an illumination invariant algorithm would be available. However, as we can learn from above, the estimation of the A (x, y) and B (x, y) can only be carried out on the degenerate images. To achieve a better estimation, we should determine an appropriate method to estimate the A (x, y) and B (x, y).

Experiences show that the B (x, y) usually presents the background of an image. In our algorithm, we use a large kernel low-pass-filter to estimate B (x, y).

$$\hat{B}(x, y) = LPF(P'(x, y)) \tag{3}$$

Once the B (x, y) has been estimated, we estimate the A (x, y). The Eq. (1) is equal to

$$P'(x, y) - B(x, y) = A(x, y)P(x, y) \tag{4}$$

It is agreed that when in complicated environments, contrasting to the alteration of P (x, y), the alteration of A (x, y) is oppositely slow. A log transform can be applied to both size of Eq. (4), then a low-pass-filter is used to filter the P (x, y) which presents the high frequency, finally we get the estimation of A (x, y) as below.

$$\hat{A}(x, y) = \exp(LPF(\ln(P'(x, y) - B(x, y)))) \tag{5}$$

Eq. (5) gives the whole process of how to estimate A (x, y). It should be noted that the LPF we used in Eq. (5) is different from the one we used in Eq. (3). Here we applied a low-pass-filter with the kernel size 7×7.

The operation of P′ (x, y) subtracts B (x, y) is used frequently, this would make the high frequency part of an image bulgy. To reduce its infection, some techniques should be done to correct it. Through experiments, we use the following correction method to get the accurate estimation of P (x, y).

$$\overline{P}(x,y) = \frac{k}{\hat{A}(x,y)\sqrt[4]{\dot{P}(x,y)}} \ln(|\dot{P}(x,y) - \hat{B}(x,y)|)$$ (6)

When in applications, some improvements are done on the algorithm. Firstly, when the result of $P'(x, y) - \hat{B}(x, y)$ is equal to 0, we shift all the values of $P'(x, y) - \hat{B}(x, y)$ in the coordinate, this would lead to some departure, we could revise the result by the help of k. Meanwhile, the distribution of $\overline{P}(x, y)$ from Eq. (6) might be overflow the thresholds T_{lo} or T_{hi} of the pixel values that the system can express, For example, when in gray image, T_{lo} and T_{hi} are equal to 0 and 255. Image normalization should be done in the last procedure of image preprocess.

$$\overline{P}(x, y) = P_{hi} \qquad when \ \overline{P}(x, y) >= T_{hi}$$
$$\overline{P}(x, y) = P_{hi} \qquad when \ \overline{P}(x, y) >= T_{hi}$$
$$\overline{P}(x,y) = \frac{P_{hi} - P_{lo}}{T_{hi} - T_{lo}}(\hat{P}(x,y) - T_{lo}) + P_{lo} \quad when \ T_{lo} < \overline{P}(x,y) < T_{hi}$$

Through a mass of experiments, we become conscious of the fact that if histogram equalization is applied before the procedure of AT algorithm, remarkable improvements can be achieved. This technique is also used in our proposal algorithm below.

2.2 Illumination Correction Using ICR Algorithm

Recent years, Jaepil Ko has put forward a new illumination correction algorithm that is named ICR and applied it in face recognition [10]. The implement of the algorithm is rather simple. As to a given face image, we simulate the illumination identity density plane through the multiple regression models. Once the plane is available, reverse transform is applied to get a new illumination density plane that is used for illumination compensation. We add this plane to the original image to compensate the illumination and get the result image.

When applying for the ICR algorithm, for face image P′ which is n×m, we extend it row by row to form a vector $X = [x_0, x_1 ...... x_{q-1}]^T$ (q=n×m, which stands for the total pixel number). Using the multiple regression model, we get the best fit illumination density plane of P′, which we used $Y = [y_0, y_1 ...... y_{q-1}]$ to present. When illumination problem exists, the distribution of this illumination density plane is similar to the real distribution of the illumination. To reduce the influence of illumination, a new plane should be build to compensate for the vile illumination, we define:

$$y_c = (\max(y_i) - \min(y_j))/2 \qquad 0 <= i, j < q$$ (7)

We can build the new illumination density plane using Eq. (8).

$$Y' = [y_c - y_{0,} y_c - y_1 \dots\dots y_c - y_{q-1}]^T \qquad (8)$$

Once the compensated plane Y' is available, the illumination corrected image can be achieved:

$$P' = P + Y' \qquad (9)$$

The implement of ICR algorithm is rather simple. However, since it only considers the compensation using illumination density plane in one direction, it is much more restricted and not robust. For these reasons, usually it does not single out to apply for illumination correction for face images.

2.3   Our Proposal Illumination Invariant Method

2.3.1   Discussion

Though the AT or ICR algorithm   does have effect on illumination normalization and good performances can be observed when they are applied to face normalization, see [9]-[10] for detail. Still the illumination problem remains. As we discuss above, the illumination problem in face recognition is very complex. In recent years, more reasonable degeneration models are built, like illumination cone. When compared to these more reasonable model, the linear degeneration model used in the AT algorithm is rather simple. Especially when the illumination becomes much worse, the AT algorithm does not work well on illumination correction. As for ICR algorithm, as we conclude above, it does work well when dealt with the side-lamp-house. However, usually it does not single out to apply for illumination correction for face images.

Compared with most face illumination normalization methods, the AT and ICR method do have their advantages. They are not time consuming and rather simple and convenient to apply. In the following part we will try to make some improvement to the algorithms and present our proposal illumination normalization method.

2.3.2   Our Method

Before we go even further, we first introduce a new algorithm named ATICR, which combines both the virtues of the above two algorithms. Here we continue to present our new illumination normalization algorithm. The frame procedure of our proposal illumination normalization algorithm is showed in Figure.1.

As Figure.1 shows, given an face image P, firstly we apply the improved AT algorithm and get the result P_a, then we apply the ICR algorithm on the original image P and get a temporary image which has been compensated for illumination. To reduce the influences of the combination, we should correct this temporary image using the improved AT algorithm. After this step, we get the result image P_i.

Through the above two procedures, we have got the two result P_a and P_i, the next step is to find an appropriate method to combine these two result image together.

Since the content of these two result images are the same. We could think it that the above two procedures are just transform the same pixel to different regions in the coordinate. However, as the whole, the two result images still have the same distribution and their histograms should be similar. This suggests that we can use the statistical method such as expectation to make the combination rational.
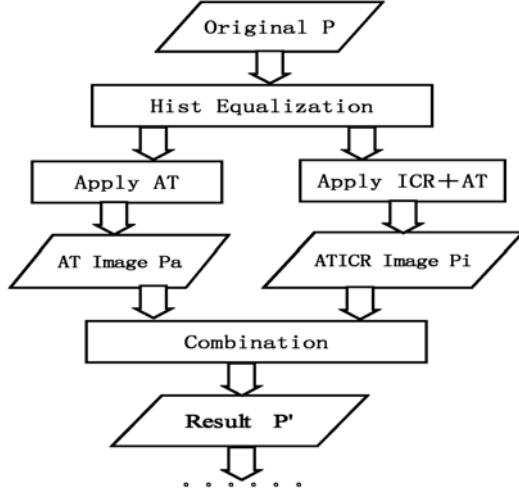
Fig. 1. The frame procedure of our proposal illumination invariant method

$$\overline{P}_a = \sum_{i=0}^{w} \sum_{j=0}^{h} P_a(i, j) \tag{10}$$

$$\overline{P}_i = \sum_{i=0}^{w} \sum_{j=0}^{h} P_i(i, j) \tag{11}$$

We define r as follow.

$$r = \frac{\overline{P}_a}{\overline{P}_i} \tag{12}$$

Then, we could determine the combined result image P' using Eq.(11).

$$P'(x, y) = \frac{P_a(x, y) + r \times P_i(x, y)}{1 + r} \tag{13}$$

Experiments shows that it can eliminate the influences that come from the different bar centers and makes the combination of the two algorithms more rational.

3   Experiments

We have tested our proposal algorithm in two illumination databases. One is our owned IFace illumination database and the other is Yale B illumination database.

The IFace illumination database is gathered in our laboratory. The intention to set up this face database is just about to evaluate the performances of different illumination invariant preprocess algorithms. All the images are captured from the camera. There are totally 106 individuals. Each has dozens of face images with different illuminations and poses. According to the illumination conditions, we group them into three sets – A, B and C. Set A represents the natural illumination. Set B represents the sidelight illumination, and the background is gloomy. The illumination problem becomes much

worse in set C and this makes the face images in set C degenerative heavily. Different illumination normalization algorithms are applied on the database and we can see their effects in Figure.2.
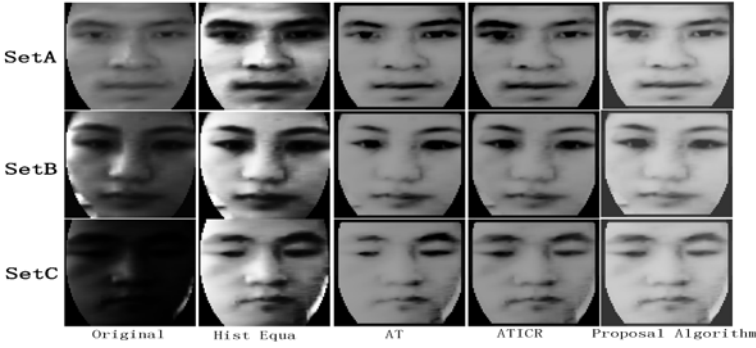
Fig. 2. Comparison of processed images after different illumination normalization methods for images in IFace database

As Figure.2 presents, our proposal illumination invariant algorithm has good performances in all three sets. Meanwhile, the result images in set C reveal that while the AT algorithm does recover the detail of the right side of the degenerative image, it has weaken the feature of the left side of the face image. However, inverse effect can be seen in the result image of ATICR. While the ATICR algorithm recovers the left side well, degeneration could be observed in the right side of the face image. This is just the motivation to combine the above two result images using our algorithm.

Figure.3 reveals the effect of our proposal algorithm in Yale B illumination database.

Fig. 3. The processed images of an individual from Yale B database using our proposal illumination normalization method

Furthermore, we have applied our method to face recognition system to evaluate the influence to the recognition rate. For the IFace illumination database, five images per person are selected out as the gallery set. In each illumination conditions, 50 images are picked out to form the probe set. The Fisher algorithm is chosen as the feature extraction and recognition method. For the famous illumination database Yale B, totally there are 10 individuals, each has 64 images in different illumination conditions. We divide the database into five subsets according to the angle that the light source direction makes with the camera axis – Subset 1 (0-12°), Subset 2 (12-25°), Subset 3 (25-50°),

Subset 4 (50-70°), Subset 5 (remnants) [4]. Subset 1 is gallery set while the other subsets are probe set. The recognition evaluation result could be seen in Table.1. It shows that our algorithm has evidently improved the recognition rate in different illumination conditions, while the tradition illumination normalization algorithm such as histogram equalization and GIC (Gamma Illumination Correction) degenerate observably when the illumination varies.

Table 1. Comparison of the CRRs (correct recognition rate)

Recognition Rate (%)	Hist. Equa.	GIC	AT	ATICR	Proposal
IFace Set A	96.83	94.14	97.17	96.83	99.17
IFace Set B	56.33	62.00	84.50	81.17	88.67
IFace Set C	37.73	57.48	63.29	64.73	82.21
Yale B Subset 2	100.00	99.86	99.38	99.38	99.86
Yale B Subset 3	89.01	92.03	96.06	91.77	98.35
Yale B Subset 4	57.26	75.48	83.60	78.56	89.44
Yale B Subset 5	36.09	50.67	66.49	67.15	74.68
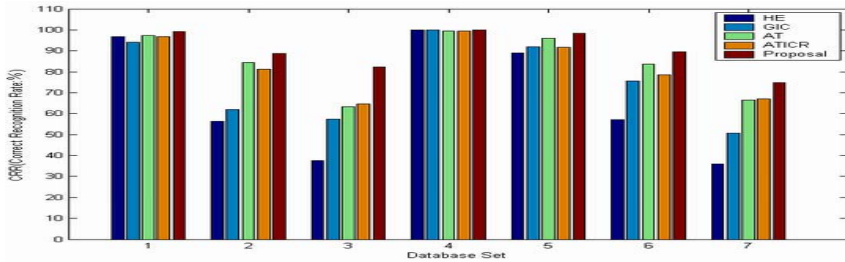
Fig. 4. The Recognition Result on Different Databases: 1.IFace Set A; 2.IFace Set B; 3. IFace Set C; 4.Yale B Subset 2; 5.Yale B Subset 3; 6.Yale B Subset 4; 7.Yale B Subset 5

Through Experiments, we also had found that our proposal algorithm has low time cost. For an image with the size 92×112, it takes about 100 millisecond to process with our novel method. This makes our proposal algorithm less restricted to apply and would not be the bottleneck of the whole face recognition system.

4   Conclusions

A new illumination normalization algorithm was presented to correct the illumination for face images. The histogram equalization was applied to improve the performances of the AT algorithm and ICR algorithm. Using the distribution information of the histogram, a method was introduced to combine several preprocess algorithms together smoothly. Experiments demonstrate that our proposed algorithm is illumination in-variant and achieves better preprocess result while the recognition rate has been greatly improved.

However, through experiments we have noticed that the proposal algorithm has lightly weakened some local features of the face image. To avoid the loss of the image information and recover more detail of the features, and to find a faster algorithm optimization method, will be the focus of future research.

References

1. P.N.Belhumeur and D.J.Kriegman. What is the set of images of an object under all possible illumination conditions?. IJCV,1998,28(3):245-260.
2. Haitao Wang, Stan Z.Li and Yangsheng Wang. Illumination Modeling and Normalization for Face Recognition. Proceeding of AMFG, 2003.
3. Rafael C. Gonzalez. Digital Image Processing. Publishing house of electronics industry. 2003.
4. Shiguang Shan, Wen Gao and Bo Cao. Illumination Normalization for Robust Face Recognition Against Varying Lighting Conditions. Proceeding of AMFG, 2003.
5. Shaohua Zhou and Rama Chellappa. Rank Constrained Recognition under Unknown Illuminations. Proceeding of AMFG, 2003.
6. Jiali Zhao, Yong Su,Dejun Wang. Illumination ratio image: synthesizing and recognition with varying illuminations. Pattern Recognition Letter 2003(24):2703-2710.
7. P.Jonathon Philips and Yehuda Vardi. Efficient illumination normalization of facial images. Pattern Recognition Letter 1996(17):921-927.
8. Haitao Wang, Hong Wei and Yangsheng Wang. Face Representation under Different Illumination Conditions. Proceeding of ICME, 2003.
9. Juhua Zhu, Bede Liu and Stuart C.Schwartz. General illumination correction and its application to face normalization. Proceeding of AMFG, 2003.
10. Jaepil Ko, Eunju Kim and Hyeran Byun. A simple illumination algorithm for face recognition. Proceeding of PRICAI, 2002.

Using Score Normalization to Solve the Score Variation Problem in Face Authentication

Fei Yang[1,3], Shiguang Shan[1], Bingpeng Ma[1,3], Xilin Chen[1,2], and Wen Gao[1,2,3]

[1] Institute of Computing Technology, CAS, Beijing, 100080, China
{fyang,sgshan,bpma,xlchen,wgao}@jdl.ac.cn
[2] Computer College, Harbin Institute of Technology, Harbin, 150001, China
[3] Graduate School of Chinese Academy of Sciences, Beijing, 100039, China

Abstract. This paper investigates the score normalization technique for enhancing the performance of face authentication. We firstly discuss the thresholding approach for face authentication and put forward the "score variation" problem. Then, two possible solutions, Subject Specific Threshold (SST) and Score Normalization (SN), are discussed. But SST is obviously impractical to many face authentication applications in which only a single example face image is available for each subject. Fortunately, we have theoretically shown that, in such cases, score normalization technique may approximately approach the SST by using a uniform threshold. Experiments on both the FERET and CAS-PEAL face database have shown the effectiveness of SN for different face authentication methods including Correlation, Eigenface, and Fisherface.

1 Introduction

The development of person identity verification based on biometric information provides wide potential applications in security, law enforcement, and commerce. A number of biometric traits have been studied for identity verification in the recent years, such as face, voice, signature, fingerprint, iris, retina, palm print, gait, etc [1]. Among them, face recognition has some unique advantages. For instance, facial images are easy to capture without any invasion by using various digital cameras or scanners, while some other traits like fingerprint and iris require special equipments, some of which even may have potential invasion. In addition, the verification results using facial images can be checked easily by common people while some other traits can be distinguished only by few experts (e.g. fingerprint or iris). Furthermore, face verification systems can be deployed in the scenarios either the person cooperates or not (even not known), which provides special advantage for security surveillance.

A face verification system usually consists of four main modules: data acquisition, feature extraction, feature matching, and decision making [2]. The data acquisition module acquires the biometric data from a user; the feature extraction module processes the acquired biometric data and extracts a feature set to represent it; the matching module compares the extracted feature set with the stored templates using some matching algorithm in order to calculate matching scores; finally, the decision module compares the matching scores with a threshold. If the score is equal or larger than the threshold, the claim is conformed, or else is denied.

For face verification, many factors cause the scores to fluctuate. Variations in expression, lighting, aging will make the similarity scores of genuine faces decrease,

even smaller than those of imposter faces under normal conditions. What is more, different persons enrolled in the system have different characteristics. Therefore, the distribution of genuine and imposter scores may vary from person to person. This problem, which we call "score variation" problem, will make the performance degrade obviously. So, using Uniform Threshold (UT) for all subjects in the decision-making module is evidently unsuitable. Two possible solutions, Subject Specific Threshold (SST) and Score Normalization (SN) [2], can be used to solve this problem.

The SST method assigns a specific threshold for each subject in the database, in order to adjust to different genuine and imposter score distribution of different subjects. Multiple example images are needed to model the genuine and imposter distribution for each person. But for many face authentication applications, only a single face image is available for each subject, so the genuine distribution is unavailable and SST can not be applied directly. In these cases, the SN methods are more commonly exploited. SN methods are mostly used in classifier fusion [2], signature verification [3], and speaker verification [4] domains. In face authentication filed, Sanderson etc. [5] use a Gaussian Mixture Model (GMM) based classifier for classification, in which the imposter distribution is modeled for normalization. Perronnin etc. [6] use relational approaches and develop two new score normalization methods R-Norm and G-Norm. Both their work need multiple images to train Gaussian model for each subject. The FRVT 2002 test [7] uses score normalization at similarity score level as a post-processing operation.

In this paper, we deal with only the similarity score, but not caring the types of the classifier. First, we put forward the constraints of the optimal Subject Specific Threshold (SST). Then, we show theoretically that Z-Norm, one of the SN methods, can approximately approach the SST method when genuine distribution is unavailable. We perform experiments to show how Z-Norm changes the distribution of scores, and its approximate performance to SST. Our experiments on FERET [8] and CAS-PEAL [9] face-database verify the effectiveness of SN for different face authentication methods including Correlation [10], Eigenface [11], and Fisherface [12].

2   Solutions to "Score Variation" Problem

In face verification, the system compares the candidate image with the image of the claimed identity to get a similarity score s. The decision module accepts or rejects the identity by comparing with a threshold θ, which is referred as the Uniform Threshold (UT) method.

$$\text{if } s \geq \theta \text{ accept, otherwise reject} \tag{1}$$

There are two probability distributions of pair-wise matching scores as illustrated in Fig. 1. The *genuine distribution* characterizes the similarity scores of two images from the same person, while the *imposter distribution* characterizes the similarity scores of two images from different persons.

Ideally, the genuine distribution will show small differences and the imposter distribution will show large differences. But for a real-world face verification system, variations in pose, lighting, expression, and aging may decrease the genuine scores significantly. In addition, since different persons enrolled in the system have different

characteristics, the distributions of genuine and imposter scores may also vary for different persons. This problem, which we call as "score variation" problem, will degrade the performance of UT-based systems obviously. Two possible solutions, Subject Specific Threshold (SST) and Score Normalization (SN), are previously proposed to solve this problem.
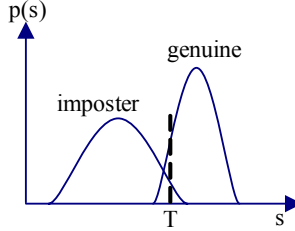
Fig. 1. Genuine and imposter distribution in a UT-based system

2.1   Subject Specific Threshold (SST)

One solution to the "score variation" problem is through the Subject Specific Threshold (SST) approach. In this approach, each subject G_i in the database has a specific threshold θ_i. For subject G_i, the decision making rule is redefined as:

$$\text{if } s \geq \theta_i \text{ accept, otherwise reject} \tag{2}$$

Then the False Alarm Rate (FAR) and Hit Rate (HR) for subject G_i are defined as:

$$FAR_i = \int_{\theta_i}^{+\infty} p(s \mid \bar{\lambda_i}) \, ds \quad\quad HR_i = \int_{\theta_i}^{+\infty} p(s \mid \lambda_i) \, ds \tag{3}$$

where $p(s \mid \bar{\lambda_i})$ and $p(s \mid \lambda_i)$ are respectively the imposter distribution and the genuine distribution for the G_i subject (refer to Fig. 1). So, the total FAR and HR for all registered subjects are defined as:

$$FAR = \frac{1}{|G|} \sum_{i=1}^{|G|} FAR_i = \frac{1}{|G|} \sum_{i=1}^{|G|} \int_{\theta_i}^{+\infty} p(s \mid \bar{\lambda_i}) \, ds \tag{4}$$

$$HR = \frac{1}{|G|} \sum_{i=1}^{|G|} HR_i = \frac{1}{|G|} \sum_{i=1}^{|G|} \int_{\theta_i}^{+\infty} p(s \mid \lambda_i) \, ds \tag{5}$$

For a face authentication system, if the FAR is fixed to t (e.g. according to the practical requirement), one should set the thresholds of each subject, $\theta_1, \theta_2, \cdots, \theta_{|G|}$, in order to get a highest HR. These thresholds can be solved by using the Lagrange Multipliers. First, we define the Lagrange function:

$$F = HR - K \cdot (FAR - t)$$
$$= \frac{1}{|G|} \sum_{i=1}^{|G|} \left[\int_{\theta_i}^{+\infty} p(s \mid \lambda_i) \, ds - K \cdot \left(\int_{\theta_i}^{+\infty} p(s \mid \bar{\lambda_i}) \, ds - t \right) \right] \tag{6}$$

To obtain a maximum of HR, the partial derivatives $\partial F / \partial \tilde{\theta}_i$ should be zero, that is,

$$\frac{\partial F}{\partial \theta_i} = -\frac{1}{|G|} (p(\theta_i \mid \lambda_i) - K \cdot p(\theta_i \mid \bar{\lambda_i})) = 0 \tag{7}$$

So, finally we have:

$$\frac{p(\tilde{\theta}_i \mid \lambda_i)}{p(\tilde{\theta}_i \mid \bar{\lambda}_i)} = K, \quad (i = 1, 2, \cdots \mid G \mid) \tag{8}$$

If we further assume the genuine scores of subject G_i satisfy a Gaussian distribution $N(\mu_i, \sigma_i)$ and imposter scores be a Gaussian distribution $N(\bar{\mu}_i, \bar{\sigma}_i)$, given a certain value K, the subject specific threshold can be solved by

$$\tilde{\theta}_i = f(K, \mu_i, \sigma_i, \bar{\mu}_i, \bar{\sigma}_i). \tag{9}$$

2.2   Score Normalization (SN)

Previous studies have shown that the performance of a number of biometric verification systems, especially those based on behavioral traits such as written signature or voice, can be improved by using score normalization method. In score normalization, the normalized score is a function of original similarity score, the input sample, the client specific information, and the information of imposters [6]. Researchers have proposed several SN methods, such as Z-Norm [3], T-Norm [6], G-Norm [8], etc. Among them, the Z-Norm method normalizes the similarity score by using the mean and standard deviation of the imposters:

$$s_{znorm} = \frac{s - u_{\bar{\lambda}_i}}{\sigma_{\bar{\lambda}_i}} \tag{10}$$

where $u_{\bar{\lambda}_i}$ is the mean of the scores from imposter images of subject G_i, $\sigma_{\bar{\lambda}_i}$ is the standard deviation.

Z-Norm actually aims to make the imposter distribution of different subjects the same standard normal distribution $N(0,1)$. Thus, one can easily configure the final verification system by setting a uniform threshold for all subjects.

3   Using Z-Norm to Approximately Approach SST

In the SST method, optimal target specific thresholds $\theta_1, \theta_2, \cdots, \theta_{|G|}$ are estimated by the candidate facial image, the genuine and imposter score distribution of identity G_i. However, in many face verification systems, only a single image per subject is enrolled in the system. Thus, the genuine distribution can not be obtained. In this case, we can neglect the genuine distribution in Eq.8, using only the imposter term:

$$p(\tilde{\theta}_i \mid \bar{\lambda}_i) = 1/K, \quad (i = 1, 2, \cdots \mid G \mid) \tag{11}$$

Suppose the imposter scores satisfy a Gaussian distribution $N(\bar{\mu}_i, \bar{\sigma}_i)$, then

$$\tilde{\theta}_i = \mu_{\bar{\lambda}_i} + \sigma_{\bar{\lambda}_i} \cdot \sqrt{2\ln(\sigma_{\bar{\lambda}_i} \cdot K) + \ln(2\pi)}, \quad (i = 1, 2, \cdots \mid G \mid) \tag{12}$$

We will see in the experimental part that $\sigma_{\bar{\lambda}_i}$ varies little for all subjects, so

$$\tilde{\theta}_i = \mu_{\bar{\lambda}_i} + \sigma_{\bar{\lambda}_i} \cdot K', \quad (i = 1, 2, \cdots \mid G \mid) \tag{13}$$

This form is equivalent to Z-Norm since

$$s \geq \tilde{\theta}_i \Leftrightarrow s \geq \mu_{\bar{\lambda}_i} + \sigma_{\bar{\lambda}_i} \cdot K' \Leftrightarrow \frac{s - \mu_{\bar{\lambda}_i}}{\sigma_{\bar{\lambda}_i}} \geq K' \Leftrightarrow s_{znorm} \geq K' \tag{14}$$

This means that the Z-Norm method approximately approaches the SST method when only imposter distribution is available in the system.

4   Experiments and Analysis

Experiments are performed on two public face databases, FERET and CAS-PEAL. For FERET database, the training set contains 1002 frontal images of 429 persons. Four testing sets Fafb, Fafc, DupI, DupII are used to test the performance under variations of expression, lighting and aging. For CAS-PEAL database, the training set contains 300 persons with 4 images per person. Six testing sets Accessory, Aging, Background, Distance, Expression and Lighting are used to test performance under corresponding variations. For each facial image used for training and testing, the preprocessing procedure consists of locating the centers of two eyes, geometrical transformation to place the center of two eyes on specific position. Each image is cropped to the size 64*64, processed by histogram equalization, and concatenated by rows to form a vector of 4096 dimension.

First, we conduct an experiment to show the effectiveness of Z-Norm on CAS-PEAL training set. The CAS-PEAL training set contains 300 subjects, 4 images per subject. Thus, for every subject, 6 genuine scores and 4784 imposter scores can be computed by using correlation method. The mean and variance of genuine and imposter scores before and after Z-Norm for each subject are shown in Fig. 2, whose horizontal axis denotes the subject number. The two figures in the top row show the mean and variance before Z-Norm and the bottom row shows them after Z-Norm. One can find that before Z-Norm, the distribution of scores fluctuates a lot for each subject, while the variances of imposter scores for each subject are almost the same. After Z-Norm, the mean and variance of imposter scores are 0 and 1 for each subject, which will greatly facilitate the setting of a uniform threshold.
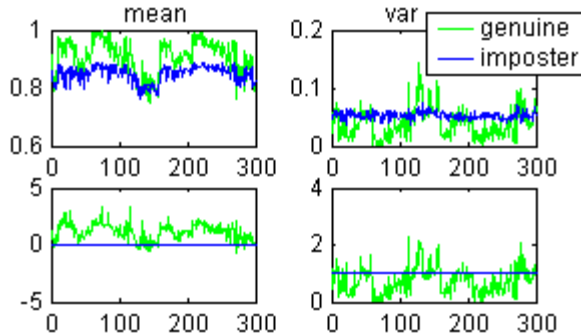
Fig. 2. Genuine and imposter scores distribution of each subject before and after Z-Norm

Secondly, we conduct experiments on FERET face database to verify the effect of Z-Norm, and SST (using Eq.12). Figure 3 shows the ROC curve of normalized corre-

lation method testing on the FERET-FB probe set. As we can see clearly, the ROC of Z-Norm and SST are approximate, and both outperform UT impressively.
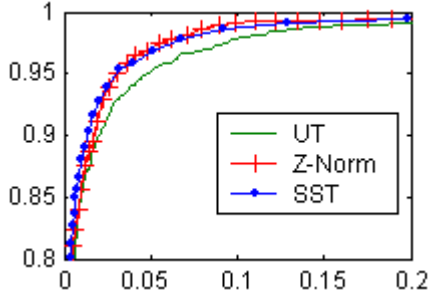
Fig. 3. ROC curves of correlation method with UT, Z-Norm and SST on fafb probe set

Finally, verification testing experiments are conducted on the FERET and CAS-PEAL face database to compare the EER of Z-Norm and UT. Since SST performs similarly to Z-Norm, it is not tested in these experiments. We test three different verification methods: Correlation, Eigenface, and Fisherface. Figure 4 and 5 show comparison results on the four probe sets of FERET and the six probe sets of CAS-PEAL respectively. For each classifier, the left column shows the EER by using UT, while the right column shows that of the Z-Norm method. From these figures, one can see that Z-Norm method can get better performance on FERET fafb, DupI, and DupII testing sets. On most of the testing sets of CAS-PEAL database, Z-Norm enhances the performance with only one exception (Fisherface on Aging test set).

The experimental comparisons on the FERET and CAS-PEAL database show that Z-Norm can generally enhance the performance for Correlation and Eigenface method. For Fisherface method, Z-Norm does not improve the verification performance obviously.
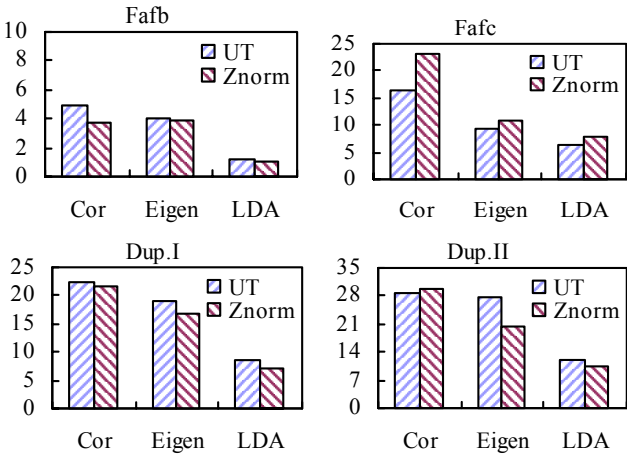
Fig. 4. EER on FERET database (Cor: Correlation, Eigen: Eigenface, Fisher: Fisherface)
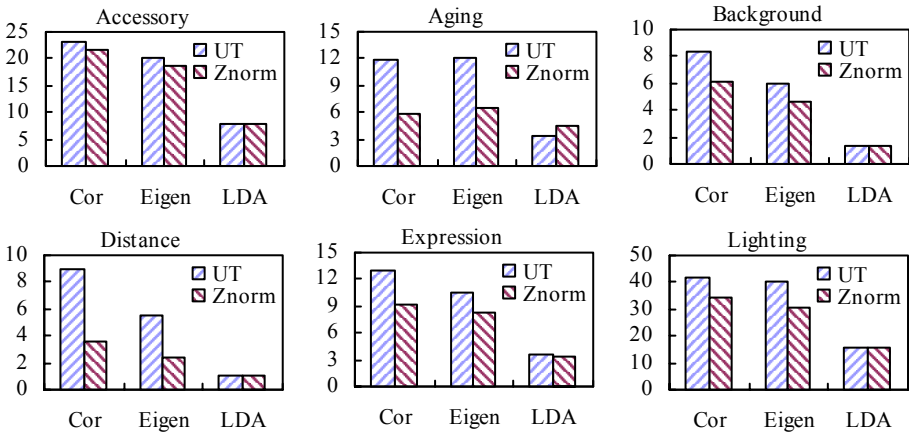
Fig. 5. EER on CAS-PEAL database (Cor: Correlation, Eigen: Eigenface, Fisher: Fisherface)

5   Conclusions

This paper discusses two methods, Subject Specific Threshold (SST) and Score Normalization (SN), aiming at the "score variation" problem caused by the significant variations in testing images due to varying lighting, pose, and expressions. We have theoretically and experimentally shown that a uniform threshold after score normalization using Z-Norm can approximate the SST method. In addition, our experiments on FERET and CAS-PEAL face database with three different face verification methods also illustrate the effectiveness of the Z-Norm method compared with the uniform threshold (UT) without score normalization.

Acknowledgements

References

1. A.K. Jain, A. Ross and S. Prabhakar: An Introduction to Biometric Recognition, IEEE Trans. on Circuits and Systems for Video Technology, Special Issue on Image and Video Based Biometrics, Vol. 14, No. 1, pp. 4-20, Jan. 2004.
2. A.K. Jain, K. Nandakumar, A. A. Ross: Score Normalization in Multimodal Biometric Systems, to appear in Pattern Recognition, 2005.
3. J.F. Aguilar, J. O. Garcia, J. G. Rodriguez: Target Dependent Score Normalization Techniques and Their Application to Signature Verification, Proc. International Conference on Bioinformatics and its Applications (ICBA), LNCS 3072, pp. 498-504, Dec. 2004.
4. C. Barras, J. L. Gauvain: Feature and Score Normalization for Speaker Verification of Cellular Data, Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Vol. II, pp. 49-52, Hong Kong, Apr. 2003.

5. C. Sanderson, K. K. Paliwal: Likelihood Normalization for Face Authentication in Variable Recording Conditions, Proc. IEEE International Conference on Image Processing (ICIP), Vol. I, pp. 301-304, Rochester, Sep. 2002.
6. F. Perronnin, J. L. Dugelay: Robust Score Normalization for Relational Approaches to Face Authentication, 12th European Signal Processing Conference, Sep. 2004.
7. P.J. Phillips, P. Grother, R. J. Micheals, D. M. Blackburn, E. Tabassi, J. M. Bone: Face Recognition Vendor Test 2002 Evaluation Report, NISTIR 6965, Mar. 2003.
8. P.J. Phillips, H. Moon, P. J. Rauss, and S. Rizvi: The FERET Evaluation Methodology for Face Recognition Algorithms, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 22, No. 10, Oct. 2000.
9. Bo Cao, Shiguang Shan, Xiaohua Zhang, Wen Gao: Baseline Evaluations on the CAS-PEAL-R1 Face Database, SinoBiometrics 2004, pp. 370-378.
10. R. Brunelli, T. Poggio: Face Recognition: Features vs. Templates, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 15, No. 10, pp. 1042-1053, Oct. 1993.
11. M. Turk and A. Pentland: Eigenface for Recognition, Journal of Cognitive Neuroscience, Vol. 3, No. 1, pp. 71-86, 1991.
12. P.N. Belhumeur, J. P. Hespanha, D. J. Kriegman: Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 19, Issue 7, Jul. 1997.

Gabor Feature Selection for Face Recognition Using Improved AdaBoost Learning

Linlin Shen[1], Li Bai[1], Daniel Bardsley[1], and Yangsheng Wang[2]

[1] School of Computer Science & IT, University of Nottingham, UK
{lls,bai,djb}@cs.nott.ac.uk
[2] Institute of Automaton, Chinese Academy of Sciences, P.R. China
wys@nlpr.ia.ac.cn

Abstract. Though AdaBoost has been widely used for feature selection and classifier learning, many of the selected features, or weak classifiers, are redundant. By incorporating mutual information into AdaBoost, we propose an improved boosting algorithm in this paper. The proposed method fully examines the redundancy between candidate classifiers and selected classifiers. The classifiers thus selected are both accurate and non-redundant. Experimental results show that the strong classifier learned using the proposed algorithm achieves a lower training error rate than AdaBoost. The proposed algorithm has also been applied to select discriminative Gabor features for face recognition. Even with the simple correlation distance measure and 1-NN classifier, the selected Gabor features achieve quite high recognition accuracy on the FERET database, where both expression and illumination variance exists. When only 140 features are used, the selected features achieve as high as 95.5% accuracy, which is about 2.5% higher than that of features selected by AdaBoost.

1 Introduction

Introduced by Freud and Schapire [1], AdaBoost has been successfully applied to object detection [2;3] and face recognition [4]. The essence of AdaBoost is to learn a number of very simple weak classifiers, which are then linearly combined into a single strong classifier. Whilst using only weak classifiers, recognition performance is just slightly better than random guessing. AdaBoost learning minimizes the upper bound on both training and generalization errors [5]. AdaBoost has been applied in applications to select Haar-like features [3] for face detection and recognition [4] as well as proving suitable for Gabor feature selection [6;7] for classification. Since minimizing the classification error rate is the ultimate objective of AdaBoost learning, the weak classifier with smallest weighted error is selected at each iteration. As a result, the learned classifiers are "individually" best and hence the combined strong classifier may not necessarily be optimum choice [8]. For feature selection, classifiers using similar features are more likely to be selected and redundancy will exist among some selected features. Stanz. Li etc. [8] proposed a floating search based algorithm, named FloatBoost, to eliminate these non-effective weak classifiers. A backtracking mechanism is applied to identify the unfavorable weak classifiers in terms of the classification error rate. The learned strong classifier thus consists of fewer weak classifiers and shows improved classification performance. During the learning process, each of the previously selected weak classifiers is individually removed from the

combined strong classifier and the results analyzed for improvements in the classification error rate. As a result of this, FloatBoosts computational requirements are approximately five times greater than that of AdaBoost. When the number of features is large (160,000+), as is normal in implementation scenarios utilising Gabor features, the FloatBoost training process could be unmanageable. A boosting algorithm, which is both effective in eliminating the non-effective classifiers and is computationally efficient, is required.

We propose in this paper a novel approach to address this issue. The proposed boosting algorithm uses the idea of mutual information for redundancy elimination. During the learning process, the mutual information between the candidate weak classifier and the selected weak classifiers is examined. As a result, the non-effective classifiers carrying information already captured by the selected feature/classifiers will be excluded. Since the mutual information is checked for only those candidate classifiers with small errors, the additional computation cost above that already required by AdaBoost is very low. In addition, mutual information is statistically calculated, which enables our technique to be classifier and decision method independent. It should be noted that several important distinctions exist between the method presented here and the InfoBoost algorithm [9]. While InfoBoost used the mutual information between the weak classifier and the class label for updating weights and weak classifier selection, our work still uses the classification error for learning, but introduces the idea of using mutual information between weak classifiers as an additional criteria. A threshold is used such that those classifiers with mutual information outside the given threshold, even with low classification error, are not selected. The experimental results show that the proposed boosting algorithm achieves a lower training error rate with fewer classifiers than other methods. Better performance has also been observed when the selected Gabor features are applied for face recognition.

2   AdaBoost Learning

Since the introduction of the original discrete AdaBoost algorithm, more general versions have been proposed. Where RealBoost was proposed to boost weak classifiers with real value output [10], AdaBoost.M1 and AdaBoost.MH [5] address the multi class problem. Here we focus on two class problems only, where weak classifiers take a discrete Boolean value. However, our method could be easily incorporated into the RealBoost or AdaBoost.MH algorithms to solve more general problems.

For the two class problem, a set of N labeled training samples is given as $(x_i, y_i), i = 1, 2, .., N$, where $y_i \in \{0,1\}$ is the class label associated with the sample $x_i \in R^n$. A large number of weak classifiers $h(x)$ could be generated to form the classifier learning pool, where $h(x) \in \{0,1\}$. The weak classifier could be very simple, for example, a threshold function on the *kth* coordinate of x in the *n*-dimensional space. The algorithm focuses on difficult training patterns, increasing their representation in successive training sets. Over a number of T rounds, T weak classifiers are selected to form the final strong classifier. In each of the iterations, the space of all possible weak classifiers is searched exhaustively to find the one with the lowest weighted classification error. This error is then used to update the weights such that the wrongly classified samples get their weights increased. The resulting strong clas-

sifier is a weighted linear combination of all T selected weak classifiers. See [1] for the listing of the AdaBoost algorithm.

3 Improved AdaBoost Learning

3.1 Entropy and Mutual Information (MI)

As a basic concept in information theory, entropy $H(X)$ is used to measure the uncertainty of a random variable (r.v.) X. If X is a discrete r.v., $H(X)$ can be defined as below:

$$H(X) = -\sum_x p(X = x)\lg(p(X = x))$$

(1)

Mutual information $I(Y;X)$ is a measure of general interdependence between two random variables X and Y:

$$I(Y;X) = H(X) + H(Y) - H(X,Y)$$

(2)

Using Bayes rule on conditional probabilities, Equation 2 can be rewritten as:

$$I(Y;X) = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X)$$

(3)

Since $H(Y)$ measures the priori uncertainty of Y and $H(Y \mid X)$ measures the conditional posteriori uncertainty of Y after X is observed, the mutual information $I(Y;X)$ measures how much the uncertainty of Y is reduced if X has been observed. It can be easily shown that if X and Y are independent, $H(X,Y) = H(X) + H(Y)$ and consequently their mutual information is zero.

The estimation of MI requires the value of marginal distribution $p(X)$, $p(Y)$ and the joint probability distribution $p(Y,X)$. For a r.v. with discrete values, the probability could be estimated by simply counting the number of possible cases and dividing that number with the total number of training samples. For a continuous r.v., its pdf could either be discretized by histogram estimation, or be approximated by Gaussian distribution.

3.2 The Proposed Algorithm

The proposed boosting algorithm incorporates the idea of MI to eliminate those non-effective weak classifiers. Each weak classifier $h(x) \in \{0,1\}$ is now considered as a r.v.. Before a new weak classifier is added, the MI between the new classifier and each of the selected ones is examined to make sure that the information carried by the new classifier has not been captured before. Given stage T where T-1 weak classifiers $\{h_{v(1)}, h_{v(2)}, \cdots h_{v(T-1)}\}$ have been selected, the function to measure the MI $R(h_j)$ between a candidate classifier h_j and the selected classifiers can be defined as follows:

$$R(h_j) = \max_t I(h_j, h_{v(t)}), t = 1,2,\cdots T-1$$

(4)

1) Input: N training samples $(x_i, y_i), i = 1, 2, .., N$ with m positive $(y_i = 1)$ and l negative $(y_i = 0)$ samples

2) Initialization: weights $w_{1,i} = \begin{cases} 1/2m, & \text{if } i \text{ is a positive sample} \\ 1/2l, & \text{if } i \text{ is a negative sample} \end{cases}$

3) For $t=1, ..., T$

    a) Normalize all weights

    b) Classifier selection and redundancy checking:

    For each candidate weak classifier h_j, calculate classification error

$$\varepsilon_j = \sum_i w_{t,i} |h_j(x_i) - y_i|$$

    For (;;)

      Choose $h_{t'}$ with lowest error $\varepsilon_{t'}$ from the candidate classifiers

      Calculate the MI $R(h_{t'})$ according to Eq. (4)

      If $R(h_{t'}) < TMI$

        The classifier found, $h_t = h_{t'}$, $\varepsilon_t = \varepsilon_{t'}$

        go to c)

      Else

        Remove $h_{t'}$ from the candidate list

    End loop

    c) Update weights: $w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$ with $e_i = \begin{cases} 1: x_i & \text{correctly classified} \\ 0: & \text{otherwise} \end{cases}$

    and $\beta_t = \varepsilon_t /(1 - \varepsilon_t)$

4) Final strong classifier: $H(x) = \begin{cases} 1 & \text{if } \sum_{t=1}^{T} \alpha_t h_t(x) > 1/2 \sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise} \end{cases}$ with $\alpha_t = \log(1/\beta_t)$

Fig. 1. Details of proposed boosting algorithm

The value of $R(h_j)$ can be directly used to decide whether the new classifier is redundant or not. The value is compared with a pre-defined Threshold Mutual Information (TMI) value, if it is bigger than the TMI, we can deduce that the information carried by the classifier has already been captured. Besides MI, the classification error of the weak classifier is also taken in to consideration, i.e., only those classifiers with small errors are selected. The classifiers thus selected will be both accurate and informative. When all the non-redundant classifiers are combined to form a strong classifier, better performance will be achieved. See Figure 1 for details of the algorithm.

4 Application for Gabor Feature Selection

Motivated by the functional similarity of Gabor filters with the cells in the visual cortex of human and mammalian visual systems, Daugman [11] presented evidence that such visual neurons could optimize the general uncertainty relations for resolution in space, spatial frequency and orientation. From an information theoretic view-

point, Okajima [12] derived Gabor functions as solutions for a certain mutual-information maximization problem. The work shows that the Gabor-type receptive field can extract the maximum possible information from local image regions. Researchers have also shown that Gabor features, when appropriately designed, are invariant against translation, rotation and scale [13]. Successful applications of Gabor filters in face recognition can be found in the FERET evaluation [14], where the Elastic Bunch Graph Matching method [15] gave the best performance. More recently, results from the face verification competition 2004 [16] also demonstrated the success of Gabor filter based approaches since both of the top two approaches apply Gabor filters for feature extraction. For face recognition applications, the number of Gabor filters used to convolve face images varies between applications, but usually 40 filters (5 scales and 8 orientations) are used [15;17-19]. However, due to the large number of convolution operations, the computation cost is quite high. Even when a parallel computer system has been used, it was reported in [17] that the convolution of a 128×128 pixel image with 40 Gabor filters took about 7 seconds. For global methods, the dimension of the feature vectors extracted are also incredibly large, e.g., 163,840 for images with a size of 64×64. Similar to the work of Viola and Jones [2], where AdaBoost was used to select Haar-like features for face detection, the task here is to select the most discriminative Gabor features for face recognition.

4.1 Gabor Features and the Personal Difference Space

Give a bank of 40 Gabor filters $\{\varphi_{u,v}(x,y),\ u=0,...,4, v=0,...7\}$, image features at different locations, frequency and orientation can be extracted by convolving the image $I(x,y)$ with the filters:

$$O_{u,v}(x,y) = \left| I(x,y) * \varphi_{u,v}(x,y) \right| \tag{5}$$

The resultant Gabor feature set thus consists of the convolution results of an input image $I(x,y)$ with all of the 40 Gabor filters:

$$S = \{O_{u,v}(x,y) : u \in \{0,...,4\}, v \in \{0,...,7\} \tag{6}$$

Figure 2 shows the magnitudes of Gabor representations of a face image with 5 scales and 8 orientations. A series of row vectors $\mathbf{O}_{u,v}$ could be converted out of $O_{u,v}(x,y)$ by concatenating its rows or columns, which are then concatenated together to generate a discriminative Gabor feature vector:

$$G(I) = \mathbf{O} = (\mathbf{O}_{0,0}\ \mathbf{O}_{0,1} \cdots \mathbf{O}_{4,7}) \tag{7}$$

Take an image with size 64×64 as an example, the convolution result will give $64\times64\times5\times8=163,840$ features. Since Gabor filter parameters are chosen empirically, it is our belief that a lot of redundant information is included, and therefore a feature selection mechanism should be used to choose the most useful features for classification.

To apply the proposed boosting algorithm, the difference space proposed in [20] is used here to convert the face recognition problem into a two class problem. Two classes, dissimilarities between faces of the same person (intra-personal space) and

dissimilarities between faces of the different people (extra-personal space) are defined. The two Gabor feature difference sets: CI (intra-personal difference) and CE (extra-personal difference) can be defined as:

$$CI = \left\{ \left\| G(I_p) - G(I_q) \right\|, p = q \right\}$$
$$CE = \left\{ \left\| G(I_p) - G(I_q) \right\|, p \neq q \right\} \tag{8}$$

where I_p and I_q are the facial images from people p and q respectively, and $G(\cdot)$ is the Gabor feature extraction operation as defined in (7). A set of M training samples in the difference space can now be described as $\{(g_1, y_1), \cdots (g_i, y_i), \cdots, g_M, y_M)\}$, $g_i = [x_1 x_2 \cdots x_n \cdots x_N]$, $y_i \in \{0,1\}$, where y_i is the class label (intra-personal or extra-personal) associated with sample g_i, N is the dimension of extracted Gabor features and $x_n = (g)_n = \left\| G(I_p) - G(I_q) \right\|_n = \left\| \mathbf{O}_p - \mathbf{O}_q \right\|_n$.
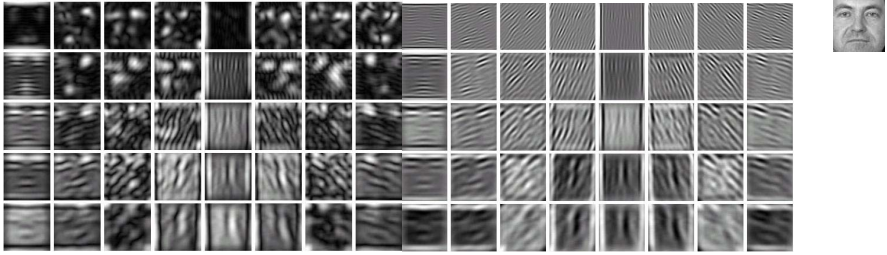
Fig. 2. Convolution result - (magnitude and real part) of an image with 40 Gabor filters

5　Experimental Results

5.1　Datasets

We analyze the performance of our algorithm using a subset of the FERET database, which is a standard test-bed for face recognition technologies [14]. 600 frontal face images corresponding to 200 subjects are extracted from the database for the experiments - each subject has three images of size 256×384 with 256 gray levels. The images were captured at different photo sessions so that they display variations in illumination and facial expression. Two images of each subject are randomly chosen for training, and the remaining one is used for testing. The following procedures were applied to normalize the face images prior to the experiments:

- The centres of the eyes of each image are manually marked
- Each image is rotated and scaled to align the centre of the eyes
- Each face image is cropped to the size of 64×64 to extract the facial region
- Each cropped face image is normalized to zero mean and unit variance

Figure 3 shows the sample images from the database. The first two rows are the example training images while the third row shows the example test images. We are trying to select the most discriminative Gabor features using the 400 (2 images per subject) training images, which are then applied to recognize the 200 test images. As

a result, 200 intra-personal and 1600 extra-personal Gabor feature difference samples are randomly generated. Once a small set of discriminative Gabor features are learned by applying AdaBoost or our proposed algorithm on the training samples, they are used for face recognition.

Fig. 3. Sample images used in experiments

5.2   Selected Gabor Features

We first applied AdaBoost on the training samples to select 200 Gabor features for intra-personal and extra-personal difference classification. To show the existence of redundancy among AdaBoost selected features (weak classifiers), the MI $R(h_j)$ for each selected feature is shown in Figure 4a. It can be observed from the figure that some of the features are highly redundant, e.g. the MI of features with numbers 149, 177 and 180 is greater than 0.99. The redundancy among selected features increases with the number of features, it is this undesired redundancy that we aim to eliminate or reduce. We have also shown the MI data for features selected with our proposed algorithm in Figure 4b (with TMI=0.1). Due to the introduction of TMI, all the selected features now show MI values of less than 0.1 and thus we can conclude that the features are informative and non-redundant.

(a)   (b)
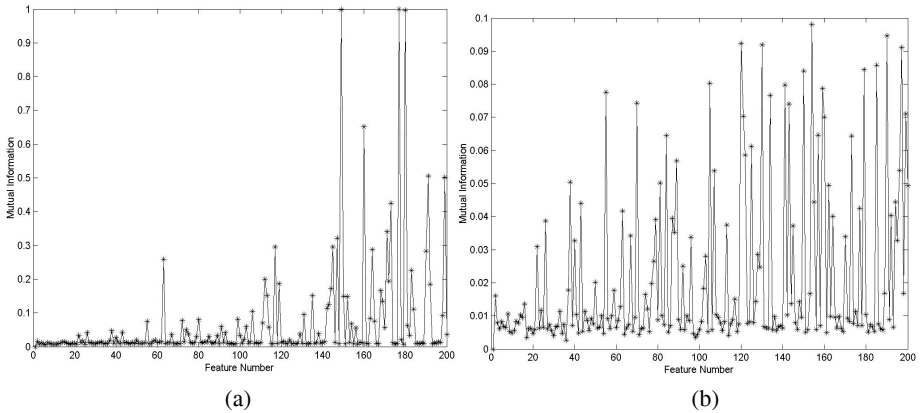
Fig. 4. MI of features selected by AdaBoost a); Proposed algorithm b)

| | | | | | | |
| (a) | (b) | (c) | (d) | (e) | (f) | (g) |
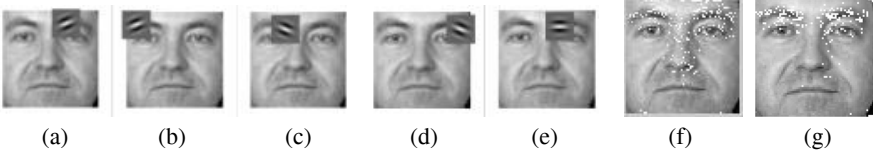
Fig. 5. First five selected Gabor features (a)-(e); and the 200 feature points selected by our algorithm (f); and AdaBoost (g)

Figure 5 (a)-(e) show the first five Gabor features selected by the proposed algorithm, with locations of the first 200 Gabor features selected by our algorithm and AdaBoost shown in (f) and (g) respectively. The features are overlapped on a typical face image in the database. It is interesting to see that most of the selected Gabor features are located around the prominent facial features such as eyebrows, eyes, nose and the chin. This indicates that these regions are more robust against the variations is expression and illumination. This result is agrees with the fact that the eye and eyebrow regions remain relatively stable when a persons expression changes. While almost all of the AdaBoost selected Gabor features are crowded in the eyebrow and eye regions, the features selected by the proposed algorithm are more widely distributed with some of the features located around the nose also being included.

5.3   Recognition Performance Comparison

Once the most discriminant Gabor features are selected, we are now able to apply them to the face recognition problem. In this experiment, 200 Gabor features selected by AdaBoost and by our proposed algorithm (TMI=0.1) are directly used for a similarity comparison, without any further processing. The normalized correlation distance measure and the nearest neighbor classifier are used. Figure 6 shows the recognition performance of Gabor features selected by both AdaBoost and our algorithm on the 200 test images. When 140 features are used, the highest accuracy achieved by AdaBoost and the proposed algorithm, are 93% and 95.5% respectively. Since the MI values for all of the first 60 features are quite small, the proposed algorithm starts by picking up much the same features as AdaBoost. However, once the number of features increases, AdaBoost starts to pick highly redundant features. The improved recognition rate accuracy over AdaBoost caused by the use of features selected using the proposed algorithm proves the usefulness of our techniques in eliminating redundancy.

5.4   Algorithm Complexity and Determination of TMI

Due to the introduction of mutual information, the proposed boosting algorithm required longer training time than that required by AdaBoost. However, the only computation cost added to AdaBoost is the loop to calculate MI values for redundancy checking, see Figure 1 for details. Table 1 shows the Average Number of Loops (ANL) required in each iteration and the corresponding TMI. The table shows that the computation burden added by the introduction of MI is actually very low (ANL is normally less than 10). As a result, the training time required by the proposed algorithm in our experiments is only about 0.1 times greater than that of AdaBoost.
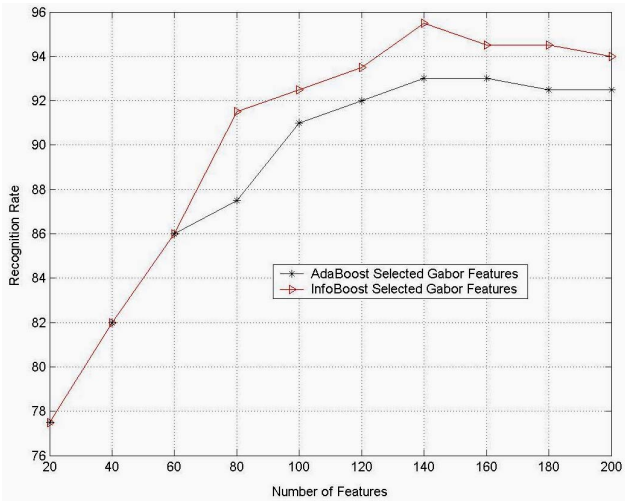
Fig. 6. Recognition performance

Table 1. ANL for different TMI

TMI	0.08	0.09	0.10	0.11	0.12
ANL	8.42	8.07	7.25	5.43	3.25

As seen from the table, the higher the value of TMI, the less ANL required, i.e. the faster training speed. Actually AdaBoost can be seen as a special case of our algorithm when the value of TMI is set as 1. In this case, the features, or weak classifiers selected by the proposed algorithm will be exactly the same as those chosen by AdaBoost. The value of TMI needs to be selected appropriately to make sure that selected features are both non-redundant and useful for classification. A cross-validation set could be used to determine the TMI for common classification problems. As shown in Figure 4, since the redundancy increases with the number of selected features, an adaptive TMI, which increases with the number of features, might be more suitable.

6   Conclusions

A new and novel boosting algorithm has been proposed in this paper. By examining the mutual information between the candidate classifier and the selected weak classifiers, redundant classifiers can be excluded. The proposed algorithm has also been applied to select discriminant Gabor features for face recognition. The results show the advantage of features selected using our techniques over those learned by AdaBoost, specifically, Gabor features selected using our method are both non-redundant and achieve higher recognition accuracy on the subset of FERET database used for testing. Though the proposed algorithm addresses weak classifiers with discrete values only, it can be easily extended to classifiers with continuous values, where histogram estimation could be used to estimate the probability distribution.

Whilst the selected features/classifiers are non-redundant and thus achieve higher accuracy for classification, the computation burden added to AdaBoost is very low. In our experiments, the training time of the proposed algorithm is only about 0.1 times longer than that of AdaBoost.

It should also be noted that though the face images are normalized manually in this paper, the work in [21] proved the robustness of non-redundant features against the mis-alignment of automatic face detection system. We aim to prove the tolerance of our system against the localized in the future. Currently, we are working on designing an adaptive TMI determination algorithm to improve the proposed boosting algorithm further.

Reference

1. Y. Freund and R. Schapire, "A decision-theoristic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119-139, 2000.
2. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition* Kauai, Hawaii: 2001.
3. R. Lienhart and J. Maydt, "An extended set of Haar-like features for rapid object detection," in *Proc. IEEE Conference on Image Processing* 2002, pp. 900-903.
4. J. Michael and P. Viola, "Face recognition using boosted local features," in *Proc. of International Conference on Computer Vision (ICCV)* 2003.
5. Y. Freund and R. Schapire, "A short introduction to boosting," *Journal of Japanese Society for Artifical Intelligence*, vol. 14, no. 5, pp. 771-780, 1999.
6. L. Shen and L. Bai, "AdaBoost Gabor Feature Selection for Classification," in *Proc. of Image and Vision Computing NewZealand* 2004, pp. 77-83.
7. P. Yang, S. G. Shan, W. Gao, S. Z. Li, and D. Zhang, "Face recognition using ada-boosted gabor features," *Sixth Ieee International Conference on Automatic Face and Gesture Recognition, Proceedings*, pp. 356-361, 2004.
8. S. Z. Li and Z. Q. Zhang, "FloatBoost learning and statistical face detection," *Ieee Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1112-1123, 2004.
9. J. Aslam, "Improving Algorithms for Boosting," in *Proceedings of the Thirteenth Annual Conference on Computational Learning Theory* 2000, pp. 200-207.
10. R. E. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," *Machine Learning*, vol. 37, no. 3, pp. 297-336, 1999.
11. J. G. Daugman, "Uncertainty Relation for Resolution in Space, Spatial- Frequency, and Orientation Optimized by Two-Dimensional Visual Cortical Filters," *Journal of the Optical Society of America A-Optics Image Science and Vision*, vol. 2, no. 7, pp. 1160-1169, 1985.
12. K. Okajima, "Two-dimensional Gabor-type receptive field as derived by mutual information maximization," *Neural Networks*, vol. 11, no. 3, pp. 441-447, 1998.
13. V. Kyrki, J. K. Kamarainen, and H. Kalviainen, "Simple Gabor feature space for invariant object recognition," *Pattern Recognition Letters*, vol. 25, no. 3, pp. 311-318, 2004.
14. P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *Ieee Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090-1104, 2000.
15. L. Wiskott, J. M. Fellous, N. Kruger, and C. vonderMalsburg, "Face recognition by elastic bunch graph matching," *Ieee Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 775-779, 1997.

16. K. Messer, J. Kittler, M. Sadeghi, M. Hamouz, A. Kostin, F. Cardinaux, S. Marcel, S. Bengio, C. Sanderson, N. Poh, Y. Rondriguez, J. Czyz, L. Vandendorpe, C. McCool, S. Lowther, S. Sridharan, V. Chandran, R. P. Palacios, E. Vidal, L. Bai, L. Shen, Y. Wang, Y. H. Chiang, H. C. Liu, Y. P. Huang, A. Heinrichs, M. Miiller, A. Tewes, C. v. d. Malsburg, R. Wiirtz, Z. G. Wang, F. Xue, Y. Ma, Q. Yang, C. Fang, X. Q. Ding, S. Lucey, R. Goss, and H. Schneiderman, "Face authentication test on the BANCA database," in *Proc. of International Conference on Pattern Recognition* Cambridge, UK: 2004.
17. M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lange, C. Vandermalsburg, R. P. Wurtz, and W. Konen, "Distortion invariant object recognition in the Dynamic Link Architecture," *Ieee Transactions on Computers*, vol. 42, no. 3, pp. 300-311, 1993.
18. C. J. Liu and H. Wechsler, "Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition," *Ieee Transactions on Image Processing*, vol. 11, no. 4, pp. 467-476, 2002.
19. L. Shen and L. Bai, "Face recognition based on Gabor features using kernel methods," in *Proc. of the 6th IEEE Conference on Face and Gesture Recognition* Korea: 2004, pp. 170-175.
20. P. J. Phillips, "Support vector machines applied to face recognition," in *Proceedings of the 1998 conference on Advances in neural information processing systems II* MIT press, 1999, pp. 803-809.
21. G. Littlewort, M. S. Bartlett, I. R. Fasel, J. Chenu, and J. R. Movellan, "Analysis of machine learning methods for real-time recognition of expressions from video," 2003.

An Automatic Method
of Building 3D Morphable Face Model*

Hui Guo, Chengming Liu, and Liming Zhang

Fudan University, Dept of Electronic Engineering
Laboratory of Image and Intelligence, 220 Handan Road, Shanghai, China
{guohui,cmliu,lmzhang}@fudan.edu.cn

Abstract. In this paper, we propose an automatic method to set up a 3D morphable face model. The main idea is to combine the algorithm of Fast-AAM with Thin Plate Splines (TPS) for 3D data alignment. Our method is better than the algorithm of optical flow proposed in [3][5] because it avoids the local minima problem, and also it is better than the adaptive multi-resolution fitting algorithm [8][9] in 3D alignment because it is fully automatic and the latter needs to be manually initialized. Result shows that our method is practicable in speed and accuracy.

1   Introduction

With the development of face detection and recognition, traditional method based on 2D images cannot handle the issues of real 3D face easily with the problems of pose, illumination and expression (PIE). However, 3D face model has great advantage in 3D information analysis and PIE face recognition, so it is urgent to set up a morphable 3D model for Chinese or Asia faces and study the method for it.

In 2003, Blanz and Vetter proposed a new method for 3D expression and recognition. This method uses PCA to model the shape and texture of 3D faces, and set up a unified mathematical expression of 3D morphable model with the 3D mean face and a few parameters [4]. Generally, a 3D face includes 100,000 to 1000,000 vertexes. Once these vertexes for each 3D face data can be arranged in order of shape V and texture C, we can get the mean shape $\overline{V}$ and texture $\overline{C}$ from several 3D data through PCA, and the original 3D data can be expressed as follows:

$$V = \overline{V} + \sum_{i=1}^{n_v} a_i V_i ; C = \overline{C} + \sum_{j=1}^{n_c} \beta_j C_j \tag{1}$$

In order to make 3D PCA, we should first get the mean face in Eq.(1). Since the distribution of these known vertexes of 3D face data depends upon the complexity of face surface's texture, their displacement are not uniform, shown as figure 1(a). In addition, the vertexes' distribution between different people is also different, shown as figure 1(b) and 1(c), which brings difficulty in sorting these vertexes. Besides, the diversity of shooting position, facial features and different height of people also affect

---

the quality of 3D model. Before the PCA process, preprocessing, re-sampling and face alignment are considered at first. That is, to put the vertexes of key points such as eye and nose tip in the same position. Proper alignment leads to greater interrelation between the data in the semantic positions, fewer parameters in Eq.(1) and better representation of the 3D morphable model.

<div align="center">(a)   (b)   (c)</div>
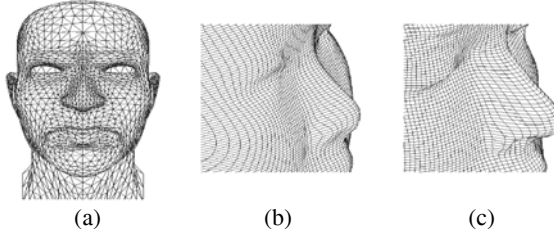
Fig. 1. 3D vertex and meshes (a)not uniform vertex (b) (c) different distribution

The corresponding displacement can be worked out by complicated optical flow algorithm [4][1], as Blanz and Vetter applied in their German 3D face set derived from 3D laser scans. Our implementation, however, could not give a satisfying result using optical flow for our 3D face data. It is due to the large variance between different faces, so that the brightness conservation assumption does not work. Moreover, it is very time consuming for mass optimizing. And the adaptive multi-resolution fitting algorithm employed by Beijing University of Technology[8] and Institute of Automation, CAS[7] needs to be manually initialized and the total progress costs a long time.

To tackle the lack of a fully automatic and precise 3D modeling method, we propose a novel strategy. The main idea of our method is to combine the algorithm of Fast-AAM with Thin Plate Splines (TPS) for 3D data alignment. We use the Fast-AAM to automatically search the facial feature points and then employ the TPS to align all the other vertexes on a 3D face based on its feature points. After alignment, Principle Component Analysis (PCA) is used to build the 3D morphable model. Testing results show that our method is quite acceptable.

This paper is arranged as follows: Section 2 introduces our method of automatic 3D face modeling, which includes 2 algorithms: Fast-AAM and TPS, Section 3 gives some experimental results; and the discussion and conclusion is drawn in Section 4.

2   The Automatic Modeling Method

Figure 2 shows the flow chart of building the 3D morphable model. It can be seen from Fig 2 that the preprocessing and resampling is first introduces to adjust the 3D sample data into an approximate uniformity. In AAM, a few landmarks is trained from several original 3D data as standard feature points to get the AAM model, and then use the algorithm of Fast-AAM based on it to find all the corresponding feature points in all the other 3D faces. The module of alignment makes use of TPS to reorder the remained points in 3D faces according to the feature points. After that, we use PCA to build the 3D morphable model. Each step in the process will be explained in this section separately.
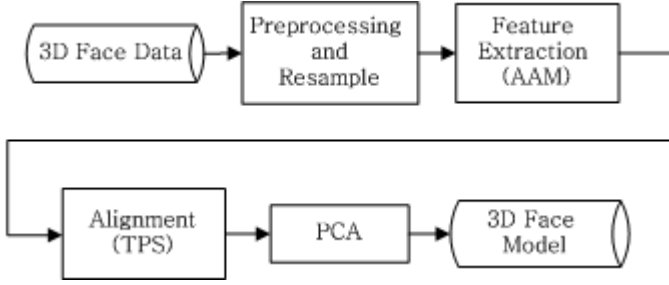
Fig. 2. Flow Chart of the Automatic 3D Modeling

2.1 Preprocessing and Resampling

To homogenize the unequally distributed vertexes, resampling is then employed. Since human face is distributed on the surface of an ellipsoid, a cylindrical coordinates (h, ϕ, r) is introduced to resample the 3D faces. The transformation from Cartesian coordinate to it is shown as follows.

$$r = \sqrt{x^2 + y^2 + z^2}; h = z; \phi = tg^{-1}(y/x) \tag{2}$$

As introduce in [4], the shape variable r and three texture variable R,G and B in cylindrical coordinates can be resampled from the facial surface by parameters h and ϕ. Then, each vertex can be described as follows:

$$I(h, \phi) = (r(h, \phi), R(h, \phi), G(h, \phi), B(h, \phi))^T$$
$$h, \phi \in \{1, \dots, 360\} \tag{3}$$

Where r is the depth information, and (R,G,B) is its texture information. Thus, each vertex $v(x, y, z)$ can find an exclusive expression $I(h, \phi)$ in cylindrical coordinates.

2.2 Feature Extraction (AAM)

After resampling, the points in 3D faces is uniformly distributed, but not point-to-point corresponding. So, we adopt Fast-AAM method which usually used in 2D face tracking to automatically find K obvious feature points as eyes, nose tip and corners of the mouth in every faces [5]. Then TPS algorithm is introduced to align the remaining points.

Suppose a 3D face in cylindrical coordinates to be defined K feature points shown in Fig. 3 (K = 60). We put the cylindrical coordinates of these K points in vector L as follows:

$$L = (h_1, \phi_1, h_2, \phi_2, \dots, h_K, \phi_K)^T \tag{4}$$

The contour of L shown as Fig. 3 constitutes a mesh M, and we put the texture of each feature point in M into a vector S:

$$S = (R_1, G_1, B_1, R_2, G_2, B_2, \dots, R_K, G_K, B_K)^T \tag{5}$$

By AAM, we can find the corresponding relations between the feature points' vector L of different resampled faces through adjusting the position and applying the triangle affine transformation in mesh M. AAM model in PCA format is as follows:
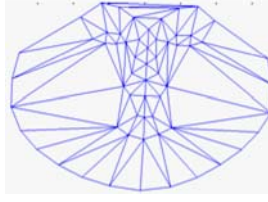
Fig. 3. Feature Points and Meshs

$$L = L_0 + \sum_{i=1}^{n} p_i L_i; S(c) = S_0 + \sum_{j=1}^{m} \lambda_i S_i \quad \forall c \in M \tag{6}$$

Where p_i and λ_i are the shape and texture coefficients of PCA, and L_i, S_j represent the basis vector of shape and texture respectively. The number of PCA basis vector for shape and texture is n and m. Our AAM model is trained by 40 resampled 3D faces in cylindrical coordinates which partly used in our 3D modeling.

The fitting cost function of FAST-AAM [5][6]is defined as follows:

$$J = \left\| S_0(c) + \sum_{j=1}^{m} \lambda_j S_j - I(W(c; p + \Delta p)) \right\|^2$$

$$= \sum_c [S_0(c) + \sum_{j=1}^{m} \lambda_j S_j - I(W(c; p + \Delta p))]^2, c \in M \tag{7}$$

Where I is the input image, and W is the affine warp according to the parameter p. If we denote the linear subspace spanned by a collection of vectors S_j by $span(S_j)$ and its orthogonal complement by $span(S_j)^{\perp}$, Eq. (7) can be rewritten as:

$$J = \left\| S_0(c) - I(W(c; p + \Delta p)) \right\|^2_{span(A_i)^{\perp}} + \left\| S_0(c) + \sum_{j=1}^{m} \lambda_j S_j - I(W(c; p + \Delta p)) \right\|^2_{span(A_i)} \tag{8}$$

The minimum value of J can be found sequentially by minimizing the first term with respect to p alone, and then using that optimal value of p as a constant to minimize the second term with respect to λ_i [5]. But in this paper, we ignore the texture coefficient λ_i since the texture part has no use for our method. So after we get the parameter p, the feature points vector, L, in Eq. (6) can be easily calculated. The fitting cost function can be simplified as follows:

$$J = \left\| S_0(c) - I(W(c; p + \Delta p)) \right\|^2_{span(A_i)^{\perp}} = \sum_c [S_o(c) - I(W(c; p + \Delta p))]^2 \tag{9}$$

Given a point c in mesh M, the destination of this point under affine warp is $W(c; p)$. The mesh M is triangulated as many patches based on these feature points in each face. So for every patch, we can calculate its parameters for affine warp $(a_1, a_2, \ldots, a_6)$ between this patch and the corresponding one in the reference face from the position of known feature points.

$$W(c; p) = (a_1 + a_2 \cdot \phi + a_3 \cdot h, a_4 + a_5 \cdot \phi + a_6 \cdot h)^T \tag{10}$$

The least square's solution of Eq. (9) for Δp is:

$$\Delta p = H^{-1} \sum_c [\nabla I \frac{\partial W}{\partial p}]^T [S_0(c) - I(W(c; p))] \quad H = \sum_c [\nabla I \frac{\partial W}{\partial p}]^T [\nabla I \frac{\partial W}{\partial p}] \tag{11}$$

Since the Hessian matrix H depends on p, it needs to be recomputed in every iteration. So it is a slow operation.

We introduced algorithm in [5] to calculate the affine warp of the mean shape S_0 instead of the input image. Eq.(9) is modified as

$$J = \sum_c [S_o(W(c; \Delta p)) - I(W(c; p))]^2 \tag{12}$$

Since now there is nothing in Hessian matrix depends on p, it can be precomputed and far more efficient than the original one can be obtained:

$$\Delta p = H^{-1} \sum_c [\nabla S_0 \frac{\partial W}{\partial p}]^T [I(W(c; p)) - S_0(c)] \quad H = \sum_c [\nabla S_0 \frac{\partial W}{\partial p}]^T [\nabla S_0 \frac{\partial W}{\partial p}] \tag{13}$$

For every input image I, we get the shape parameter p by optimizing the function Eq.(12) in the orthogonal subspace of texture and its L by Eq. (6). Thus, we automatically find K obvious feature points of each face and cost just a little time.

2.3 Alignment (TPS)

Thin Plate Spline [2], which proposed in late 80s, is one of the tools for interpolating surfaces over scattered data by solving a set of equations. In 2004, we applied this method in aligning different faces [9] by manually marked feature points. This paper applies the method in the mesh M after AAM feature extraction.

Suppose the correspondent displacement of vertices between I and I_0 is $\Delta c = (\Delta h_j(h, \phi), \Delta \phi_j(h, \phi)); j = 1, 2, \ldots, N$. We require a continuous function f, which can project each vertex of face I into that of the reference face I_0. Thus, we can get the displacement of each points by $[f(c) - c], \forall c \in M$.

We choose a reference face I_0 in cylindrical coordinates of 3D data, and the K feature points are already known by real-time AAM, $L^0 \{l_i^0\}; i = 1, \ldots, K$. The feature points of an input image I is $L\{l^i\}; i = 1, \ldots, K$, where l_i^0 and l^i is the i-th point in the shape vector. Suppose the mapping function f from L to L^0 be as follows:

$$f(l^i) = l_i^0 \qquad \forall i = 1, \ldots, K \tag{14}$$

Given such a mapping function, we can project each vertex of face I into that of the reference face I_0. We define the function f as a weighted sum of function U, called TPS function.

$$f(c) = \sum_{i=1}^{60} \omega_i U(\|c - l^i\|) + a_0 + a_\phi l^i(\phi) + a_h l^i(h) \tag{15}$$

Where $\omega_i, a_0, a_h, a_\phi$ are the weights, and c denotes the position vector (h, ϕ) of any vertex in face I. U is the kernel function of TPS [2]. And the constraints of the weights are:

$$\sum_{i=1}^{K} \omega_i = 0, \ \sum_{i=1}^{K} \omega_i l^i(h) = 0, \ \sum_{i=1}^{K} \omega_i l^i(\phi) = 0 \qquad (16)$$

Eq.(16) ensures that when c is at infinity, the first part in the right side of Eq. (15) sums to zero. Commonly, the right side of Eq.(15) is divided into two parts: a sum of function U which can be shown to be bounded and asymptotically flat, and an affine part representing the behavior of f when c is at infinity. Eqs.(14)-(16) include K+3 equations with K+3 unknown parameters, so it is easy to get mapping function f. The detailed algorithm is in our paper [9].

Thus, the dense point-to-point correspondence between each face and the reference face can be easily decided by using the TPS function. The alignment is then accomplished by reorder the shape and texture vector of each face according to the displacement from the reference face.

2.4 Morphable Model (PCA)

To keep constancy, we replace the face data into Cartesian coordinate after automatic alignment in cylindrical coordinates.

$$\mathbf{V} = (x_1, y_1, z_1, x_2, \ldots, x_N, y_N, z_N)^T,$$
$$\mathbf{C} = (R_1, G_1, B_1, R_2 \ldots, R_N, G_N, B_N)^T \qquad (17)$$

Where N is the number of vertexes in mesh M. We perform a Principle Component Analysis (PCA) on the set of vectors. Ignoring the correlation of shape and texture, we perform the PCA separately on shape V and texture C like Eq.(1).

For shape V, we calculate the average shape model $\overline{V}$ and the covariance matrix $Con = \frac{1}{k}\sum_{i=1}^{k}(\mathbf{V}_i - \overline{\mathbf{V}})(\mathbf{V}_i - \overline{\mathbf{V}})^T$, where k is the number of sample faces. After calculating the eigenvalue and eigenvector of Con, 98% of total energy is selected to reduce the dimension of shape and texture space. In our experiment we take 24 shape and 33 texture PCA bases for k=40.

Thus, the morphable 3D face model is shown as following:

$$\mathbf{V} = \overline{\mathbf{V}} + \sum_{i=1}^{24} \alpha_i \mathbf{V}_i, \quad \mathbf{C} = \overline{\mathbf{C}} + \sum_{i=1}^{33} \beta_i \mathbf{C}_i \qquad (18)$$

Where α_i and β_i are the shape and texture coefficients of PCA, and V_i and C_i is the basis vectors of shape and texture respectively.

3 Experiment Result and Discussion

Our morphable model of 3D faces was generated from 40 Chinese individuals' 3D face data, including 20 males and 20 females, which come from synthesis of two cameras. The performance estimation of our proposed method is shown as follows.

a) The Performance of Speed
The work is performed under the environment of MatLab7.0 in a work station with PIV 3.6GHz processor. Ignoring the preprocessing time, for every 3D data in the

training set, the total modeling process costs about 4.8s, in which the FAST-AAM costs about 0.6s and the TPS alignment costs about 4.2s. Compared with the adaptive multi-resolution fitting algorithm [8,9], which costs about 2 min to process a 3D data, our method is far more efficient.

b) The Performance of Accuracy

The resampled data in cylindrical coordinates from our 3D face training set is about 360x360 pixels. The mean residual of the result of FAST-AAM and manually marked feature points is 2.9646 pixels. If we ignore the points of face counter which have little effect in alignment, the mean residual is about 1.2336 pixels, 0.3427% of the image witdh.

Figure 4 shows 8 faces, the upper row shows the original 3D data and the bottom row shows the reconstruction of these faces. Under the same illumination and pose, the mean PSNR between 2D projection of the original faces and that of the reconstructed faces is 35.9526dB.

The definition of PSNR is as follows:

$$\text{PSNR}=20\log_{10}(\frac{255}{RMSE}) \quad RMSE = \frac{\sqrt{\sum_i \sum_j [I_1(i,j)-I_2(i,j)]^2}}{N} \tag{19}$$

where I_1 and I_2 are two input images with size NxN pixels.

Table 1 shows the mean PSNR of the 2D projection between the original face and reconstructed one in different illumination and pose:
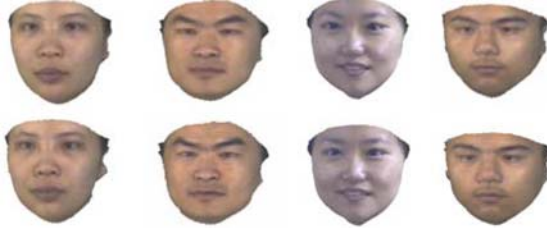
Fig. 4. The original faces and the reconstructed ones

Table 1. (The unit is dB)

Pose	-30$^{\text{o}}$	-15$^{\text{o}}$	0$^{\text{o}}$	15$^{\text{o}}$	30$^{\text{o}}$
With illumination	37.3886	35.8885	34.7367	35.2006	36.0384
Without illumination	38.3144	36.6213	35.4162	36.0351	37.2346

c) The Performance of Representation

Our morphable model of 3D faces is show in Fig. 5. The first two columns show the coefficient variation of the first two shape principle component under the mean texture, visualized by adding $\pm\frac{1}{2}\sqrt{ei\ gv}$, where eigv is eigenvalue of the corresponding shape principle component. And the forth and fifth columns show the variation of the first two texture principle component under the mean shape, visualized by adding

$\pm\frac{1}{2}\sqrt{\text{ei gc}}$,where eigc is the corresponding texture principle component. The middle

column shows the mean face with α_i and β_i equals to zeros which presents the mean texture and mean shape. Fig. 6. shows different 3D shapes with the same texture (mean face texture).

4   Conclusion

To tackle the lack of a fully automatic and precise 3D modeling method, we introduce a novel strategy, which combine the algorithm of Fast-AAM with Thin Plate Splines (TPS) for 3D data alignment.

The results of reconstruction show that our strategy can automatically set up a 3D morphable model with high speed and enough precision. By changing the coefficients α_i and β_i in Eq. (18), we can reconstruct different 3D faces which generally cover the training set and get 3D face with different shape and texture.

Although our original data in this paper are not as precise as the foreign ones, which are recorded with a laser scanner, it can still build a 3D morphable model in practicality.
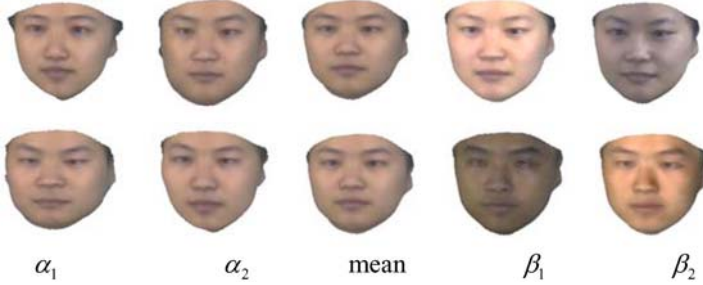
Fig. 5. 3D Morphable faces

Fig. 6. Different 3D Shapes with same texture

Recently, a laser scanner has been used to create our own 3D faces database, so our future work will focus on setting up a more comprehensive Chinese 3D faces morphable model from the precise original data.

Acknowledgement

References

1. B.K.P. Horn and B.G. Schunck, "Determining Optical Flow", *Artificial Intelligence*, vol.17, pp.185-203, 1981
2. Fred L. Bookstein "Priniple Warps: Thin-Plate Spline and the Decomposition of Deformation", *IEEE trans on PAMI* VOL11,No 6,June:567-585,1989
3. V. Blanz, T. Vetter, "A Morphable Model For the Synthesis Of 3D Faces", in SIGGRAPH'99 Conference Proceedings. ACM SIGGRAPH, 1999.
4. V. Blanz and T. Vetter, "Face Recognition Based on Fitting a 3D Morphable Model", *IEEE trans on PAMI*, Sept 2003
5. Baker S, Matthews I "Lucas-Kanade 20 years on: A unifying framework", International Journal Of Computer Vision 56 (3): 221-255 Feb-Mar 2004
6. Matthews I, Baker S "Active appearance models revisited", International Journal Of Computer Vision 60 (2): 135-164 Nov 2004
7. C.Xu, L.Quan, Y.Wang, "Adaptive Multi-resolution Fitting and its Application to Realistic Head Modeling", *IEEE Geometric Modeling and Processong(2004) 345-348*
8. Hu Yongli, Yin Baocai, "Researvh on Key Technology in Construction of a Chinese 3D Face Database", ISSN 1000-1239/CN 11-1777/TP 42(4): 622-628, 2005
9. Guohui,Jiangjiayan "Building a 3D Morphable Face Model by Using Thin Plate Splines for Face Reconstruction", *FGR'04*

Procrustes Analysis and Moore-Penrose Inverse Based Classifiers for Face Recognition

K.R. Sujith[*] and Gurumurthi V. Ramanan[**]

AU-KBC Research Centre,
MIT Campus of Anna University,
Chromepet, Chennai 600 044, India
{sujithkr,gurumurthi}@au-kbc.org

Abstract. We propose two new classifiers, one based on the classical Procrustes analysis and the other on the Moore-Penrose inverse in the context of face recognition. The Procrustes based classifier has recognition rates of 97.5%, 96.19%, 71.40% and 96.22% for the ORL, YALE, GIT and the FERET database respectively. The Moore-Penrose classifier has comparative recognition rates of 98%, 99.04%, 87.40% and 96.22% for the same databases. In addition to these classifiers, we also propose new parameters that are useful for comparing classifiers based on their discriminatory power and not just on their recognition rates. We also compare the performance of our classifiers with the baseline PCA and LDA techniques as well as the recently proposed discriminative common vectors technique for the above face databases.

1 Introduction

The past few decades have seen a flurry of classifier algorithms in the area of face recognition. In this paper, we formulate the face recognition problem in a manner that admits application of the classical Procrustes analysis as well as the Moore-Penrose inverse for minimizing a suitably defined error function [1–5]. Our objective of bringing the intraclass distances as low as possible and the interclass distances as high as possible is similar to that of LDA [6]. Even though there seem to be no previous applications of Procrustes analysis to face recognition, it has been used in the context of alignment of faces in images [2]. For an elegant one-page exposition of this method, we refer to the classical text of Golub and Van Loan [1]. Applying the Moore-Penrose inverse for pattern classification is not new. Wee's application of the Moore-Penrose inverse to derive an adaptive learning algorithm is well known. At present, it is not clear how it is related to our technique [4, 5].

The paper is organized as follows. In Section 2, we formulate the problem in a manner that admits its possible solution through the method of Procrustes

---

[*] The results of this paper are part of the MS thesis done under the direction of the second author

[**] Corresponding author

analysis as well as the generalized inverse technique. These two approaches are discussed in Sections 3 and 4. In Section 5, we present the recognition rates of our classifiers using the ORL,YALE, GIT and the FERET face databases. We also compare our classifiers with the baseline PCA and LDA as well as the recently proposed discriminative common vectors (DCV) technique [7]. In Section 6, we propose new parameters that are useful for comparing classifiers based on their discriminatory power and not on their recognition rates alone.

2   Formulation of the Problem

Let n and p be the number and dimension of feature vectors, respectively. Let D be a $n \times p$ matrix with the feature vectors as its rows. We aim to find a transformed (sub)space where the feature vectors can be identified "correctly". Let

$$B - DQ = \hat{\epsilon}, \tag{1}$$

where B is a $n \times n$ matrix with the transformed vectors as its rows, Q, a $p \times n$ matrix, is the unknown transformation to be computed and $\hat{\epsilon}$ is the criterion to be minimized. In the case of a labelled training set, we identify the matrix B in equation (1) with the matrix whose rows and columns are indexed by the feature vectors and the $(i, j)^{\text{th}}$ entry is 0 if the ith and the jth feature vectors belong to the same class and 1 if they belong to different classes. In the context of face recognition, n and p are the number of input face images and the number of pixels respectively, and D is the $n \times p$ matrix whose rows are the input face images. The criterion in equation (1) can be minimized in many ways. Our contribution to the face recognition problem is to formulate it in a manner that allows the application of Procrustes analysis and the Moore-Penrose inverse to minimize the error in equation (1) in two different ways.

We rewrite equation (1) in the form

$$||B - DQ|| = \epsilon \tag{2}$$

where ϵ is the criterion function to be minimized. Given the matrices D and B, our objective is to find a transformation matrix Q, which minimizes the error ϵ in equation (2).

As pointed out by the referees, it is clear that our approach is not restricted to the face recognition problem even though it has been validated extensively for small datasets in this context. It was pointed out by one of the referees that our method resembles the neural network technique which was one of the methods surveyed in Zhang et al [8]. The approach surveyed in that paper is motivated by the different considerations, the model is different from ours, the techniques for solving the problem are different and most importantly, as the authors themselves pointed out, the results were not satisfactory (page 1433, [8]) and the recognition rates not as high as ours. The discrepancy between their results and ours will be explored in an article which is being prepared for a journal.

3   Procrustes Analysis Based Classifier

Procrustes analysis is a classical technique of finding a transformation Q *with orthonormal columns,* which minimizes the error ϵ in equation (2). The additional constraint of the orthonormal columns means that equation (2) has to be recast in the following form, where $|| \; ||_F$ denotes the Frobenius norm.

$$\text{minimize } ||B - DQ||_F \tag{3}$$

subject to $Q^T Q = I_n$. For any matrix $G \in \Re^{m \times n}$,

$$||G||_F = (tr(G^T G))^{1/2} \tag{4}$$

where $tr(G)$ denotes trace(G). Since Q has orthonormal columns, we have

$$||B - DQ||_F^2 = tr(B^T B) - 2 \; tr(Q^T D^T B) + tr(D^T D) \tag{5}$$

Given the matrices B and D, equation (5) is equivalent to the problem of maximizing $tr(Q^T D^T B)$. The Q that maximizes $tr(Q^T D^T B)$ can be found using the the SVD of $D^T B$. Let

$$D^T B = U \Sigma V^T \tag{6}$$

where Σ=diag($\sigma_1, \sigma_2, \ldots, \sigma_n$). We define an orthogonal matrix Z by Z=$V^T Q^T U$. Then

$$tr(Q^T D^T B) = tr(Q^T U \Sigma V^T) = tr(Z \Sigma) = \sum_{i=1}^{n} z_{ii} \sigma_i \; \leq \; \sum_{i=1}^{n} \sigma_i. \tag{7}$$

Since $Z = I_n$, the trace in equation(7) is maximized by setting $Q = UV^T$.

The algorithm for the training part of the Procrustes based classifier is formally in Algorithm 1. The testing is done by projecting the test vector into the subspace spanned by Q and identifying the class of the test vector with that of the nearest neighbour. This is given in Algorithm 2.

Algorithm 1 The Procrustes based classifier: The training algorithm

---
INPUT: A labelled dataset of n feature vectors, $\mathbf{x_i} \in \Re^p$ with c classes.
D: $n \times p$ matrix with the feature vectors $\mathbf{x_i}$ as its rows
B: $n \times n$ matrix whose $(i,j)^{\text{th}}$entry is 0, if $\mathbf{x_i}$ and $\mathbf{x_j}$ belong to the same class and 1, otherwise
OUTPUT:
Q: $p \times n$ orthogonal transformation matrix such that $||B - DQ||_F$ is minimum
DQ: $n \times n$ matrix with the transformed feature vectors $\mathbf{y_i}$ as its rows; *i.e., DQ* =$[\mathbf{y_1}, \mathbf{y_2}, \ldots, \mathbf{y_n}]$
Step 1: Let $C = D^T B$
Step 2: Compute the SVD of C , $C = U \Sigma V^T$ and save the matrices U and V
Step 3: The transformation matrix $Q = UV^T$

Algorithm 2 The Procrustes based classifier: The testing algorithm

---

INPUT: A test feature vector, $\mathbf{x} \in \Re^p$

Q: $p \times n$ orthogonal transformation matrix such that $||B - DQ||_F$ is minimum

DQ: $n \times n$ matrix with the transformed feature vectors $\mathbf{y_i}$ as its rows; *i.e., DQ* $=[\mathbf{y_1}, \mathbf{y_2}, \ldots, \mathbf{y_n}]$

OUTPUT: The feature vector $\mathbf{x_j}$ or the class containing $\mathbf{x_j}$

Step 1: $j = \overset{\text{argmin}}{\underset{i}{}} ||\mathbf{y_i} - \mathbf{x}Q||$, for all i

Step 2: $\mathbf{x_j}$ or the class containing $\mathbf{x_j}$

---

4   The Moore-Penrose Classifier

The objective of finding a transformation matrix Q in equation (2) can also be achieved using the generalized inverse (Moore-Penrose inverse) in the following manner. Let the singular value decomposition of D be $U \Sigma V^T$, where Σ is the $n \times p$ diagonal matrix with entries $\sigma_1, \ldots, \sigma_n$ which are the *singular values* of D. The generalized inverse of D is

$$D^\dagger = V \Sigma^\dagger U^T, \tag{8}$$

where $\Sigma^\dagger$ is the $p \times n$ diagonal matrix with entries $\frac{1}{\sigma_1}, \ldots, \frac{1}{\sigma_n}$. We premultiply equation (1) with $D^\dagger$ and get

$$D^\dagger B = Q + D^\dagger \hat{\epsilon} \tag{9}$$

When the error term is neglected, the transformation matrix becomes,

$$Q = D^\dagger B \tag{10}$$

This technique gives minimum error when the number of feature vectors in the training set is less than the dimension of the feature vectors, as in the case of most face data sets. We call the classifier based on the above transformation the Moore-Penrose classifier. The training and testing algorithms for the Moore-Penrose classifier are similar to those described in Algorithm 1 and Algorithm 2 which we omit due to lack of space.

5   Experiments with Face Datasets

The YALE face database, ORL face database, GIT Face database and the FERET face database were used to test the performance of the two classifiers proposed in this paper [6, 9–11]. In some of the experiments done using the proposed classifiers, the images were resized to 128×128. We experimented with our classifiers by varying the number of training samples per person and using the remaining images for testing. In the case of the GIT database, the eye coordinates were manually marked, rotated and scaled, so that all the eye coordinates

are aligned. From the FERET database, we manually selected 106 people with each having 4 frontal images.

Table 1 gives information regarding the databases used in our experiments. Table 3 contains the recognition rates of the classifiers for varying number of training samples. It is shown that the recognition rates improve with the increase in training samples. It is also seen that the Moore-Penrose classifier outperforms PCA, LDA, DCV as well as the Procrustes based classifier for the ORL, YALE and GIT databases.

Table 1. Face databases used in the experiments

Database	No. of people	No. images per person
YALE	15	11
ORL	40	10
GIT	50	15
FERET	106	4

The FERET database has images taken at different times and exhibits extreme variations in hairstyles and illumination. We experimented with this dataset by measuring and comparing the performance of our classifiers with that of PCA, LDA and DCV. We used varying training samples per person and also samples taken at the *same* and *different* sessions and compared the performance of our classifiers across these different training samples. It is shown that the two classifiers proposed in this paper outperform the baseline PCA, LDA and DCV algorithms [6, 7, 12]. These results are tabulated in Table 2.

Table 2. Recognition rates for FERET face database

Techniques	1	2 (same)	2 (different)	3
PCA	55.34	42.45	91.03	91.50
LDA	57.86	45.75	92.22	94.33
DCV	57.86	35.84	87.73	91.50
Procrustes	57.86	46.22	96.22	95.28
MP Classifier	61.94	57.54	96.22	94.33

Even though it is claimed that DCV performs better than LDA, our experiments indicated the contrary. It is evident that the Procrustes based classifier and the Moore-Penrose classifier outperformed the other three algorithms they were compared with.

The following section contains some interesting observations regarding the comparison of the distribution of the distances for the various classifiers considered in this paper.

Table 3. Recognition rates with varying number of training images for the ORL, YALE and GIT face databases

Dataset	techniques	1	2	3	4	5
ORL	PCA	48.33	68.43	78.92	90.41	93.5
	DCV	60.83	58.43	69.28	83.33	91.50
	LDA	60.83	77.81	86.78	93.75	96.00
	Procrustes	60.83	79.06	89.28	95.00	97.50
	MP classifier	62.77	80.00	91.07	95.41	**98.00**
YALE	PCA	46.66	82.22	84.16	91.42	91.11
	DCV	86.00	42.96	57.5	71.42	81.11
	LDA	86.00	91.85	95.00	96.19	95.55
	Procrustes	86.00	91.85	95.00	96.19	93.33
	MP classifier	90.00	95.55	95.83	**99.04**	98.88
GIT	PCA	34.85	45.69	51.50	54.90	66.00
	DCV	34.57	28.92	36.16	39.63	56.00
	LDA	34.57	44.92	47.83	50.18	66.66
	Procrustes	34.57	48.46	53.83	57.45	71.40
	MP classifier	43.00	60.00	69.33	74.90	**87.40**

6   A Technique for Comparing Classifiers

For the various classifiers considered in this paper, let $m(ps)$ be the mean distance for the passed images, $m(fl)$, the mean distance for the failed images, $std(ps)$, the standard deviation of distances for the passed images and $std(fl)$, the standard deviation of distances for the failed images. We define the pass and the fail threshold as in equations (11) and (12).

$$pstr = m(ps) + std(ps) \qquad (11)$$

$$fltr = m(fl) - std(fl) \qquad (12)$$

Using these parameters, we have devised a simple and effective method for comparing classifiers in the following fashion. Define

$$olp = \frac{pstr - fltr}{m(fl) - m(ps)}. \qquad (13)$$

This parameter measures how well the distances of the images that pass and the distances of the images that fail are separated. For a geometric explanation of these parameters see Figure 1. For a robust classifier, it is clear that the value of olp in equation (13) should be as low as possible. If the value of olp is negative then $pstr$ and $fltr$ in equation (11) and equation (12) are well separated. It should be emphasized that this method compares the classifiers based *only* on their discriminatory power and *not* on their recognition rates. These parameters are also important while selecting the threshold in the case of one-one matching, i.e., verification.
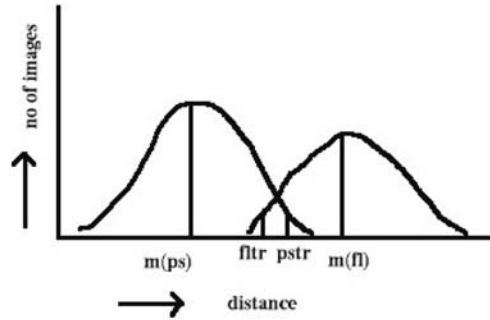
Fig. 1. Geometric explanation of the parameters using sample density plot of distances

In Table 4, it can be seen that our classifiers outperform the PCA, LDA and the DCV classifiers by having lower values of *olp* for *all* the databases considered in this paper.

7   Conclusion and Future Directions

We have proposed two new classifiers for the face recognition problem which match the best recognition rates that have been reported in the literature for the ORL, YALE, GIT and FERET face databases. We compared the performance

Table 4. Mean, standard deviation, threshold and overall performance of classifiers

Dataset	Techniques	m(ps)	std(ps)	m(fl)	std(fl)	pstr	fltr	olp
ORL	PCA	0.2027	0.0552	0.2919	0.0430	0.2578	0.2490	0.0996
	DCV	2662.1	903.78	3944.3	643.79	3565.9	3300.5	0.2070
	LDA	0.2223	0.0606	0.3240	0.0482	0.2830	0.2758	0.0703
	MP Classifier	1.2161	0.3319	1.9446	0.2370	1.5480	1.7076	-0.2191
	Procrustes	0.1321	0.0447	0.2485	0.0421	0.1768	0.2064	-0.2545
YALE	PCA	0.1637	0.0462	0.2529	0.0507	0.2099	0.2022	0.0863
	DCV	1834.7	622.38	2667.7	932.67	2457.1	1735	0.8667
	LDA	0.1845	0.0573	0.3173	0.0441	0.2419	0.2732	-0.2359
	MP Classifier	0.7627	0.3096	1.7041	0	1.0723	1.7041	-0.6711
	Procrustes	0.0978	0.0423	0.2132	0.0617	0.1401	0.1515	-0.0989
GIT	PCA	0.2585	0.0491	0.2919	0.0408	0.3076	0.2528	1.5667
	DCV	3717	872	3944.3	789.48	4589	3527.3	1.7715
	LDA	0.2831	0.0553	0.3240	0.0442	0.3384	0.2827	1.2692
	MP Classifier	1.6045	0.3013	1.9446	0.1352	1.9058	1.8181	0.2513
	Procrustes	0.1962	0.0487	0.2485	0.0394	0.2450	0.2040	0.8686
FERET	PCA	0.1923	0.0400	0.2245	0.0297	0.2324	0.1947	1.1706
	DCV	3506.9	671.1	4078	608.54	4178	3469.5	1.2406
	LDA	0.2147	0.0438	0.2495	0.0348	0.2585	0.2147	1.2599
	MP Classifier	1.2998	0.2052	1.4761	0.0923	1.5051	1.3837	0.6885
	Procrustes	0.1929	0.0431	0.2287	0.0337	0.2360	0.1950	1.1457

of these classifiers with the baseline PCA, LDA and the recently proposed DCV technique of Cevikalp et al. [7]. We have shown that our classifiers outperform them all in the databases considered so far. We also devised new parameters which help us compare classifiers based only on their discriminatory power and not just their recognition rates. The classifiers proposed in this paper outperform the PCA, LDA and DCV classifiers with respect to these parameters. An exhaustive comparison of various other classifiers using this formalism is also under way.

A detailed analysis of the proposed classifiers and methods of boosting the recognition rates for larger datasets are currently being investigated. We have also discovered many variants of our classifiers, which have higher recognition rates than the ones announced here.

References

1. Golub, G.H., Loan, C.F.V.: Matrix Computations. The Johns Hopkins University Press, Baltimore (1996)
2. Bellino, K.A.: Computational Algorithms for Face Alignment and Recognition. M.s thesis, Virginia Polytechnic Institute and State University (2002)
3. Everson, R.: (Orthogonal, but not orthonormal, procrustes problems)
4. Wee, W.G.: Generalized inverse approach to adaptive pattern classification. IEEE Transactions on Computers **C-17** (1968) 1157–1164
5. Wee, W.G.: Generalized inverse approach to clustering feature selection, and classification. IEEE Transactions on Information Theory **17** (1971) 262–269
6. 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 **19** (1997) 711–720
7. Cevikalp, H., Neamtu, M., Wilkes, M., Barkana, A.: Discriminative common vectors for face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **27** (2005) 4–13
8. Zhang, J., Yan, Y., Lades, M.: Face recognition: Eigenface, elastic matching, and neural nets. Proceedings of the IEEE **85** (1997) 1423–1435
9. ORL: (ORL face database)
10. GIT: (Face database of georgia institute of technology, read-me)
11. P. J. Phillips, H. Wechsler, J.H., Rauss, P.: The FERET database and evaluation procedure for face recognition algorithms. Image Vision Comput. **16** (1998) 295–306
12. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of cognitive Neuroscience **3** (1991) 72–86

Two Factor Face Authentication Scheme with Cancelable Feature

Jeonil Kang[1], DaeHun Nyang[1], and KyungHee Lee[2]

[1] Information Security Research Laboratory, INHA University[⋆]
dreamx@seclab.inha.ac.kr, nyang@inha.ac.kr
http://seclab.inha.ac.k
[2] Department of Electrical Engineering, University of Suwon
khlee@suwon.ac.kr

Abstract. Though authentication using biometric techniques has conveniences for people, security problems like the leakage of personal bio-information would be serious. Even if cancelable biometric is a good solution for the problems, only a few biometric authentication scheme with cancelable feature has been published. In this paper, we suggest a face authentication scheme with two security factors: password and face image. Using matching algorithm in the permuted domain, our scheme is designed to be cancelable in the sense that templates that is composed of permutation and weight vector can be changed freely.

1 Introduction

Many biometirc technologies have been studied to protect bio-information from stolen. Generally "matching" is process that it compares a template with acquired image, and thus, bio-information may be recoverable from templates. However, this property works as week point in security side.

Therefore, N.K. Ratha's "Cancelable Biometric"[1] concept which can cancel encoded information from database is needed. Database has information that is encoded as a template, and acquired information become encoded by the same function as database. These original and acquired information are compared in encoded domain, and the result is the same as the result of comparison in non-encoded domain. It is good for the encoding function to have a feature of non-invertible(or one-wayness). Since, however, generally this encoding processing is also based on image processing, it is too hard to apply fully random permutation in bit. Hence, it is still too hard to use cryptographic method. Mario Savvides's scheme is an example of cancelable biometric[8].

M. Braithwaite proposed an idea that matching is performed in transformed domain by some function, where they did not define what the functions are[2]. In this paper, we suggest an instance of their concept. We propose a cancelable biometric authentication scheme by combining PCA(Principal Component Analysis)[3]-[6] and PBKDF(Password-Based Key Derivation Function)

method[7]. Basically, this scheme works on local system, but we expect that
it can be applied to authentication through remote system.

2   Face Recognition Using Eigenfaces

In 1991, Mattew A. Turk published face recognition scheme using eigenface which
is based on PCA[3][4]. PCA is a way of identifying patterns in data, and ex-
pressing the data in such a way as to highlight its similarities and differences[6].
Eigenface is a face space that can be obtained by analyzing sample face images
with PCA method, and it contains features for identifying whether an acquired
image is a face or whos face is if the image is a face.

Let the training set of face images be $\Gamma_1, \Gamma_2, \cdots, \Gamma_M$. The average face of the
set is defined by $\Psi = \frac{1}{M} \sum_{n=1}^{M} \Gamma_n$ and different faces from the average face are
defined as $\Phi_i = \Gamma_i - \Psi$. And then we can get covariance matrix C about face
vector.

$$C = \frac{1}{M} \sum_{n=1}^{M} \Phi_n \Phi_n^T = AA^T \tag{1}$$

where the matrix $A = [\Phi_1 \Phi_2 \cdots \Phi_M]$. If size of face image is $N \times N$, the matrix
C is $N^2 \times N^2$. Since the N^2 eigenvectors and eigenvalues are computationally
infeasible and hard to handle, it has to modify the matrix C to $M \times M$ matric
using mathematical trick. M is much smaller than N^2. (You can see more detail
processes in [4].) For now, u_i denotes the eigenvectors for $i = 1, \cdots, M'$, where
M' is a value that is gotten for the face detection heuristically ($M' \leq M$), and
$U = [u_1 u_2 \cdots u_{M'}]$.

Each different face image Φ_k can be represented as a linear combination of
the eigenvectors u_i.

$$\Omega_k = U^T \Phi_k = U^T (\Gamma_k - \Psi) \tag{2}$$

where $k = 1, 2, \cdots, N_c$ and N_c is the number of face classes. Each normalized
training face Φ_k is represented by a vector $\Omega_k = [w_1^k, w_2^k, \cdots, w_{M'}^k]^T$.

Given an unknown face image Γ is normalized and projected on the eigen-
space.

$$\Omega = U^T (\Gamma - \Psi) \tag{3}$$

A vector $\Omega = [w_1 w_2 \cdots w_{M'}]^T$ means the contribution of each eigenface in rep-
resenting the input face image. The simplest method for determining which face
class provides the best description of an input face image is to find the face class k
that minimizes the Euclidean distance which is below some chosen threshold θ_ϵ.

$$\epsilon_k = \| \Omega - \Omega_k \| \tag{4}$$

If $\epsilon_k > \theta_\epsilon$, the face is classified as "unknown."

3   Principles of Macthing in Permuted Domain

In this section, we describe our face authentication scheme with cancelable feature. The cancelable feature is implemented by performing matching procedure in permuted domain. Biometrics information is stored at authentication system in permuted form and also the permutation itself is not stored anywhere. Thus, even though the user template is exposed to an attacker, it cannot be used for the attacker to be authenticated in other systems. Also, he cannot be authenticated even to the very system where the template has been stolen if the password that is used for generating the permutation is not exposed. Thus, we can say that our system is two factor authentication scheme using password and face image.

If the user want to cancel the biometrics information, he can do it by simply changing the password. More precisely, by changing password, user cannot change the biometrics information but he can chang the authentication information stored at the system. Even though the attacker gets the original face image, he cannot succeed in being authenticated in some other place unless he also can find out the password at each site.

3.1   Face Recognition in Permuted Domain

In the scheme using eigenface, average face Ψ, eigenvectors U and each weighted factor Ω_k can be represented in matrix form. They can be pre-computed and restored in a database for face recognition system. If the scheme using eigenface is used in an authentication system and Ω_k are exposed, someone might be able to be authenticated as specific authorized person by making ϵ to zero using exposed Ω_k. To do this, he must offer Γ_k to the system because the system computes Ω_k from Γ_k received.

However, if Ψ, U, and Ω_k are exposed in one scoop, an attacker can obtain face images from equation (2).

$$\Gamma_k = (UU^T)^{-1}U\Omega_k + \Psi \tag{5}$$

Thus, we need to store them in database only after they are securely encoded with some method such as cryptographic hash functions. However, cryptographic operation prevents face authentication algorithm from matching face exactly. To solve this problem, we need some transformation that does not effect matching procedure even in the transformed domain. In the authentication scheme using eigenface, permutation is a good choice.

Definition 1. *Lets $f_{c,k}(A)$ and $f_{r,k}(A)$ be some functions that permute some matrix A in the same sequence by column and by row respectively. Here, k means a user index.*

Let $P_{c,k}$ and $P_{r,k}$ be some $N \times N$ permutation matrices and A be $N \times 1$ matrix. And then, we can represent these functions by

$$f_{c,k}(A) = P_{c,k}A \tag{6}$$
$$f_{r,k}(A) = P_{r,k}A \tag{7}$$

Also, $P_{c,k}$ and $P_{r,k}$ have a relation of

$$P_{c,k}^T = P_{r,k} \tag{8}$$

where T denotes the transpose of a matrix.

Lemma 1. *If there is relationship between given face image Γ, average face image Ψ and eigenvectors U as (3), it satisfies*

$$U^T(\Gamma - \Psi) = f_{c,k}(U^T)\{f_{r,k}(\Gamma) - f_{r,k}(\Psi)\} \tag{9}$$

Proof. Let A and B be $N \times 1$ matrices, D be $M \times N$ matrix. And then, because of the characteristic of matrix arithmetic, we can say

$$f_{r,k}(A - B) = P_{r,k}(A - B) = P_{r,k}A - P_{r,k}B = f_{r,k}(A) - f_{r,k}(B) \tag{10}$$

Also, when we multiply D and A, D_{ij}(an element of D of i-th row and j-th column) is always multiplied by A_i(i-th element of A) because they have the same permutation sequence, even though D and A are permuted by $f_{c,k}()$ or $f_{r,k}()$. Thus,

$$DA = f_{c,k}(D)f_{r,k}(A) \tag{11}$$

From equations (10) and (11), we can obtain

$$D(A - B) = f_{c,k}(D)f_{r,k}(A - B) = f_{c,k}(D)\{f_{r,k}(A) - f_{r,k}(B)\} \tag{12}$$

$\square$

According to **Lemma 1**, now, the system can accept permuted Γ_k instead of non-permuted one to match face images. Note that because Ω_k has different dimension from other matrices (e.g. U has $N^2 \times M'$ and Ψ and Γ_k have $N^2 \times 1$, but Ω_k has $M' \times 1$), it cannot be permuted by $f_{c,k}()$ or $f_{r,k}()$.

If the system has been hacked and Ω_k has been exposed, someone may worry about security problems caused by his or her carelessness that Ω_k is not permuted and the possibility that different databases have the same Ω_k values for each person. However, since Ω_k is decided along face image Γ_k, eigenvectors U, and average face Ψ, if different systems have different U and Ψ, Ω_k of one system is different from that of other systems.

Mattew A. Turk said in his paper[3] "In many of our test cases, based on $M = 16$ face images. $M' = 7$ eigenfaces were used. The number of eigenfaces to be used is chosen heuristically based on the eigenvalues." If $M' = 7$ and Ω_k are permuted, all of the possible permutation of Ω_k are 5040(= 7!) cases, and there is no value for security. To increase security strength, thus, it has to use more face images for training and more eigenvalues about minimum 34 or 35. ($34! < 2^{128} < 35!$) From that reason, Ω_k is not permuted in this paper.

3.2   Derivation of Permutation Matrix from User Password

Because the system does not have any information about the permutation itself, that is, face image acquiring device must send permuted face image to the system. To make permutation on the face image, acquiring device gets a password from user and provides the permutation matrix.

To obtain secure permutation matrix, user password must be changed to a string with high entropy by cryptographic method like PBKDF(Password-Based Key Derivation Function) in PKCS#5 [7].

Definition 2. $D_K()$ *denotes password-based key derivation function, and this function returns a random key string.*

$$random\ key\ string = D_K(user\ password)$$

$D_K()$ is fully based on PBKDF in PKCS#5. Basically functions of PKCS#5 consist of hash functions or pseudo random generator functions, and it is infeasible to compute the inverse.

Definition 3. $D_{M_x}()$ *denotes permutation matrix derivation function from random key string, and this function returns a column-permuted matrix or a row-permuted matrix according to* x. x *can be* r *or* c.

$$permutation\ matrix = D_{M_x}(random\ key\ string)$$

A permutation matrix is generated the unit matrix I by permuting by column or row using to the random key string RKS. Let $RKS_{<i..j>}$ (described as RKS<i..j>) be a bit sequence (can be recognized as integer) from the i-th bit to the j-th bit of random key string. Then, permutation matrix derivation function $D_{M_x}()$ works as follows.

```
// RKS : random key string, input
// n : image height * image width
// m : larger integer then upper bound lg n

loop i := 0 to n
    j := RKS<i..i+m-1>
    if j >= n
        j := j % n
    if x = r
        swap(the i-th row of I, the j-th row of I)
    else if x = c
        swap(the i-th column of I, the j-th column of I)

return I
```

With $D_{M_x}()$, we can obtain a permutation matrix $P_{c,k}$ and $P_{r,k}$, and we can compute permutation function $f_{c,k}()$ or $f_{r,k}()$. We can rewrite (6) and (7) by

$$f_{x,k}(A) = D_{M_x}(D_K(user\ password))A \qquad (13)$$

Also, we can extract the *inverse permutation matrix* which can be computed from the original matrix using the same key, RKS easily.

4   Face Authentication Scheme with Cancelable Biometric

4.1   System Requirement

If the system stores permuted U^T and permuted Ψ only, eigenface-based match-ing algorithm works well with permuted acquired face images. In this case, how-ever, memory required to store user specific information is quite large. So, the system stores only one global U_{RR}^T and Ψ_{RR} that are permuted with some *master random key*, and then only user specific information required to store is just one permutation per user. By doing this, we can save lots of memory. Global vari-ables shown in below are secured by being permuted with permutation matrix from MRK(*Master Random Key*) that is never stored in the system and never to be obtained by any method.

Table 1. Global Variables and the explanation

Var	Explanation
U_{RR}^T	is a set of eigenvectors which is permuted by permutation matrix derived from MRK using $D_{P_c}()$.
Ψ_{RR}	is average face image which is permuted by permutation ma-trix derived from MRK using $D_{P_r}()$.

Each user account shown in below is consisted of user specific weight factor and permutation matrix.

Table 2. Account Variables and the explanation

Var	Explanation
Ω_k	is a user specific weight factor. It indicates how to reconstruct user's original face image using eigenvector U and average face Ψ.
$S_{x,k}$	means *User Specific Permutation* which can be used in face recognition process. It makes global variables to user specific variables when it is multiplied by global variables. Optionally, it can be compressed to small size because it has many zero elements. Also, $S_{c,k}^T = S_{r,k}$.

Consequently, we show some variation of previous equations here.

$$U_{RR}^T = D_{M_c}(MRK)U^T \tag{14}$$
$$\Psi_{RR} = D_{M_r}(MRK)\Psi \tag{15}$$

If matrix $f_{c,k}(U^T)$ and $f_{r,k}(\Psi)$ are user specific matrices permuted with permu-tation matrix from user passwords, then

$$f_{c,k}(U^T) = D_{M_c}(D_K(user\ password))U^T = S_{c,k}U_{RR}^T \tag{16}$$

$$f_{r,k}(\Psi) = D_{M_r}(D_K(user\ password))\Psi = S_{r,k}\Psi_{RR} \tag{17}$$

Thus, user specific data required to store are $S_{x,k}$ and Ω_k.

4.2 Face Recognition

When a user wants to be authenticated at the system that adopts our scheme, he should offer his face image and his password to "acquiring part." Let that face image be Γ, and then "face image permutation part" should compute permuted face image Γ_P using his password.

$$\Gamma_P = D_{M_r}(D_K(user\ password))\Gamma \tag{18}$$

After that, it is sent to "main process part" which tries to find user's weight factor Ω by searching in the user accounts. For each user account, main process part should compute candidates of Ω, Ω'_k.

$$\Omega'_k = f_{c,k}(U^T)\{\Gamma_P - f_{r,k}(\Psi)\} = S_{c,k}U_{RR}^T\{\Gamma_P - S_{r,k}\Psi_{RR}\} \tag{19}$$

And, it might be able to find a legal user account by comparing these Ω'_k vectors with user specific weight factor Ω_k.
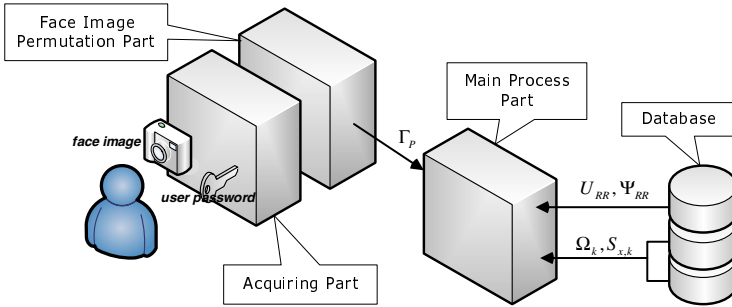
Fig. 1. Face Recognition Process. This diagram has been illustrated with the assumption that our scheme works in local system. At remote system, it will be separated acquiring and face image permutation part from other parts as client device

Since each account has different permutaion sequence, Euclidean distance between Ω'_k and Ω_k might have a large gap between the right account and wrong accounts.

4.3 Canceling Template

A template in our scheme is defined to be $(S_{x,k}, \Omega_k)$. Canceling template, thus, is to change $S_{x,k}$. The process to change password requires some security for preventing leakage of user face image during the the transmission, if the process is performed remotely. When user wants to change his password by security

risk, he must offer his password to the system to recompute $S_{x,k}$. However, it is too dangerous to offer in form of raw *user password*, $D_K(user\ password)$ or $D_{M_r}(D_K(user\ password))$ all. Because, $\Gamma = P_r^{-1} \cdot \Gamma_P$ and Γ_P can be obtained by observing authentication process. Thus, we assume that the password information is encrypted by session key safely.

Moreover, we must consider the case that a new password is adapted. If U^T and Ψ were recovered during this process, it is dangerous because U^T and Ψ might be able to be exposed to someone. Thus, the following pre-computed matrix should be offered for preventing the recovery.

$$P_{new}(P_{old})^{-1} = D_{M_r}(D_K(new\ password))D_{M_r}^{-1}(D_K(old\ password)) \quad (20)$$

$D_{M_x}^{-1}()$ denotes the inverse permutation matrix function that computes the inverse matrix from *random key string*. $P_{new}(P_{old})^{-1}$ cannot be separated and it changes $U_{P_{old}}^T$ to $U_{P_{new}}^T$ and $\Psi_{P_{old}}$ to $\Psi_{P_{new}}$ directly.

And then, the system must compute USP_k newly using U_{RR}^T and $U_{P_{new}}^T$, or Ψ_{RR} and $\Psi_{P_{new}}$. Because of (17),

$$S_{c,k} = U_{P_k}^T(U_{RR}^T)^{-1} = (\Psi_{P_k}(\Psi_{RR})^{-1})^T = S_{r,k}^T \quad (21)$$

5   Security Analysis

An attacker that does not know the right faces of the user who participates in a system might try to obtain user face image, or would like to obtain permitted authority to deal with the system.

Attack 1. *User face recovery from Γ_P:* An attacker might try to recover an user face image from a permuted face image Γ_P. However, this is infeasible because size of image is too large about $400(=20\times20)$ or $900(=30\times30)$. Trials complexity in such like this case can be about 400! or 900!. This make it too hard to get a good result in some months (or years).

Attack 2. *User face recovery from $U_{RR}^T \cdots$:* A hacker might be able to obtain U_{RR}^T, Ψ_{RR}, Ω_k, and $S_{x,k}$ by intrusion to the system. Using (5), he might try to recover user's face. However, $(U_{RR}U_{RR}^T)^{-1}U_{RR}\Omega_k+\Psi_{RR}$ is the same as permuted face image Γ_P.

Attack 3. *Raw U or Ψ recovery from $U_{RR}^T \cdots$:* If an attacker can recover a raw U or Ψ, he might be able to recover the user face images. Surely, they are not good images to see, because U does not contain whole information about training image set, but it has enough information to be authenticated. However, this is also infeasible because the size of the matrix is too large as like face image Γ_P.

Even though our scheme does not allow the leakage of face images, an attacker might be able to obtain face image of user who participates in system from other sources. Then, the attacker might try to do these attacks.

Attack 4. *Dictionary attack against user password:* Our scheme seems to be able to get user password by trying dictionary attack. However, it is too hard to attack in a local system: an attacker must enter the dictionary words one by one. Additionally, the system manufacturer might use some methods that can delay dictionary attack.

Attack 5. *Discovery of the password creates similar permutation matrix:* If an attacker knows similar permutation matrix in which a few rows or columns are different from original one, he might enter the system legally because of the characteristic of Euclidean distance. However, this is infeasible, even though it reduces the size of the searching space greatly, but it is almost 300-398! or 800-898!. Thus, he may expect to find out the password which creates similar permutation matrix in his dictionary. If he can find it, he can save much time to do dictionary attack. Unfortunately, he can find it in a probability of $|Dic| \times (N^2 - ds)!^{-1}$, where $|Dic|$ denotes the size of the dictionary and ds denotes the limit value of decision about "the similar". This probability is due to the difference of search spaces: dictionary space $(= |Dic|)$, random key string space $(= 2^{N^2 + log_2 N^2})$, and permutation space $(= N^2!)$.

Attack 6. *Discovery of other users' face images:* If an attacker succeeds in stealing the whole database from the authentication system, and by some means, attacker finds out a user's password, he can recover U and Ψ after computing inverse permutation from the password. Once he obtains U and Ψ, it is very easy task to compute all other users' original face images by evaluating equation (5). However, even if we assume this system cracking situation, attacker cannot use the original face image to be authenticated to other systems and even to the very system cracked unless he does know exactly the corresponding password at each system.

6   Conclusion

In this paper, we proposed a cancelable face authentication scheme based on eigenface methodology. Our scheme consists of two factors for strengthening the security weakness: comparison in the permuted domain and permutation sequence generated from user password.

After that, we thought and illustrated some attack methods that might be able to break our scheme, and we could not find any reasonable weak points from our scheme under the assumption that user face image can not be exposed by hacking the system. Even if a user face image was exposed from other sources, an attacker cannot attempt a successful attack that might collapse the system. Moreover, we suggested the method by which we can change user password safely.

In order to make the application with a good performance and strong security, more and deeper researches and verifications should be required, especially about U and Ψ as the result of training face images because U and Ψ have different

pattern in different training group(at the ages, countries, races, etc.) and it should have flexibility in the training steps.

References

1. N.K. Ratha, J.K. Connell, and R.M. Bolle: Enhancing security and privacy in biometrics-based authentication systems, IBM Systems Journal, vol. 40, no. 3, pp. 614-634, 2001
2. Michael Braithwaite, Ulf Cahn von Seelen, James Cambier, John Daugman, Randy Glass, Russ moore, and Ian Scott: Application-Specific Biometric Templates, Iridian Technologies Inc., Proceedings of AutoID, pp. 167 - 171, 2002
3. Mattew A. Turk and Alex P. Pentland: Face Recognition Using Eigenfaces, Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on 3-6 pp. 586 - 591, June 1991
4. Mattew A. Turk and Alex P. Pentland: Eigenfaces for Recognition, Journal of Cognitive Neuroscience, Vol. 3, No. 1, pp. 71-86, 1991.
5. Zhuji and Y.L. Yu: Face recognition with eigenfaces, Industrial Technology, 1994. Proceedings of the IEEE International Conference on 5-9, pp. 434 - 438, Dec. 1994
6. Lindsay I Smith: A tutorial on Principal Components Analysis, February 26, 2002
7. PKCS#5 v2.0: Password-Based Cryptography Standard, RSA Laboratories, March 25, 1999
8. Mario Savvides, B.V.K. Vijaya Kumar, and P.K. Khosla: Cancelable biometric filters for face recognition, Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on Vol.3, pp. 922- 925, Aug. 2004
9. Anil K. Jain, Umut Uludag, and Rein-Lien Hsu: Hiding a Face in a Fingerprint Image, Proc. of ICPR, Aug. 2002
10. http://www.biometrics.org, Biometric Consortium

Local Feature Extraction in Fingerprints by Complex Filtering

Hartwig Fronthaler, Klaus Kollreider, and Josef Bigun

Halmstad University, SE-30118, Sweden
{hartwig.fronthaler,klaus.kollreider,josef.bigun}@ide.hh.se

Abstract. A set of local feature descriptors for fingerprints is proposed. Minutia points are detected in a novel way by complex filtering of the structure tensor, not only revealing their position but also their direction. Parabolic and linear symmetry descriptions are used to model and extract local features including ridge orientation and reliability, which can be reused in several stages of fingerprint processing. Experimental results on the proposed technique are presented.

1 Introduction

The important features of a fingerprint are divided into global and local features. Examples of global features are, e.g. singular points, around which the ridges and valleys are wrapped, ridge orientation and frequency. Local features are minutia points, which are discontinuities in the ridge flow. The most prominent minutiae are ridge bifurcation and termination (=ending), which refer to points where a ridge divides into two and a ridge ends, respectively. The estimation of ridge orientation (and frequency) is done at a scale level comparable to when detecting minutiae. For this reason we can consider ridge orientation and the reliability of its estimation as a local feature, too. In this work we extract such local features, i.e. we propose a novel method to detect the minutiae's position and direction. A topical review of fingerprint processing technologies can be found in [1]. For a description of recent techniques we refer to [2]. Existing minutia extraction approaches comprise so-called "direct grey-scale" and "binarization-based" methods. In "binarization-based" methods [3], the fingerprint image is binarized and morphologically analysed for minutiae. In "direct grey-scale" methods mainly the image gradient and local grey-scale neighbourhood are used to locate minutiae, either by tracking ridges [4], or by classifying directional filter responses [5]. The direction of the minutiae is mostly inherited from the associated ridge. The presented work can be assigned to "direct grey-scale", as it operates exclusively on the local direction field of a fingerprint image. Previously, the image is enhanced by using a contextual filtering method [6].

Complex filtering in fingerprints has recently been shown to deliver both the position and direction of singular points (of type core and delta) in a single separable filtering step [7]. Having partial fingerprints lacking singular points, only local features can be used for alignment and matching purposes. In this study

we present a novel way to detect the minutia points' position and direction by use of filters sensitive to parabolic and linear symmetries. The targeted minutia types are ridge bifurcation and termination. In this work, the minutiae are used for alignment (registration) of two fingerprints only. This allows us to keep the required number of minutiae low, which is advantageous when facing partial or low quality fingerprints. The fingerprint registration using minutiae is followed by a matching step employing distinctive area correlation similar to [8], except for using our features. Linear and parabolic symmetry are the only features used throughout our work.

2   Local Feature Extraction

As mentioned in [9], bifurcation and termination minutiae show (parabolic) symmetry properties similar to those of the core type singular point - but at a different scale level - thus suggesting similar extraction techniques. Linear symmetry detection is also useful for locating minutiae in an inverse fashion, as "lack of linear symmetry" occurs at minutia points [10].

2.1   Symmetry Filters

We use two types of symmetries to model and extract the local structure in a fingerprint, which are parabolic and linear symmetry. Both symmetries can be derived by separable filtering of the orientation tensor. For a more detailed review of symmetry filters, i.e. symmetry derivatives of Gaussians, we refer to [11]. Our approach starts with the orientation tensor image, described in equation (1),

$$z = (f_x + if_y)^2 \tag{1}$$

where f_x and f_y denote the derivatives of the image in x and y direction respectively. We implement the tensor by convoluting the grey value image with separable Gaussians and their derivatives. Already the calculation of the tensor implies complex filtering since the outcome z is complex. A symmetry and the associated filter detecting it can be modelled by $\exp(im\phi)$, where m represents its order [11–13]. It was shown that an estimation for $m \geq 0$ can be done by using the following polynomial in conjunction with a Gaussian window function:

$$h_m = (x + iy)^m \cdot \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \tag{2}$$

First of all we are interested in parabolic symmetry of order $m = 1$, since the parabolic pattern is most similar to a minutia point in a fingerprint. Equation (3) describes how complex filtering is applied to detect this type of symmetry, where $\langle,\rangle$ represents the 2D scalar product and $g(x, y)$ denotes the Gaussian window.

$$\text{PS} = \langle z, h_1 \rangle = \left\langle (f_x + if_y)^2, (x + iy) \cdot g(x, y) \right\rangle \tag{3}$$

The two grey-scale images in figure 1 show minutia points of type ridge bifurcation (left) and ridge ending (right), with their direction indicated. Furthermore,
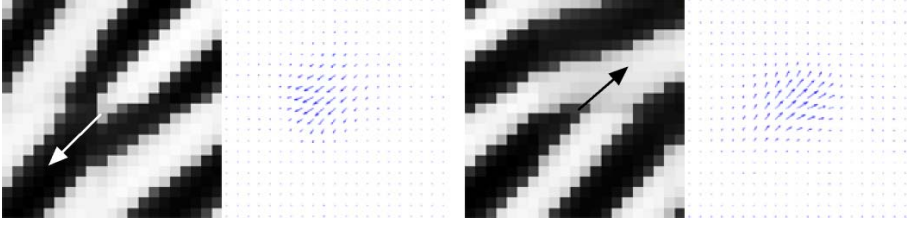
Fig. 1. Left: ridge bifurcation and filter response; Right: ridge ending and filter response

the corresponding complex filter responses of h_1 are displayed. The latter can be described as $c_1 = \mu \cdot \exp(i\alpha)$. The value μ is a certainty measure and the argument α represents the geometric orientation of the symmetric pattern in order $m = 1$. An important property of parabolic symmetry filtering for minutia detection is that the minutia direction is retrieved at the same time (see figure 1). We wish to stress that the estimated minutia direction is independent from the associated ridge.

The second type of symmetry used in this work is linear symmetry, which occurs at points of coherent ridge flow. The discrimination power of this information has been shown in fingerprint pattern matching [14]. In the case of linear symmetry, which can also be described as symmetry of order $m = 0$ the polynomial in equation (2) reduces to a Gaussian filter. Applying this filter to the orientation tensor $z(x, y)$ corresponds to summing up the local neighbourhood in each point. In order to have a reliable measure for the linear symmetry, we first calculate the second order complex moments $I_{20} = \langle z, h_0 \rangle$ and $I_{11} = \langle |z|, h_0 \rangle$, as suggested by[15, 16]. The measure for linear symmetry used in this work is denoted in equation (4).

$$\text{LS} = \left(\frac{I_{20}}{I_{11}} \right) = \frac{\langle z, h_0 \rangle}{\langle |z|, h_0 \rangle} \tag{4}$$

I_{11} acts as an upper boundary for the linear symmetry certainty, and by dividing I_{20} through I_{11} unreliable orientations are attenuated, whereas the strong ones are promoted.

2.2   Detection Process

After the image enhancement and the calculation of linear and parabolic symmetry, some more steps have to be considered in order to reliably detect minutia points. First, the selectivity of the parabolic symmetry filter responses is improved, using the following inhibition scheme [17]:

$$\text{PSi} = \text{PS} \cdot (1 - |\text{LS}|) \tag{5}$$

Essentially, in equation (5), the parabolic symmetry is attenuated if the linear symmetry is high, whereas it is preserved in the opposite case. In figure 2 the
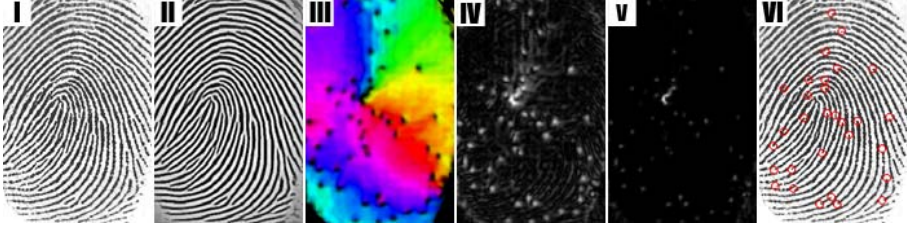
Fig. 2. Minutia point detection process

minutia detection process is visualized. The first two images visualize the initial fingerprint and its enhanced version respectively. The parabolic symmetry displayed in image IV (|PS|) is inhibited with the absolute value of the linear symmetry shown in image III (LS). The latter image also represents the local orientation. The resulting sharpened magnitudes |PSi| are displayed in image V. In a further step all filter responses below a certain threshold $\tau_{\mathrm{reliablePS}}$ are set to zero. LS provides a good measure for segmenting the fingerprint from the background in order to discard responses in the marginal area of the fingerprint image. The remaining filter answers, which are concentrated to pixel islands, are used for the extraction of minutia candidates. During this extraction process we search for the highest filter response in a small neighbourhood of 9x9 throughout the fingerprint area in order to avoid multiple detection of the same minutia. Finally we demand each minutia to be fully surrounded by high linear symmetry, by checking whether the average linear symmetry on a ring around a minutia candidate is above a threshold $\tau_{\mathrm{averageLS}}$. Thus we can exclude spurious minutia points which occur at the transition from the fingerprint to the background and at impurities within the fingerprint area. If this condition is fulfilled, the position and the complex filter response are stored. The final minutia list is ordered by magnitude. In image VI of figure 2, circles indicate detected minutiae when considering only the 30 highest magnitudes.

3   Pairing of Minutiae

In order to establish correspondences between two fingerprints, we implement a local minutiae matching approach inspired by triangular matching [18]. This essentially means establishing a connected series of triangles, which are equal with respect to both fingerprints and have corresponding minutiae as their corners. From the detection process we receive a minutiae list, containing the position and direction of a fingerprint's minutiae. For each minutia, additional attributes are derived from this list, which describe the relation to all other minutiae. For two arbitrary minutiae m_i and m_j contained in a list, the following attributes are stored:

1. the distance d_{ij} ($= d_{ji}$) between the two minutiae
2. the angles α_{ij} and α_{ji}, which describe the angle of the minutiae with respect to a line between each other. Note, that all angles are signed in order to be unambiguous.
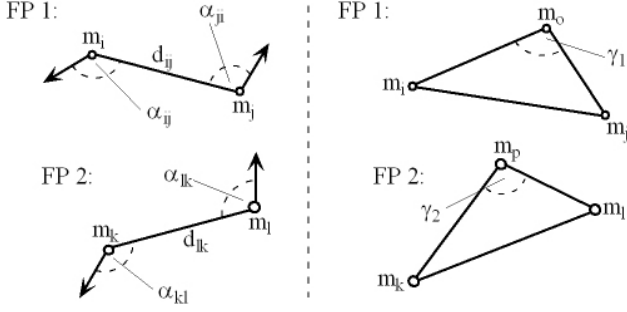
Fig. 3. Left: corresponding couples for 2 fingerprints; Right: triangles for 2 fingerprints

This is also visualized in the left part of figure 3. Having this extended minutiae list, it is possible to pre-select corresponding couples in two fingerprints, meaning two corresponding minutiae per fingerprint. Having two arbitrary minutiae m_k and m_l of a second minutiae list, the correspondence is fulfilled if $|d_{ij} - d_{kl}| < \lambda_{\text{dist}}(d_{ij})$ AND $(|\alpha_{ij} - \alpha_{kl}| + |\alpha_{ji} - \alpha_{lk}|) < \lambda_{\text{angle}}(d_{ij})$. If corresponding, the two couples are stored in a structure $C(n) = \{m_i, m_j; m_k, m_l\}$ with $n = 1..N$, where N is the number of totally found corresponding couples. The threshold functions λ_{dist} and λ_{angle} are designed to adapt to the distance between the tested minutiae. For a larger distance the angle tolerance can be decreased whereas the distance tolerance must be increased. The general use of threshold boxes, whether dynamical or not, is preferable since global distortions and inexact minutia detection can occur [2, 19].

Among all corresponding couples, which cannot be assumed to be truly corresponding in a global scope, we look for those which have a minutia in common in both of the fingerprints. Thus, taking $\{m_i, m_j; m_k, m_l\}$ as a reference, it may be that $\{m_i, m_o; m_k, m_p\}$ and $\{m_j, m_o; m_l, m_p\}$ are corresponding couples and therefore contained in C, too (i,j,o and k,l,p are minutia indices of two fingerprints respectively). This is also visualized to the right in figure 3. This constellation would imply that m_o and m_p constituted neighbours to the reference couples $\{m_i, m_j\}$ and $\{m_k, m_l\}$ respectively. To verify neighbours, we additionally demand the closing angles γ_1 and γ_2 (see figure) to be similar, which means $|\gamma_1 - \gamma_2| < \lambda_{\text{closingAngle}}(d_{io} + d_{oj})$. The latter threshold is decreasing with distance and is generally stricter than any other threshold. In this way neighbours are consecutively assigned to a reference corresponding couple, the equivalent of establishing equal triangles with respect to both fingerprints sharing a common side. Each corresponding couple is taken as a reference once. The corresponding couple, to which most neighbours can be found, is considered for further purposes. It would also be possible to cancel the search, as soon as a minimum number of neighbours was reached. Note, that the remaining triangle corners on the right hand side of figure 3 can immediately be considered and verified as neighbours to the opposite couples as well. At last the couples (one of each fingerprint) having the best correspondence are stored in a pairing list followed by their mated neighbours. This list simply links minutiae indices between the two fingerprints, e.g. $\{i, k\}$, $\{j, l\}$, $\{o, p\}$, etc.

4   Fingerprint Alignment and Matching

We assume the alignment to be a rigid transformation, since we only consider translation and rotation. The determination of the translation and rotation parameter Δx, Δy and $\Delta \phi$ is explained by means of an example using only four established minutia pairs between two fingerprints. From the paired minutia indices, we derive a list $\{A_1, A_2\}$, $\{B_1, B_2\}$, $\{C_1, C_2\}$ and $\{D_1, D_2\}$, linking the spatial coordinates of corresponding minutiae. Note, that the subscripts are used to distinguish between the two fingerprints, and each capital letter is enclosing 2D coordinates. The translation parameter is calculated by $[\Delta x \ \Delta y] = \overrightarrow{A_1 A_2}$ using the first minutia pair only. The rotation angle is determined by calculating the angle difference of two vectors between corresponding minutiae in both fingerprint parts. In order to increase the precision of the final rotation parameter $\Delta \phi$, several rotation angles are calculated and averaged. In our example $\Delta \phi$ is determined by $\Delta \phi = (\measuredangle(\overrightarrow{A_1 B_1}, \overrightarrow{A_2 B_2}) + \measuredangle(\overrightarrow{A_1 C_1}, \overrightarrow{A_2 C_2}) + \measuredangle(\overrightarrow{A_1 D_1}, \overrightarrow{A_2 D_2}))/3$. Furthermore we calculate the mean value m and the standard deviation σ and exclude all angles, which are not in $m \pm 0.5\sigma$ to minimize the error. It is worth mentioning that no further fine alignment is performed. Having the translation parameters and using A_2 as rotation centre, the actual alignment is straight forward [20].

After the registration of two fingerprints a simple matching using correlation can be done. Therefore we extract small areas from LS around the detected minutia points in the first fingerprint and calculate the normalised correlation with areas at the same positions in the second fingerprint. Only areas containing well-defined fingerprint regions, having an average linear symmetry higher than $\tau_{\text{averageLS}}$ are considered for matching. In order to determine the final matching score, we calculate the mean value of the single similarity scores.

5   Experiments

We present the combined performance of fingerprint alignment using our minutiae detection and pairing method, and fingerprint matching using feature correlation. For our experiments we employ the FVC2000 DB2 database (set A), which contains 800 fingerprints (100 persons à 8 fingerprints).

We carry out a fingerprint verification using the whole database. The number of impostor trials is 4950, whereas it is 2800 for the genuine trials. The maximum number of minutiae to detect in either of the fingerprints was set to 25, the size of one correlation area in the matching step was 51x51. The results are shown as a pair of FAR (False Acceptance Rate) and FRR (False Rejection Rate) curves in figure 4. The achieved EER (equal error rate) is 4.7%, also indicated by the curves' intersection point (y coordinate) in figure 4. Although not used for fingerprint matching purposes, the number of paired minutiae per trial were recorded. On the average, 11 ± 4 pairs were established in genuine trials, whereas 3 ± 1 pairs were found in the impostor trials.
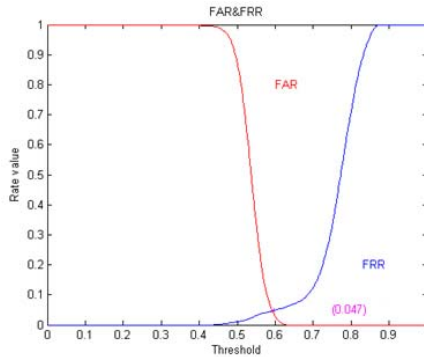
Fig. 4. FAR and FRR curves of the fingerprint verification test

6   Discussion and Conclusions

Symmetry descriptions to model and extract local features in fingerprints have been proposed in this work. Their application in several stages of a fingerprint recognition system has been shown to be efficient. In particular, the proposed minutiae detection method is novel, fast and direction aware. The minutia pairing is done intuitively and yields high accuracy enabling the fingerprint alignment. The average number of minutiae pairs established in the test trials supports the conclusion that the method can be included as an independent expert completing other experts, e.g experts using singular values, or experts matching directional maps, since efficient complementariness is increasingly being used as an argument when fusing experts and sensors in multimodal biometrics [21]. For this purpose, fingerprint distortions have to be studied further. The implemented alignment based on minutiae is not dependent either on any global features or on the size of available fingerprint area. The matching stage is able to discard unreliable fingerprint regions from the score decision. The overall performance with an achieved EER of 4.7% on a challenging database is encouraging.

Acknowledgement

References

1. Jain, A.K., Pankanti, S., Brabhakar, S., Ross, A.: Recent advances in fingerprint verification. In: Audio and Video-Based Biometric Person Authentication, Springer Ver. LNCS 2091; J. Bigun and F. Smeraldi, Eds. (2001) 182–191
2. Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: Handbook of fingerprint recognition. (2003) Includes DVD-ROM.
3. Jain, A., Hong, L., Bolle, R.: On-line fingerprint verification. IEEE-PAMI **19** (1997) 302–314

4. Maio, D., Maltoni, D.: Direct gray-scale minutiae detection in fingerprints. IEEE-PAMI **19** (1997) 27–40
5. Leung, M.T., Engeler, W., Frank, P.: Fingerprint image processing using neural networks. In: Computer and Communication Systems. (1990) 582–586
6. Chikkerur, S., Wu, C., Govindaraju, V.: A systematic approach for feature extraction in fingerprint images. In: International Conference on Bioinformatics and its Applications. (2004) 344–350
7. Nilsson, K., Bigun, J.: Localization of corresponding points in fingerprints by complex filtering. Pattern Recognition Letters **24** (2003) 2135–2144
8. Nandakumar, K., Jain, A.K.: Local correlation-based fingerprint matching. In: Proc. of Indian Conference on Computer Vision, Graphics and Image Processing. (2004) 503–508
9. Nilsson, K.: Symmetry Filters Applied to Fingerprints. PhD thesis, Chalmers University of Technology, Sweden, Chalmers SE-412 96 Göteborg, Sweden (2005) Dissertation No 2290, ISBN 91-7291-608-7.
10. Nilsson, K., Bigun, J.: Using linear symmetry features as a pre-procesing step for fingerprint images. In Bigun, J., Smeraldi, F., eds.: Audio and Video based Person Authentication - AVBPA 2001, Springer (2001) 247–252
11. Bigun, J., Bigun, T., Nilsson, K.: Recognition by symmetry derivatives and the generalized structure tensor. IEEE-PAMI **26** (2004) 1590–1605
12. Bigun, J.: Recognition of local symmetries in gray value images by harmonic functions. In: Ninth International Conference on Pattern Recognition, Rome, IEEE Computer Society Press (1988) 345–347
13. Knutsson, H., Hedlund, M., Granlund, G.H.: Apparatus for determining the degree of consistency of a feature in a region of an image that is divided into discrete picture elements. In: US. patent, 4.747.152. (1988)
14. Bigun, J., Fronthaler, H., Kollreider, K.: Assuring liveness in biometric identity authentication by real-time face tracking. In: CIHSPS2004 - IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety, Venice, Italy, IEEE Catalog No. 04EX815, ISBN 0-7803-8381-8 (2004) 104–112
15. Bigun, J., Granlund, G.H.: Optimal orientation detection of linear symmetry. In: First International Conference on Computer Vision, ICCV, June 8–11, London, IEEE Computer Society Press, Washington, DC. (1987) 433–438
16. Kass, M., Witkin, A.: Analyzing oriented patterns. Computer Vision, Graphics, and Image Processing **37** (1987) 362–385
17. Johansson, B.: Multiscale curvature detection in computer vision. Lic. thesis no. 877; LIU-TEK-LIC-2001:14, Linkoping University, Dep. EE (ISY), SE-581 83, Linkoping (2001)
18. Kovacs-Vajna, Z.: A fingerprint verification system based on triangular matching and dynamic time warping. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 1266–1276
19. Jea, T., Govindaraju, V.: A minutia-based partial fingerprint recognition system. The Journal of Pattern Recognition (2005)
20. Wilkinson, F., Wilson, H.R., Habak, C.: Alignment using distributions of local geometric properties. IEEE Trans. Pattern Analysis and Machine Intelligence **21** (1999) 1031–1043
21. Fierrez-Aguilar, J., Nanni, L., Ortega-Garcia, J., Cappelli, R., Maltoni, D.: Combining multiple matchers for fingerprint verification: a case study in fvc2004. In: Proc. of 13th IAPR Intl. Conf. on Image Analysis and Processing, Cagliari, Italy. Volume LNCS-3617., Springer (2005) 1035–1042

A TSVM-Based Minutiae Matching Approach for Fingerprint Verification

Jia Jia and Lianhong Cai

Key Laboratory of Pervasive Computing (Tsinghua University), Ministry of Education
Beijing 100084, P.R. China
`jiajia@mails.tsinghua.edu.cn, clh-dcs@tsinghua.edu.cn`

Abstract. This paper introduces Transductive Support Vector Machine (TSVM) into fingerprint verification. An improved fingerprint matching approach using TSVM is presented. In the proposed approach, the traditional minutiae-based fingerprint matching task is transformed to a classification task using TSVM. The paper presents an analysis of why TSVM are well suited for fingerprint matching, especially for small training sets. The approach is supported by experiments on five test collections, including both international and domestic fingerprint verification competition databases. Experimental results show that our approach is insensitive to noise as well as with effective performance.

1 Introduction

Fingerprint authentication is one of the most important biometric technologies [1]. Fingerprint is the pattern of ridges and valleys on the surface of fingers. As the fingerprint of a person is unique and immutable, automatic fingerprint verification system (AFVS) can be widely used in both anti-criminal and civilian applications. Therefore, performance improvements are the key points in current research on AFVS.

Operationally, AFVS can be decomposed into two steps: fingerprint preprocessing and fingerprint matching. Our main work focuses on the minutiae-based matching approach. Minutiae are the pattern of ridge bifurcations and endings. They are extracted from the thinned image which is obtained in the preprocessing step [2, 5, 6], as shown in Fig.1.

Fig. 1. Fingerprint minutiae. A ridge ending and a ridge bifurcation

Traditional minutiae-based matching algorithms (TMMA) use minutiae localizations (positions and orientations) [3], or combine them with texture information [4, 6] as the fingerprint features. After getting the feature vector which is the matching result of the query and template images' features, TMMA normally determine the final conclusion, fingerprints matched or not, by comparing a matching score with a *certain threshold* (Fig.2). The matching score is the dot product of the feature vector and a *certain weight vector*. The higher the matching score the better the match.
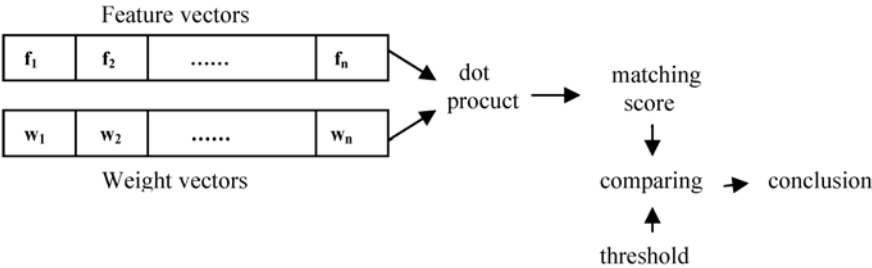
Fig. 2. The final conclusion is determined by comparing the matching score with a certain threshold. The matching score $= (f_1, f_2, \ldots\ldots f_n) \odot (w_1, w_2, \ldots\ldots w_n)$, where f_n is a feature dimension and w_n is the corresponding weight dimension

So our proposed approach is motivated by the following observations:

1) Due to the various skin and impression conditions, *the certain weight vector* may lead a false conclusion to the final result. For example, we define a weight vector $w = (0.5, 0.5)$ and a feature vector $f = (f_1, f_2)$ for a fingerprint database, where f_1 is the total number of matched minutiae and f_2 presents the similarity of two fingerprints' texture. In such case, if the sensors capture only a small area of the fingertip, which means a query and a template image have a small overlap of common region (Fig.3), the matching score $= 0.5f_1 + 0.5f_2$ may have a low value because of the low value of f_1. Finally the low matching score will lead to a "not matched" result although the two images have a high similarity of texture.

Fig. 3. Example of fingerprint images that having small overlap of common region but high similarity of texture. The two fingerprints is captured from the some finger

2) *The certain threshold* for result determination should be evaluated according to the different fingerprint images' quality for different databases. This course is time-consuming, and may be inaccurate to the final conclusion (Fig.4(a)).

In order to solve these two problems, the appropriate way is to avoid using certain weights and certain threshold, but divide the feature vectors standing for matched or not by a machine learning technique. Our key insight is that the matching problem can be regarded as a two-class classification problem (samples matched or not), so some classification methods, such as the multidimensional classifier using a hyperplane, can be adopted. That means we transform the problem which is hard to be solved in one-dimension space to a multidimensional space. This can be explained clearly by Fig.4(b).
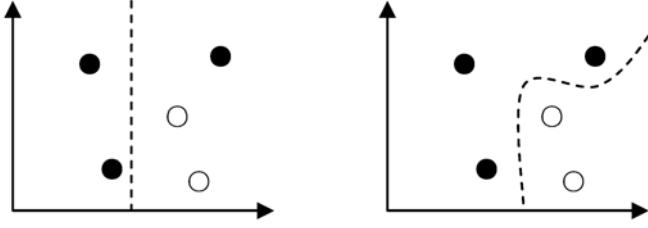
Fig. 4. White circles standing for the positive samples and the black ones standing for the negatives. (a) For the positive and negative samples, a certain one-dimensional threshold can not divide them exactly; (b) For the positive and negative samples, their projection to either axis has superposition, but in two-dimensional space, they are linear-dividable

Therefore, we propose a new fingerprint matching approach that transforming the matching problem to a classification problem and choosing a classifier TSVM [10] to solve it. The approach includes the following steps: 1) extracting and purifying the minutiae localizations to get the real minutiae information; 2) using minutiae localizations for the global matching to get the total number of matched minutiae; 3) defining a matching vector standing for the similarity of two fingerprints according to the results of 2); 4) using TSVM to label the matching vector with conclusion "matched or not". The performance of our approach is illustrated over both international and domestic fingerprint verification competition datasets, and experimental results show substantial improvements in the performance of fingerprint verification.

The rest of this paper is organized as follows. In section 2, we describe the minutiae extraction and purification scheme. In section 3, our matching approach based on TSVM is presented in detail, and we also contrast TSVM with general SVM to give the reasons of choosing TSVM as the classifier. Before concluded in section 5, experimental results and discussions are given in section 4.

2   Minutiae Extraction and Purification

Before matching step, we first present a minutiae extraction and purification scheme which is based on binary and thinned fingerprint images to obtain real minutiae [2].

For each minutia, we define a 4-tuple ($x, y, type, theta$) to describe its features. The x and y indicate the minutia's coordinate. The value of *type*, 1 or 2, indicates that the minutia is a ridge ending or a bifurcation. The *theta* indicates the minutia's orientation. Here we use an important concept, used in Xiao and Raafat [11], which will help us find the endings and bifurcations in the thinned image is the *condition number* (C_N).

Definition 1. The condition number of a given foreground pixel is:

$$C_N = \sum_{k=1}^{8} \left| \Gamma(k+1) - \Gamma(k) \right| \text{ where } \Gamma(9) = \Gamma(1) \qquad \Gamma(p) = \begin{cases} 1, \text{if pixel } p \text{ is foregroud;} \\ 0, \text{otherwise;} \end{cases} \qquad (1)$$

the index k scans the eight neighbors of the pixel in clockwise direction.

If C_N is 2, the pixel is an ending and, if C_N is 6, the pixel is a bifurcation. All other values of C_N are ignored.

Definition 2. A minutia's orientation *theta* is the local ridge orientation of the associated ridge.

Divide the fingerprint image into blocks of size $W \times W$. Compute the gradients G_x and G_y at each pixel in each block. Then the local orientation of each block is [5]

$$\theta(i,j) = \frac{1}{2} \tan^{-1} \frac{\sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} 2 G_x(u,v) G_y(u,v)}{\sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} (G_x^2(u,v) - G_y^2(u,v))} \quad (2)$$

Due to variations in skin conditions like postnatal marks or occupational marks, and impression conditions, a significant percentage of fingerprint images are in poor quality, in which ridge structures are corrupted. This leads to the problem that a large amount of false minutiae may be introduced. So it's clear that a post-processing stage is necessary to purify minutiae before matching.

The false minutiae include small holes, breaks, bridges, and spurs and so on. Here we choose a minutiae purify algorithm using local ridge structure [12]. The purification course is shown in Fig 5.
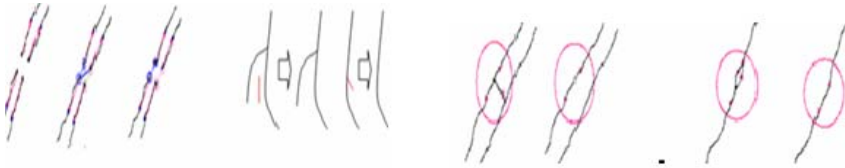
Fig. 5. Experimental results of minutiae purification: breaks, spurs, bridges and holes

3   A Matching Approach Based on TSVM

3.1   SVM VS. TSVM [13]

Support Vector Machine [7, 8] is a powerful classification method. It is based on a solid theoretical foundation – *structural risk minimization* [9]. In its simplest linear form, an SVM is a hyperplane that separates the positive and negative training examples with maximum margin, as shown in Fig 6. Large margin between positive and negative examples has been proven to lead to good generalization [9].
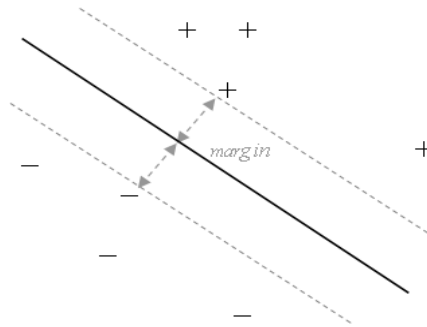
Fig. 6. An SVM/TSVM is a hyperplane that separates the positive and negative examples with maximum margin. The examples closest to the hyperplane are called support vectors

Transductive SVM (TSVM) introduced by Joachims [10] extends SVM to transductive learning setting. A TSVM is essentially a hyperplane that separates the positive and negative training examples with maximum margin on both training and test examples.

A regular SVM tries to induce a general classifying function which has high accuracy on the whole distribution of examples. However, this so-called inductive learning setting is often unnecessarily complex [13]. In fact, we do not care about the general classifying function, but rather attempt to achieve good classification performance on set of test examples. This is exactly the goal of transductive learning [9].

Why can TSVM be better than SVM or other classifier for fingerprint verification? There usually exists a clustering structure of training and test examples in fingerprint datasets: the examples in same class, which means from the same finger, tend to be close to each other in feature space. As explained in [10], it is this clustering structure of examples that TSVM exploits as prior knowledge to boost classification performance. This is especially beneficial when the number of training examples is small, like for fingerprint verification competition training case.

3.2   Global Matching

Global matching is the step to compute total number of matched minutiae between the test image and template image.

To ensure an overlap of common regions, the two images need to be aligned first. This is done by determining the transformation parameters (t_x, t_y, ρ, θ), where t_x, t_y indicate the adjustable distances in x-axis and y-axis, ρ indicates the flex coefficient and θ indicates the rotated angle. (t_x, t_y, ρ, θ) is computed by coordinates of two pairs of minutiae (usually delta points and core points) from both images [3]. These two pairs of minutiae are called reference points.

Once the transformation parameters (t_x, t_y, ρ, θ) are obtained, the test image can be aligned. Let (x, y) represent the original coordinate, then the aligned coordinate (u, v), is obtained as Eq.3.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \rho \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \tag{3}$$

Due to the structures of fingerprints themselves and the conditions of impression, some fingerprint images do not have delta points or center points [5]. In such cases, every two pairs of minutiae from test image and template image, if the minutiae of pair having the same value of *type*, should be chosen to be as reference points, and get the corresponding transformation parameters (t_x, t_y, ρ, θ). For different reference points, there will be different numbers of matched minutiae, and the maximum number indicates the similarity of the two images.

The pseudo code of our global matching scheme is given below:

3.3   Getting the Matching Vector

We define a matching vector $V(n, m, t, err)$ to describe the similarity of I_{test} and I_{temp}. This vector V is obtained by the next two steps:

```
Algorithm  global matching
Objective:  compute the total number of matched minut-iae
between the test image and template image
Input:
   I_test :  the test image.
   I_temp :  the template image.
   {p_n}:  the minutiae sequence of I_test.
   {q_m}:  the minutiae sequence of I_temp.
Output:
   the maximum number of matched minutiae pairs: t.
Method:
1.   For p_i, q_j (1 ≤ i ≤ n, 1 ≤ j ≤ m)

          if  type [p_i] = type [q_j]    add (p_i, q_j) to set A

Forall  (p_i1, q_j1), (p_i2, q_j2) ∈ A,  do Step2 to  Step5

2.   compute (t_x, t_y, ρ, θ) by Eq. (1)

3.   select (t_x, t_y, ρ, θ)

          if |ρ − 1| > 0.1 or θ > π/3  skip Step4 and  Step5

//This selection ensures that the flex coefficient ρ ap-

proximates to 1 and the rotation angle θ is less than 60°

4.   For {p_n}

          compute (x_i, y_i, theta_i)_new by Eq. (1) to  gain {p_n}_new

5.   match {p_n}_new with {q_m}. Record the number of matched minu-

     tiae pairs. Go Step2

6.   return t and the corresponding (t'_x, t'_y, ρ', θ')
```

Fig. 7. Pseudo code for global matching scheme

Step 1: Record the minutiae number n of I_{test} and m of I_{temp}, and maximum matched minutiae number t.

Step 2: Get a 100pixel×100pixel sub-image I_{sub} from the center of the threshold image of I_{test}. Translate I_{sub} by parameters $(t'_x, t'_y, ρ', θ')$ to I_{sub}', and let *err* represent the number of pixels in I_{sub}' that have the same intensity as its corresponding pixel in I_{temp}.

The *err* describes an estimation of the matching error.

3.4   Determining the Results by Matching Vector and SVM/TSVM

For a matching vector $V(n, m, t, err)$, we need to label it with "matching success" or "matching failure". The decision function of an SVM/TSVM is shown in Eq.4.

$$f(V) = \langle w \bullet V \rangle + b \tag{4}$$

$\langle w \bullet V \rangle$ is the dot product between w (the normal vector to the hyperplane) and V (the matching vector). The margin for an input vector V_i is $y_i f(V_i)$ where $y_i \in \{-1,1\}$ is the correct class label for V_i. Seeking the maximum margin can be expressed as minimizing $\langle w \bullet w \rangle$ subject to $y_i (\langle w \bullet V_i \rangle + b) \geq 1, \forall i$. We allow but penalize the examples falling to the wrong side of the hyperplane.

4 Experiments and Discussion

We conducted experiments with data of fingerprint verification competitions, to demonstrate the advantages of our proposed approach to fingerprint verification.

4.1 Datasets

In order to prove the influence of different image qualities and image amount to our matching approach, we have collected 5 datasets from FVC2002[1] (The Second International Fingerprint Verification Competition) and BVC2004[2] (Biometrics Verification Competition 2004). The information of each dataset is shown in Table 1.

Each fingerprint image allows a rotation angle that belongs to $[-\pi/4, \pi/4]$ (compared with the vertical line). Every two images from one finger have an overlap of common region. But there may be no delta points or core points in some fingerprint images.

Table 1. The information of datasets

	The source of the datasets	different fingers / total images	Sensors	Image size	Resolution
1st db	BVC2004 DB1	40/400	Optical sensor	412 x 362	500 dpi
2nd db	BVC2004 DB2	40/400	CMOS sensor	256 x 300	500 dpi
3rd db	BVC2004 DB3	40/400	Thermal sweeping sensor	300 x 480	500 dpi
4th db	BVC2004 DB4	40/400	Fingerpass	380 x 460	500 dpi
5th db	FVC2002 DB1	230/1840	Optical sensor	388 x 374	500 dpi

4.2 Experiments Setup

We posed 2 experiments. For each experiment, we compared our approach using TSVM with SVM and also with TMMA which uses threshold for result determination [5]. All approaches use the same image preprocessing algorithms [2]. Both the experiments are done by the method of 5-folder cross validation, but have differences in the size of test sets and training sets.

Experiment 1. For db1 to db4, 400 images are divided into 5 parts, each of which has 80 images. Both the algorithms run five times. For each time, **four** of the five parts are used as training sets (our approach only)**,** and the other **one** part is used as test set. The averaged verification result will be reported over these 5 times.

---

[1] http://bias.csr.unibo.it/fvc2002/

[2] http://www.sinobiometrics.com/chinese/conferences/sinobiometrics%2704.htm

Experiment 2. For db5, 1840 images are divided into 5 parts, each of which has 368 images. Both the algorithms run five times. For each time, **one** of the five parts is used as training set (our approach only), and the other **four** parts are used as test sets. The averaged verification result will be reported over these 5 times.

We use SVMlight[3] for the implementation of SVM/TSVM [8, 10], and take linear kernel in experiments.

4.3   Measures

The performance of a fingerprint verification algorithm can be measured by FNMR (False Non Match Rate: each sample in the subset A is matched against the remaining samples of the same finger), FMR (False Match Rate: the first sample of each finger in the subset A is matched against the first sample of the remaining fingers in A) and the average time of matching two images. Especially for Experiment 2, we also measure the maximum memory size of our approach. The configuration of running computer is PIV1.0G, 256M DDR.

4.4   Results and Discussions

The experimental results of db1 to db4 are shown in Table 2 and the results of db5 are shown in Table 3. We see that our approach using SVM/TSVM really can achieve much better accuracy than TMMA for fingerprint verification, and the average time the maximum memory for matching a pair of fingerprints is also acceptable.

Table 2. Experimental results of db1 to db4

	Avg match time (TSVM)	FMR	FNMR		
			TMMA	SVM	TSVM
1st database	0.901 sec	~ 0	6.67%	2.03%	1.94%
2nd database	0.524 sec	~ 0	30.28%	12.78%	9.31%
3rd database	1.004 sec	~ 0	24.17%	7.78%	7.50%
4th database	0.933 sec	~ 0	13.05%	5.28%	5.14%

Table 3. Experimental results of db5

FNMR			FMR	Avg match time (TSVM)	Max match Mem (TSVM)
TMMA	SVM	TSVM			
13.84%	4.08%	4.03%	~ 0.1%	0.459sec	8884 Kbytes

As shown in Table 1, fingerprints of these five datasets are captured by sensors of different types. So the images have different qualities. This strongly suggests that our feature extraction and purification scheme and SVM/TSVM methods capture well the information needed for fingerprint verification, and have a low influence by fingerprint image quality.

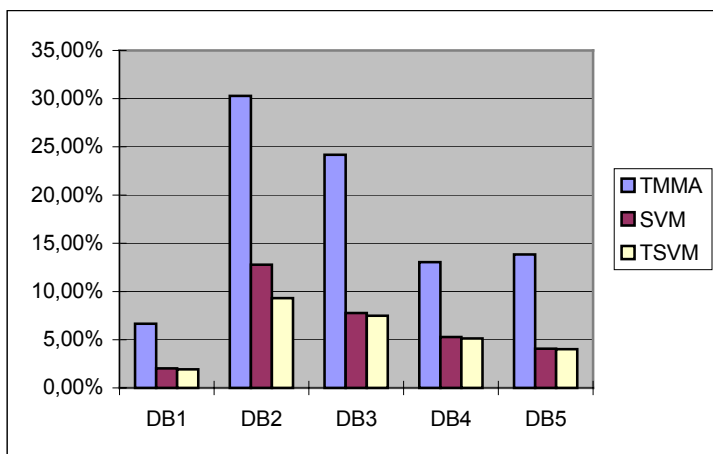---

[3] http://svmlight.joachims.org/

Fig. 8. Comparing the FNMR of TMMA with our approach using SVM and TSVM in five different datasets

We see that although the proportion of training sets is reduced, and the number of test members is increased in db5, our approach using SVM/TSVM still works better than TMMA. This implies that these machine learning methods have a low influence by fingerprint image amount.

Comparing the experimental results of SVM and TSVM, it turns out that the transductive learning technique can provide some help to fingerprint matching. We think this is because TSVM makes effective use of the matching vectors to enhance classification.

The experimental results of our approach and TMMA are compared in Figure 5. It is clear that our matching approach outperforms TMMA consistently and significantly.

5   Conclusion

Our main contributions to fingerprint verification are: 1) introduces SVM /TSVM for fingerprint verification that transforming the traditional minutiae-based matching task to a classification task, and using a powerful classifier to solve it; 2) proposing a corresponding definition of matching vectors which capture well the information needed for SVM/TSVM classifier; 3)proposing a corresponding minutiae extraction and purification scheme to getting minutiae information for fingerprint images of poor quality.

Future work may include: looking for a more accurate fingerprint preprocessing a method to accelerate the proposed TSVM approach, and so forth.

Acknowledgment

I would like to thank reviewers for their kindly review and helpful comments for this paper. ☺

References

1. Newham, E.: The Biometric Report. SJB Services. New York (1995)
2. Lin, H., Wang, Y.F., Anil, J.: Fingerprint Image Enhancement: Algorithm and Performance Evaluation. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No.8 (1998)
3. Wang, C.W., Li J.W.: Fingerprint Identification Using Point Pattern Matching. Journal of Chongqing University (Natural Science Edition), Vol. 25, No.6 (2002)
4. Ross, A., Anil, J.: A Hybrid Fingerprint Matcher. Pattern Recognition, 36 (2003) 1661-1673
5. Anil, J., Lin, H.: On-line fingerprint verification. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No.4 (1997) 302-314
6. Anil, J., Ross A., Prabhakar S.: Fingerprint Matching using Minutiae and Texture Features. In: International Conference on Image Processing (ICIP). Thessaloniki, Greece (2001) 282-285
7. Cristianini, N., Shawe-Taylor J.: An Introduction to Support Vector Machines. Cambridge University Press. Cambridge, UK (2000)
8. Yuan, Y., Paolo, F., Massimiliano P.: Fingerprint Classification with Combinations of Support Vector Machines. Lecture Notes in Computer Science, Vol. 2091. Springer-Verlag, Berlin Heidelberg New York (2001) 253–258
9. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer-Verlag, Berlin Heidelberg New York (2000)
10. Joachims, T.: Transductive Inference for Text Classification using Support Vector Machines. In: Proceedings of the 16th International Conference on Machine Learning (ICML). Bled, Slovenia (1999) 200-209
11. Xiao, Q., Raafat, H.:Pattern Recognition, 24,10 (1991) 985-992
12. Kasaei.S.,Deriche.,M.: Fingerprint Feature Extraction Using Block-Direction on Reconstructed Images. Journal of Speech and Image Technologies for Computing and Telecommunications, IEEE Tencon (1997)
13. Zhang, D., Lee, W.S.: Web Taxonomy Integration using Support Vector Machines. In: Proceedings of WWW 2004, New York, USA. ACM 1-58113-844-X/04/0005

A Robust Orientation Estimation Algorithm for Low Quality Fingerprints

Xinjian Chen, Jie Tian*, Yangyang Zhang, and Xin Yang

Center for Biometrics and Security Research, Key Laboratory of Complex Systems
and Intelligence Science, Institute of Automation, Chinese Academy of Sciences,
Graduate School of the Chinese Academy of Science, P.O.Box 2728, Beijing 100080 China
`tian@doctor.com`, `jie.tian@mail.ia.ac.cn`
`http://www.fingerpass.net`

Abstract. It is a difficult and challenge task to extract the accurate orientation field for the low quality fingerprints. This paper proposed a robust orientation estimation with feedback algorithm to get the accurate fingerprint orientation. First, the fingerprint image is segmented into recoverable and unrecoverable regions. The following orientation field estimation and orientation correction algorithms were only processed in the recoverable regions. Second, the coarse orientation image is estimated from the input fingerprint image using the gradient-based approach. Then we computed the reliability of orientation from the gradient image. If the reliability of the estimated orientation is less than pre-specified threshold, the orientation will be corrected by the proposed mixed orientation model. The proposed algorithm has been evaluated on the databases of FVC2004. Experimental results confirm that the proposed algorithm is a reliable and effective algorithm for the extraction orientation field of the low quality fingerprints.

1   Introduction

The orientation field, as a global essential feature of a fingerprint image, it summarizes the overall pattern of a fingerprint's ridges and valleys and plays a very important role in fingerprint recognition processing [1]. Many orientation detecting techniques have been developed to obtain fast and reliable orientation field. N. Ratha et al[2] proposed a gradient based algorithm to compute the orientation field. They estimated the local orientation at each pixel based on an analysis of grayscale gradients in the block centered at the given pixel. Sarat C. Dass[3] utilized a Bayesian formulation for reliable and robust extraction of the directional field in fingerprint images with a class of spatially smooth priors. The spatial smoothness allows for robust directional field estimation in the presence of moderate noise levels. A.M. Bazen and S.H. Gerez[4] proposed an algorithm to estimate a high resolution directional field of

---

fingerprints. The method computes both the direction in any pixel location and its coherence. It provides exactly the same results as the "averaged square-gradient method" that is known from literature.

Another type of algorithms has been advanced to model the overall orientation pattern globally, which are more adaptive to noise and corrupt in fingerprint images. Sherlock and Monro [5] represented the orientation based on a zero-pole model by mapping the image plane to a complex plane, where the core point was considered as pole and delta point as zero. These poles and zeros completely determine the overall fingerprint orientation. Li et al. [1] adapted the polynomial mode to refine the fingerprint orientation predicted by the coarse model. Zhou et al. [6] combined a polynomial model which is represented to reconstruct the overall orientation and a point charge model which modified the local orientation near the singular points, so that a more accurate orientation field could be obtained. Pedro et al. [7] divided the fingerprint image into several parts in terms of singular points. For different parts, a piecewise linear function was analyzed and applied.

The algorithms mentioned above could detect the orientation field successfully for the good and moderate quality fingerprint. However, the performance decreases greatly with the degradation of the fingerprint quality. This paper proposes a robust orientation estimation with feedback algorithm to get the accurate fingerprint orientation. First, the fingerprint image is segmented into recoverable or unrecoverable regions. The following orientation field estimation and orientation correction algorithm was only implemented in the recoverable regions. Second, the coarse orientation image is estimated from the input fingerprint image using the gradient-based approach. Then we computed the reliability of orientation from the gradient image. If the reliability of the estimated orientation is less than pre-specified threshold, the orientation will be corrected by the proposed mixed orientation model.

This paper is organized as following. Section 2 indicates out the details of orientation estimation of fingerprint images. Section 3 shows the performance of the proposed algorithm by experiments. Section 4 gives out our conclusion.

2   Fingerprint Orientation Estimation Algorithm

We proposed an orientation estimation with feedback algorithm to get the accurate fingerprint orientation. The flowchart of the proposed orientation estimation algorithm is shown in figure 1.

2.1   Segmentation of Fingerprints

In our algorithm, we classify each pixel in an input fingerprint image into a recoverable or an unrecoverable region. The fingerprint is divided into blocks of $w \times w$ pixels (w=12 in our algorithm). We select three features that contain useful information for segmentation. These three features are block clusters degree, block mean information and block variance. An optimal linear classifier has been trained for the classification per block and the criteria of minimal number of misclassified samples are used. Morphology has been applied as post-processing to reduce the number of classification errors [9].
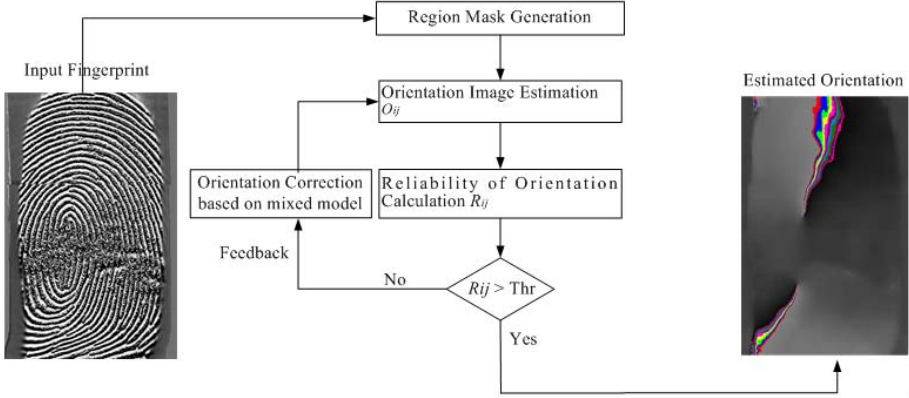
Fig. 1. The flowchart of the proposed fingerprint orientation estimation algorithm

2.1.1 Three Block Features

(1) The block clusters degree CluD

The block clusters degree CluD measures how well the ridge pixels are clustering. Using I as the intensity of image, the block clusters degree is defined as:

$$CluD = \sum_{i,j \in Block} sign(I_{ij}, Img_{mean}) \bullet sign(D_{ij}, Thre_{CluD}) \tag{1}$$

where $D_{ij} = \sum_{m=i-2}^{i+2} \sum_{n=j-2}^{j+2} sign(I_{mn}, Img_{mean})$, $sign(\alpha, \beta) = \begin{cases} 1 & if\,(\alpha < \beta) \\ 0 & otherwise \end{cases}$,

Img_{mean} is the intensity mean of the whole image. $Thre_{CluD}$ is an empirical parameter, $Thre_{CluD}$ =15 in our algorithm.

(2) The block mean information $MeanI$

We use the difference of local block mean and global image mean as the second feature for fingerprints segmentation. The mean information $MeanI$ for each block is given by:

$$MeanI = (\frac{1}{w \bullet w} \sum_{Block} I) - Img_{mean} \tag{2}$$

(3) The block variance Var

The block variance Var is the third feature that is used. In general, the variance of the ridge-valley structures in the foreground is higher than the variance of the noise in the background. The block variance Var for each block is given by:

$$Var = \frac{1}{w \bullet w} \sum_{Block} (I - mean)^2 \tag{3}$$

2.1.2 Linear Classification Design

In this paper, we follow a supervised approach since the block features of samples in both areas are available. Using this method, a classification algorithm can be con-

structed that minimizes the probability of misclassifying. In our algorithm we use the criteria of minimal number of misclassified samples to classify the blocks. The criteria function can be defined as:

$$J(w) = \left\| (Xw - b) - |Xw - b| \right\|^2 \tag{4}$$

Our aim is to find a vector w to make the value of J(w) minimal. The detailed steps of algorithm can be seen from our previous work [9].

2.2   Coarse Orientation Field Estimation Based on Gradient

In our algorithm, the gradient-based approach proposed by Lin et al [8] was used to compute the coarse orientation Oij for each location (i,j). However, their block-wise scheme is coarse and cannot obtain fine orientation field. In order to estimate the orientation field more accurately, we extend their method into a pixel-wise one. For each pixel, a block with size (15*15) centered at the pixel is used to estimate the orientation of each pixel. To reduce the computational cost, a sliding window technique is implemented.

2.3   Reliability of Orientation Computing

An additional value rij is associated with each orientation element Oij to denote the reliability of the orientation. The value rij is low for noise and seriously corrupted regions and high for good quality regions in the fingerprint image. The reliability rij is derived by the coherence of the gradient Gij within its neighborhood. It is defined as follows:

$$r_{ij} = \frac{\left| \sum_W (G_{i,x}, G_{j,y}) \right|}{\sum_W \left| (G_{i,x}, G_{j,y}) \right|} = \frac{\sqrt{(G_{xx} - G_{yy})^2 + 4G_{xy}{}^2}}{G_{xx} + G_{yy}} \tag{5}$$

Where $(G_{i,x}, G_{j,y})$ is the squared gradient, $G_{xx} = \sum_w G_x{}^2$, $G_{yy} = \sum_w G_y{}^2$, $G_{xy} = \sum_w G_x \cdot G_y$ and (G_x, G_y) is the local gradient. W is taken as 11*11 block around (i,j).

2.4   Orientation Correction Based on Mixed Model

We proposed a mixed orientation model to correct the estimated orientation where its reliability is less than threshold thr. Zhou et al. [6] proposed a combination model to estimate the orientation field. However, in their algorithm, the fingerprints were not segmented into recoverable and unrecoverable regions. When their algorithm was applied to unrecoverable region, it will result many spurious minutiae in the post processing although the orientation field could be modeled.

2.4.1   The Mixed Orientation Model
The mixed orientation model is consisted of two parts, polynomial model and singular model. Due to the smoothness of the original orientation field, we could choose

proper polynomial curves to approach it. As the value of fingerprints' orientation is defined on $[0, \pi)$, it seems that this representation has an intrinsic discontinuity [6]. We map the orientation field to a continuous complex plane [10]. Denote $\theta(x, y)$ as the orientation field. The mapping is defined as:

$$U = R + iI = \cos(2\theta) + i \sin(2\theta) \tag{6}$$

where R and I denote the real part and image part of the unit-length complex, U, respectively.

To globally approach the function R and I, a common bivariate polynomial model is chosen for them respectively, which can be formulated as:

$$(1 \; x \; \cdots \; x^n) \cdot \begin{bmatrix} p_{00} & p_{01} & \cdots & p_{0n} \\ p_{10} & p_{11} & \cdots & p_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n0} & p_{n1} & \cdots & p_{nn} \end{bmatrix} \cdot \begin{pmatrix} 1 \\ y \\ \vdots \\ y^n \end{pmatrix} \tag{7}$$

where the order n can be determined ahead.

The singular area is the area near the singular points where the ridge curvature is higher than normal and where the direction of the ridge changes rapidly. So it is difficult to be modeled with polynomial functions. The orientation model proposed by Sherlock and Monro [5] is added at each singular point. We name Sherlock's model as the singular model. The model allows a consistent directional map to be calculated from the position of the cores and deltas only. In this model the image is located in the complex plane and the orientation is the phase of the square root of a complex rational function with the fingerprint macro-singularities (cores and deltas). Let c_i $(i = 1..n_c)$ and d_i $(i = 1..n_d)$ be the coordinates of the cores and deltas respectively; the orientation O' at each point (x,y) is calculated as:

$$O'(z) = O_0 + \frac{1}{2} \left[\sum_{i=1}^{n_d} \arg(z - d_i) - \sum_{i=1}^{n_c} \arg(z - c_i) \right] \tag{8}$$

where O_0 is the background orientation (we set $O_0 = 0$), and the function arg(z) returns the argument of the complex number z (x,y).

To combine the polynomial model with singular model smoothly, a weight function is defined for singular model, its weight at (x, y) is defined as:

$$w = \begin{cases} 0 & if \; (\sum_{i=1}^{k} w_i > 1) \\ 1 - \sum_{i=1}^{k} w_i & otherwise \end{cases} \tag{9}$$

$$w_i = \begin{cases} 0 & if \; (D_i(x, y) > r_i) \\ 1 - D_i(x, y)/r_i & otherwise \end{cases} \tag{10}$$

where k is the number of singular points, i is the ordinal number of singular points, $D_i(x,y)$ is the distance between point (x,y) and i-th singular point, r_i is i-th singular point's effective radius.

It is clear that for each point, its orientation is mainly controlled by singular model if it is near one of the singular points and it follows the polynomial model if it is far from the singular points.

Finally, the mixed model for the whole fingerprint's orientation field can be formulated as:

$$O_m = (1-w) \cdot \theta + w \cdot O^{'} \tag{11}$$

If the reliability of the estimated orientation rij is less than threshold thr, we will using the orientation O_m derived from the mixed model instead of the original estimated orientation.

2.4.2   Detection of the Singular Points and the Estimation of the Parameters

In order to implement the orientation correction algorithm, the position and type of singular points are need to detected. In our algorithm, the Poincare index method is used to detect the singular points. In order to compute the singular points more robust, we pre-process the original orientation field with the low bandpass filters to decrease the noise and smooth the orientation flows.

To implement the mixed orientation model, many parameters need to be ascertained. Some of them are initiated and modified based on the experiments while others are computed by least square method.

3   Experimental Results

The proposed algorithm has been evaluated on the databases of FVC2004 [11]. As the limits of pages, only the results on FVC2004 DB3 were listed in this paper.

The fingerprints of FVC2004 DB3 were acquired through thermal sweeping sensor "FingerChip FCD4B14CB" by Atmel. The size of the image is 300*480 pixels with the resolution of 512 dpi. Figure 2 show an example of low quality fingerprint in FVC2004 DB3 and its experimental results using the proposed algorithm. It can be seen from figure 2 that the computed orientation field of this poor fingerprint is smoothly and precisely.

Experiments were also done to compare the orientation estimation algorithm with and without feedback method. The comparison results on FVC2004 DB3 are shown in figure 3. The EER was 1.64 for the algorithm with feedback method, while 3.60 for the algorithm without feedback method. It is proved that the feedback method improved the performance the recognition algorithm greatly. The average time for estimation a fingerprint orientation in FVC2004 DB3 is about 0.06 second on PC AMD Athlon 1600+ (1.41 GHz).

4   Conclusion

This paper propose an robust orientation estimation with feedback algorithm to get the accurate fingerprint orientation. The proposed algorithm has been evaluated on

(a)	(b)	(c)	(d)
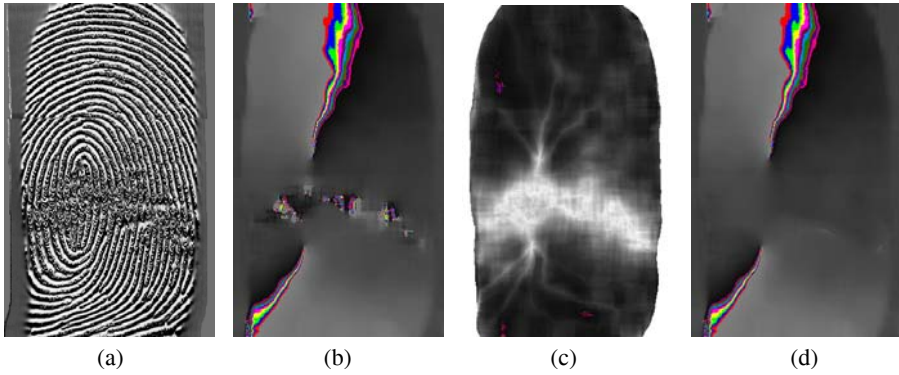
Fig. 2. Some experiment results of a fingerprint in FVC2004 DB3. (a)Original fingerprint (b)the coarse estimated orientation (c) the reliability of orientation (d) the corrected orientation
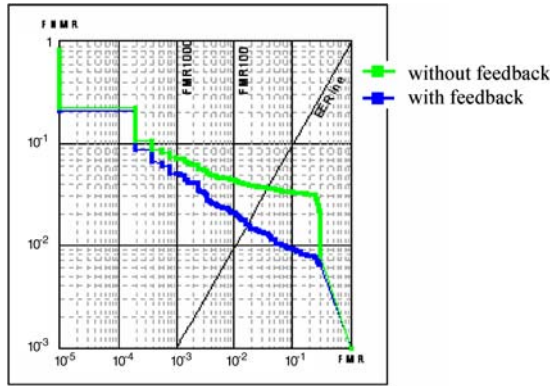
Fig. 3. The comparison the orientation estimation algorithm with and without feedback method on FVC2004 DB3

the databases of FVC2004. As the limits of pages, only the results on FVC2004 DB3 were listed in this paper. Experimental results confirm that the proposed algorithm is a reliable and effective algorithm for the extraction orientation field of the low quality fingerprints.

References

1. Li Jun, Yau Wei Yun, Prediction of Fingerprint Orientation, ICPR 2004, Aug. 2004 Volume: 4, Vol.4 On page(s): 436- 439
2. N. Ratha, K. Karu, S. Chen and A. K. Jain, A Real-time Matching System for Large Fingerprint Database, IEEE Trans. on Pattern Anal. Machine Intell., Vol. 18, No. 8, pp. 799-813, 1996.
3. Sarat C. Dass, Markov Models for Directional Field and Singularity Extraction in Fingerprint Images
4. A. Bazen, S. Gerez, "Systematic Methods for the Computation of the Direction Fields and Singular Points of Fingerprints", Pattern Analysis and Machine Intelligence,2002, 24 (7), pp. 905-919

5. B. G. Sherlock and D. M. Monro; "A Model for interpreting fingerprint topology"; Pattern Recognition, Vol. 26, No. 7, pp. 1047-1993.
6. J. Zhou and J. W. Gu, A Model-based Method for the computation of Fingerprints' Orientation Field, IEEE Trans. On Image Processing, Vol. 13, No. 6, pp.821-835, 2004
7. P.R. Vizcaya and L.A. Gerhardt; "A nonlinear orientation model for global description of fingerprints"; Pattern Recognition, Vol. 29, No. 7, pp. 1221-1231.
8. L. Hong, Y. Wan, A. K Jain, Fingerprint Image Enhancement: Algorithm and Performance Evaluation. IEEE Trans. PAMI, 20(8), pp.777–789, 1998.
9. Xinjian Chen, Jie Tian, Jianggang Cheng, Xin Yang, Segmentation of Fingerprint Images Using Linear Classifier, EURASIP Journal on Applied Signal Processing, EURASIP JASP 2004:4 (2004) 480-494.
10. G.H. Granlund and H. Knutsson, Signal Processing for Computer Vision, Kluwer Academic Publishers, Boston, 1995.
11. Biometric Systems Lab, Pattern Recognition and Image Processing Laboratory, Biometric Test Center, http://bias.csr.unibo.it/fvc2004/.

An Exact Ridge Matching Algorithm
for Fingerprint Verification*

Jianjiang Feng, Zhengyu Ouyang, Fei Su, and Anni Cai

Beijing University of Posts & Telecommunications, Post Box 111,
Beijing 100876, P.R. China
fjianjiang@263.net, brian122@vip.sina.com
{sufei,annicai}@bupt.edu.cn

Abstract. Unlike traditional minutiae-based fingerprint matching approach, which establishes the minutiae correspondence between two fingerprints, in this paper a novel ridge-based fingerprint matching algorithm, which establishes the exact ridge correspondence between two fingerprints is proposed. The ridge correspondence is called 'exact' because correspondences between points on ridges are also established. The matching score is computed as the ratio of the length of matched ridges to that of all ridges. Experimental results on FVC2002 databases show that the ridge matching approach performs comparably to minutiae-based one, and an obvious improvement on matching performance can be obtained by combining the two matchers.

1 Introduction

Although fingerprint-based identification is one of the most mature and proven automatic personal identification techniques, there are still a lot of challenges that are often caused by the following reasons: (1) low quality fingerprint images; (2) small size fingerprint sensors. (3) non-linear distortion.

A large number of fingerprint matching algorithms have been proposed in the literature. According to [1], existing fingerprint matching approaches can be coarsely classified into three families: correlation-based matching [2, 3], minutiae-based matching [4, 5], and ridge feature-based matching [6, 7]. Correlation-based methods preserve most of the information in fingerprints, but are fragile to nonlinear distortion; Minutiae-based methods are flexible, but sensitive to noise; and Ridge feature-based methods are robust to noise, but seem to be less distinctive. Since each approach has its strengths and weaknesses, hybrid approaches are also proposed [8].

The fingerprint matching approach proposed in this paper does not belong to any of the three families. This approach tries to combine the advantages of three families of matching approaches. It has the potential ability to preserve most of the information in fingerprints like correlation-based methods, to be distinctive and flexible like minutiae-based methods, and to be robust to noise like ridge feature-based methods. This approach is based on ridges. Ridge image, also called thinned image or skeleton image, is intermediate result in many fingerprint feature extraction algorithms. Although ridge-based fingerprint matching approaches have been neglected for many years, we

---

think that ridge image is very suitable for matching due to its distinctiveness and robustness.

Although ridges have been used implicitly in the correlation-based matching algorithms [2] and explicitly in the alignment algorithm [5], only Isenor and Zaky [9] tried to identify fingerprint by matching ridges explicitly. However their algorithm only establishes the rough correspondences of ridges. That is to say, their algorithm determines ridge *a* is matched to ridge *b*, but does not determine which portion of ridge *a* is matched to which portion of ridge *b*. Since fingerprints are often partially overlapped and ridges can be broken or connected wrongly due to noise or imperfection of the feature extraction algorithm, such rough correspondence of ridges is not enough for representing the exact matching degree of two fingerprints from the same finger. Meanwhile, for different fingerprints with similar global ridge patterns, good rough correspondences of ridges might be established. Therefore, establishing only rough ridge correspondences is not enough to determine whether two fingerprints are from the same finger.

In this paper, a novel fingerprint matching algorithm is proposed, which establishes exact ridge correspondence between two fingerprints. In our approach, a ridge is represented by a list of points sampled equidistantly on the ridge, and the correspondences of such points are established. It is in this sense that the ridge correspondence is called 'exact'. The ridge matching algorithm works as follows. First, a pair of ridges that are most similar is found. Then this pair of ridges is used as the base pair, and the ridges next to them are compared. Consequently the new matched ridges are used as the base pair, and the ridges next to them are compared, and so on. This recursive procedure proceeds until no more ridges can be matched. At last, a matching score is computed based on the length of matched ridges. Since the first base pair of ridges may not be the true matched pair, multiple pairs of ridges can be chosen as the initial base pairs and the above recursive matching procedure can be performed for multiple times. Finally the maximum score of the multiple procedures is used as the final matching score of the two fingerprints. Experiment results on FVC2002 databases [10] show that the proposed ridge-based matching approach performs comparably to the minutiae-based one, and an obvious improvement on matching performance is obtained by combining the two matchers.

In the next section, we describe the ridge matching algorithm in detail. The experimental results and analysis are given in section 3. Finally in section 4, summary is presented and future directions are suggested.

2   Ridge Matching

Given two grayscale fingerprint images, I_1 and I_2, we obtain their ridge images (thinned image), T_1 and T_2, and minutiae sets, M_1 and M_2, using an algorithm developed in our laboratory, which is similar to the algorithm described in [5]. Then our ridge matching algorithm accepts as input the two ridge images and the two minutiae sets, and outputs a score that indicates the degree of similarity between the two fingerprints. In 2.1, the representation of ridges is described. In 2.2, how to compare two aligned ridges is explained. In 2.3, how to find a set of initial base pairs of ridges is described. Given an initial pair of ridges, how to match all the remaining ridges, the most important part of the matching process, is presented in 2.4. In 2.5, the concept of relational distance is

adopted to prevent false matches. Finally in 2.6, the computation of the matching score is described.

2.1 Representation of Ridges

Due to noise in fingerprint images and imperfection of the feature extraction methods, there are complex structures, such as loops and bridges, in the ridge images. In order to make the ridge structure simple and consequently the following matching algorithm easier, a pre-processing step is performed: (1) Closed ridges are disconnected at an arbitrary point; (2) Ridges associated with bifurcations are split into three ridges; (3) Short ridges are removed. In the matching stage, it is unnecessary and time-consuming to examine every point on a ridge. Therefore, each ridge is sampled equidistantly and is represented by the list of sampled points.

2.2 Match Aligned Ridges

Given two aligned ridges, $\{a_i\}_{i=1}^M$ and $\{b_j\}_{j=1}^N$. The task of matching them is to find the longest continuous matched point strings. Two points are matched if their Euclidean distance is less than a threshold. We created an array $T(i,j), i=1,\cdots,M; j=1,\cdots,N$. $T(i,j)=1$ means a_i and b_j are matched. $T(i,j)=0$ means a_i and b_j are unmatched.

A list of index pairs $\{(P_l,Q_l)\}_{l=1}^K$ will be found, where P_l,Q_l are indices of $\{a_i\}_{i=1}^M$ and $\{b_j\}_{j=1}^N$ respectively, and P_l,Q_l should satisfy the following conditions: (1) $T(P_l,Q_l)=1$; (2) P_l,Q_l monotonously increase or decrease respectively; (3)

$$| P_l - P_{l-1} |\leq 2, | Q_l - Q_{l-1} |\leq 2; l = 2,...,K . \tag{1}$$

The last condition means that to make the algorithm more robust to noise, slight discontinuity in a matched point string is allowed. Such point strings can be found in T using dynamic programming technique [11]. If the longest list consists of more than K_{th} (i.e., K_{th}=3) pairs of points, the correspondence between matched points will be recorded.

2.3 Find the Initial Base Pairs of Ridges

This step is to find a set of N initial matched ridge pairs $\{(r_{1,i}, r_{2,j}) \mid r_{1,i} \in R_1, r_{2,j} \in R_2\}$ from two ridge sets R_1 and R_2. Since the pose transformation between the two fingerprints is unknown in advance, we restrict our search space to ridges associated with minutiae. We use the approach proposed in [5] to measure the similarity degree between any two ridges. The top N ridge pairs are selected as the initial matched ridge pairs.

2.4 Match All Ridges

Given an initial pair of matched ridges, the ridges next to them can be aligned and paired. Then the ridges next to the newly matched ridges can be aligned and paired, and

so on. Such procedure can be implemented as a recursive function. The pseudo-codes of the recursive function *MatchNearbyRidges* is given below. The input of the function is a pair of matched ridges br_1 and br_2, which will be used as the base ridge pair. When this function is called for the first time, the base ridge pair is one of the initial pairs of matched ridges. The function works as follows. First, the local pose transformation between br_1 and br_2 is computed. Then, for each pair of matched points in the base ridge pair and at each side of the base ridge, search for the possible matched ridges. For each pair of the possible matched ridges, align and match them using the approach described in subsection 2.2. Finally, each pair of ridges that is just matched during this call is used as the new base ridge pair to call *MatchNearbyRidges* in turn.

```
MatchNearbyRidges(br1,br2)
{
  clear matched_ridge_list;/* this list stores matched
                                 ridges during this call.*/
  trans = EstimateTransformation(br1,br2);
  for (p,q)=each pair of matched points of (br1,br2)
  {
    for side=left or right
    {
      nr1 = FindNearbyRidge(p,side);
      nr2 = FindNearbyRidge(q,side);
      if(nr1!=-1 & nr2!=-1 & !IsMatched(nr1)
           & !IsMatched(nr2))
      {
        AlignRidge(nr1,trans);
        if(MatchRidges(nr1,nr2)==true)
          add (nr1,nr2) to matched_ridge_list;
      }
    }
  }
  for (mr1,mr2)=each pair of matched_ridge_list;
    MatchNearbyRidges(mr1,mr2);
}
```

The result of comparing two aligned ridges can be completely unmatched, completely matched or partly matched. Due to noise, nonlinear deformation, partial overlap, and imperfection of the feature extraction algorithm, the corresponding ridges between two fingerprints from the same finger are not necessarily completely matched. Therefore, in the case of partly matched, unmatched parts of the ridges will be left over as new ridges that will be matched later.

2.5   Consistency Constraint

As described in the previous section, in order to deal with in-class variations in ridge structures, a ridge can be split into several short ridges. As a result, there are dangers that ridges from different fingers may match well with each other due to multiple splits. We adopted the concept of relational distance defined by Haralick and Shapiro [12] to prevent such cases.

Let $MR_1 = \{mr_{1,i}\}_{i=1,...,N}$ and $MR_2 = \{mr_{2,i}\}_{i=1,...,N}$ denote two sets of ridges, where ridge $mr_{1,i}$ is matched to ridge $mr_{2,i}$. We will define adjacency relation for matched ridges in the following fashion. Ridge i and ridge j have adjacency relation if they are from the same ridge and directly connected. Adjacency relation also contains the information whether head or tail of ridges is adjacent. The head and tail of a ridge can be specified arbitrarily, but for matched ridges, they should be consistent. Such information is recorded when a pair of ridges is found to be partly matched and ridges are split into multiple ridges. Let $s = (i, j, ei, ej)$ denote a relation, which means ei of ridge i and ej of ridge j are adjacent, where ei, ej can be head or tail. Let $S_1 = \{s_{1,i}\}_{i=1,..,N_1}$ and $S_2 = \{s_{2,i}\}_{i=1,...,N_2}$ denote all relations over MR_1 and MR_2, respectively. We say relation $s_1 = (i_1, j_1, ei_1, ej_1)$ and relation $s_2 = (i_2, j_2, ei_2, ej_2)$ are matched if one of the following conditions is satisfied: (1) Ridge i_1 is matched to i_2, and ridge j_1 is matched to j_2, and $ei_1 = ei_2$, and $ej_1 = ej_2$; (2) Ridge i_1 is matched to j_2, and ridge j_1 is matched to i_2, and $ei_1 = ej_2$, and $ej_1 = ei_2$.

Using the above definitions, we can compute the relational distance as follows. At first, set the relational distance to 0. For each relation in S_1, if no matched relation can be found in S_2, relational distance is increased by 1. Then for each relation in S_2, if no matched relation can be found in S_1, relational distance is increased by 1. We can reasonably expect that the relational distance of a genuine match should be small, and it is 0 when ridge matches are completely consistent. If the relational distance is greater than a threshold, the matching score for this matching procedure is set to 0.

2.6   Matching Score

As two fingerprints may share a small overlapped region, in order to evaluate the similarity degree of them, we should consider only the matching status in the overlapped region. For each of the two fingerprints, compute the convex hull of ridge images. The overlapped region is the intersection of two convex hulls that are aligned according to the initial pairs of ridges. The matching score $score_i (0 \leq score_i \leq 1)$ of the ith matching procedure can be computed according to the following formula:

$$score_i = \frac{N_{im}^{\ 2}}{N_{i1} \cdot N_{i2}} \quad . \tag{2}$$

where N_{i1} and N_{i2} is the number of ridge points of two fingerprints in overlapped region, and N_{im} is the number of matched sampled points. Then the final matching score is equal to the maximum of $score_i$.

3   Experimental Results

Experiments have been conducted on DB1_A of FVC2002. This databases consists of 800 fingerprint images from 100 fingers (eight impressions per finger). The performance evaluation protocol used in FVC2002 [10] has been adopted. For genuine match, each impression of each finger is compared with other impressions of the same finger. The number of matches is $C_8^2 \times 100 = 2800$. For impostor match, the first impression of

each finger is compared with the first impression of other fingers. The number of matches is $C_{100}^2 = 4950$.

Two examples, one for genuine match and another for impostor match, are given in Fig. 1. From the figure, we can see that most of the ridges in the overlapped region of two fingerprints from the same finger are matched, while most of the ridges in the overlapped region of two fingerprints from different fingers are unmatched. For examples shown in this figure, the matching score of genuine match is 0.95 and the matching score of impostor match is 0.11, which are obviously easy to be distinguished from each other.

The algorithm presented in this paper is compared with a minutiae-based matching algorithm previously developed in our lab. The ROC (Receiver Operating Characteristic) curves of these two methods are plotted in Fig. 2. The EERs of the ridge-based one and the minutiae-based one are 1.2% and 1.8%, respectively. In terms of EER (Equal Error Rate), the performance of ridge matching approach is slightly better than that of minutia matching one. In terms of ROC, however, it is hard to determine which one is better. In the region of low threshold, the performance of ridge matching approach is better than that of minutia matching one. In the region of high threshold, the result turns out contrary. Superiority in the region of low threshold means that the ridge matching approach is good at handling difficult genuine matches. This proves that, due to amount of information carried by ridges, the ridge matching approach is relatively robust to noise, small overlap area and nonlinear deformations. The Inferiority in the region of high threshold means that the current ridge matching approach behaves not well when handling difficult impostor matches. This should be caused by the following factors: the ridge images of different fingerprints of the same category can be similar, and local ridge images of different fingerprints can be similar too. However, we do not think this means distinctiveness of the ridge matching approach is lower. There is plenty of room for improvement of the current algorithm.

Both intuition and analysis on ROC curves remind us that ridges are complementary to minutiae and the combination of the two approaches could provide better performance. Therefore, a classifier combination with sum of matching scores given by the two matchers is tested on the same databases, using the same performance evaluation protocol. The EER is 1.0%. The ROC curves are also plotted in Fig. 2. From the figure, we can see that the ROC curve of the combiner is better than any single matcher in the whole threshold region.

The above experiments were conducted on a PC with P4 2.2 processor, running Windows 2000. The matching times of the ridge-based one and the minutiae-based one are 63 ms and 9 ms, respectively. Since the number of ridge points is approximately 20 times larger than that of minutiae, this fact is acceptable.

4   Summary and Future Work

We have proposed a ridge-based fingerprint matching algorithm. The basic idea is to establish the exact ridge correspondence between two fingerprints and compute matching score based on the length of matched ridges. The algorithm incrementally matches ridges using a recursive function. To allow for in-class variations in ridge structures, ridges can be split into small ridges. To prevent impostor matches due to ridge splitting, consistency constraints are imposed. Experiments on FVC2002 data-

Fig. 1. (a) Match result for fingerprints from same finger; (b) Match result for fingerprints from different fingers. Thick lines represent matched ridges and thin lines represent unmatched ridges
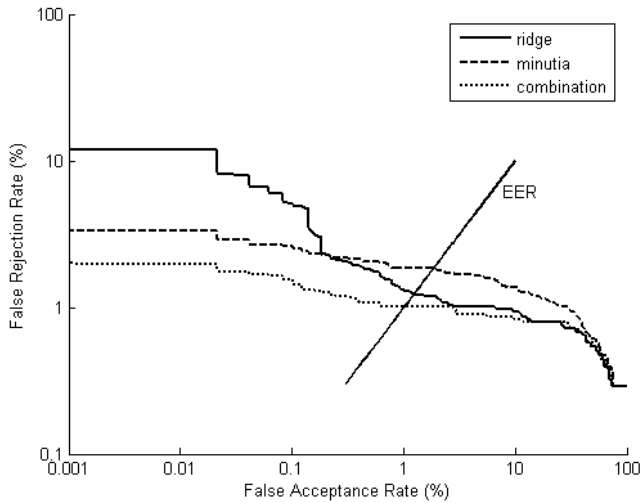
Fig. 2. ROC curves of ridge-based (*solid line*), minutiae-based (*dashed line*), combination algorithm (*dot line*) on DB1_A of FVC2002

bases show that the ridge-based approach performs comparably to the minutiae-based one and the matching performance can be further improved by a combination of the two complementary matchers.

There are some defects in current ridge matching algorithm. For genuine matches, searching for local optimal matches of individual ridges does not necessarily lead to a global optimal match of the whole ridge patterns; moreover, the matching procedure can be stopped at seriously damaged regions of the ridge images. For impostor matches, the local ridge relations exploited currently are too weak. Stronger consistent constraints should be introduced in our work in future. Furthermore, we notice that that in a ridge image, some ridges are more important or reliable than others. Therefore, it will be beneficial to assign a weight to each ridge to indicate its importance and to incorporate this information into the matching algorithm. Finally, the computational complexity of the algorithm should also be decreased.

References

1. Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: Handbook of Fingerprint Recognition. Springer-Verlag, Berlin Heidelberg New York (1996)
2. Wilson, C., Watson, C., Paek, E.: Effect of Resolution and Image Quality on Combined Optical and Neural Network Fingerprint Matching. Pattern Recognition 33 (2) (2000) 317-331
3. KovQacs-Vajna, Z.M.: A Fingerprint Verification System Based on Triangular Matching and Dynamic Time Warping. IEEE Trans. Pattern Anal. Mach. Intell. 22 (11) (2000) 1266–1276
4. Ratha, N.K., Karu, K., Chen, S., Jain, A.K.: A Real-time Matching System for Large Fingerprint Databases. IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) (1996) 799-813
5. Jain, A.K., Hong, L., Bolle, R.: On-line Fingerprint Verification. IEEE Trans. Pattern Anal. Mach. Intell. 19 (4) (1997) 302-313
6. Jain, A.K., Prabhakar, S., Hong, L., Pankanti, S.: Filterbank-based Fingerprint Matching. IEEE Trans. Image Processing 9 (5) (2000) 846-859
7. Coetzee, L., Botha, E.C.: Fingerprint Recognition in Low Quality Images. Pattern Recognition 26 (10) (1993) 1441-1460
8. Ross, A., Jain, A.K. Reisman, J.: A Hybrid Fingerprint Matcher. Pattern Recognition 36 (7) (2003) 1661-1673
9. Isenor, D.K., Zaky, S.G.: Fingerprint Identification Using Graph Matching. Pattern Recognition 19 (2) (1986) 113-122
10. Maio, D., Maltoni, D., Cappelli, R., Wayman, J.L., Jain, A.K.: FVC2002: Second Fingerprint Verification Competition. In: Proceedings of the International Conference on Pattern Recognition (ICPR), Quebec City, Canada (2002) 744–747
11. Cormen, T.H., Leiserson, C.E., Rivest, R.L.: Introduction to Algorithms. McGraw-Hill, New York (1990)
12. Haralick, R., Shapiro, L.: The Consistent Labeling Problem II. IEEE Trans. Pattern Anal. Mach. Intell. 2 (1980) 193-203

Adaptive Fingerprint Enhancement by Combination of Quality Factor and Quantitative Filters

Xuchu Wang[1], Jianwei Li[1], Yanmin Niu[2], Weimin Chen[1], and Wei Wang[1]

[1] Key Lab on Opto-Electronic Technique of State Education Ministry, Chongqing University,
400044, Chongqing, P.R. China
Seadrift@sina.com, jwli@cqu.edu.cn
[2] College of Physics and Information Techniques, Chongqing Normal University,
400047, Chongqing, P.R. China
nym4612@yahoo.com.cn

Abstract. Fingerprint enhancement is a crucial step in automatic fingerprint recognition system, undiscriminating repeated filtering easily lead to false structure in low-quality regions of a fingerprint image. This paper presents a new adaptive enhancement algorithm that can automatically adjust the parameters of filters and the time of filtering according to the quality factors in different regions. In order to improve filtering efficiency, a template bank of 4-dimenstion scaleable array is also designed to quantize the filter and forms the basis of a new fused filter. Experimental results in eight low-quality images from FVC2004 data sets show that the proposed algorithm is higher 23.7% in Good Index (GI), and saves 54.06% time consumptions than traditional Gabor-based methods. Since the eight images are extremely bad, a little improvement is very meaningful.

1 Introduction

Fingerprint image enhancement plays an important role in many biometrics recognition systems based on fingerprint. Since a finger impression is often regarded as a pattern of ridges and valleys on the surface of a fingertip [1], this character must be maturely considered during the enhancement process. Many approaches have investigated in this topic by virtue of orientations in ridges and distances between ridges, and the corresponding filters were designed mainly in spatial domain [2] and frequency domain [3][4], Especially in [4], Gabor function was employed as filter and became a basis for fingerprint image enhancement. However, the filtering is relatively time-consuming and performs not very well in low-quality regions.

In previous literature, most approaches focused on how to design an enhancing filter or filter bank, but not adaptive enhancing filters. While in actual applications, due to many factors (cracks, too dry or too wet conditions of fingertip, nonlinear deformation by different pressure, problems with acquisition devices and other more), the variety in quality does not happen in different fingerprint images, but also in different regions of a same image. In these conditions, as shown in Fig.1, it is very necessary for repeated enhancements. The average way is to simply enhance the whole image without considering the difference of quality, which sharps the structure information in bad-quality regions and at the same time, is easy to destroy the minutiae information in good-quality regions. Furthermore, it increases the computational time.

Aimed at avoiding these disadvantages, we propose a new adaptive enhancement algorithm in this paper. The algorithm can automatically adjust the parameters of filters

and the time of filtering according to the quality factors in different regions. In order to improve filtering efficiency, we also design a template bank of 4-dimenstion scaleable array to quantize the filter, during the enhancement it just need to call the templates by index, and avoids repeatedly calculating the filter mask. Experimental results validated our algorithm and the comparison in eight low-quality images from FVC2004 data sets shows that our algorithm is higher 0.11 (or 23.7%) in Good Index (GI), and less 0.36625 second (or 54.06% time savings) in time consumptions than traditional Gabor-based methods. The experiment also indicates that the traditional method heavily relies on the quality of input fingerprint image and exposes the disadvantage of indistinctively repeated filtering, and it proves the correctness of way of adaptive enhancement. Our proposed method can avoid the defect of repeated filtering, protect minutiae, and improve clarity of ridges.

Fig. 1. An example of fingerprint varies in quality. The middle or above quality region (rectangle) and the low-quality region (ellipse) can be enhanced well for just one time, or for many times without consideration of their quality difference?

2   Proposed Algorithm

2.1   Estimation of Orientation and Quality Factor

Fingerprint represents the combinations of convex and concave ridges and a direction along clear ridges in local regions is often consistent, this character can be applied to estimate a dominate orientation and evaluate the level of clarity of ridges in local region. Generally, divide a fingerprint image into blocks with size of W×W, and assume the dominate orientation in each block is unique. Here we set W as 16, then construct a feature vector with size of W×W using directional masks such as Sobel, Prewitt, Marr and etc, we select Sobel mask and estimate the local dominate orientation by LMS method, and the formula is as follows:

$$D_{xy} = \sum_{m=i}^{i+w}\sum_{n=j}^{j+w} d_x(m,n)d_y(m,n)$$

$$D_{xx} = \sum_{m=i}^{i+w}\sum_{n=j}^{j+w} d_x^2(m,n)$$

$$D_{yy} = \sum_{m=i}^{i+w}\sum_{n=j}^{j+w} d_y^2(m,n)$$

$$\theta(i, j) = 0.5 * \tan^{-1}(2D_{xy}/(D_{xx} - D_{yy}))$$

(1)

where $d_x(m,n), d_y(m,n)$ are gradients of each pixel along *x, y* coordinates, and $\theta(i,j)$ is the dominant orientation of normal in the same region, in order to eliminate influence of noise factors, it is a feasible way of using a low-pass to filter the noisy orientations by the neighborhood information after transform $\theta(i,j)$ into a vector field of $\sin(2\theta(i,j))$ and $\cos(2\theta(i,j))$ to remove the break between π and 0.

In the low-quality regions, Eq.(1) often estimates false orientations and after filtering, although their orientations will tend to rightness, the clarity of ridge structure is often obscure and the contrast is relatively poor, we need to filter repeated in these regions to emphasis the structures of ridges. So we design two quality factors $Q(i,j)$ and $C(i,j)$ to extract these regions.

$$Q(i,j) = \sqrt{\frac{(D_{xx}-D_{yy})^2 + 4D_{xy}^2}{(D_{xx}+D_{yy})^2}}$$

$$C(i,j) = \sqrt{\frac{D_{xx}+D_{yy}}{2\,|D_{xy}|}}$$

(2)

The intention of Eq.(2) is to give the statistical scatter level of gradients along *x, y* coordinates according to the characters that there is no evidently dominate orientation or diverges the right orientations apparently. In the regions where $Q(i,j)$ is near to 1, the region has clear ridges and just needs to enhance once or twice. Similarly, if $Q(i,j)$ in a region is near to 0, which means this region either belongs to background where $C(i,j)$ is near to 0, then we needn't to enhance it, or it belongs to a class of bad region, and we need to enhance it repeatedly. During the repeated filtering, the clarity of regions in low-quality will be gradually improved and make $Q(i,j)$, $C(i,j)$ both near to 1, then the enhancement will be automatically stopped.

2.2   Estimation of Frequency of Ridges

Like the dominant orientation of ridges in local regions, the average distance between ridges is another essential character of fingerprints. The reciprocal of distance is regarded as frequency of ridges, and it can be estimated by the assumption of the similarity of distance in local regions. We can set a rectangle whose center is same as the center of the local block, and whose width and height are parallel to the ridge orientation and the normal orientation, respectively. Then accumulate the gray sum value of R pixels along the ridge orientation and calculate the average gray value. If the local region is clear, the array of the average gray value will be a discrete sine wave, calculate average distance of two neighbor peaks and the reciprocal is frequency of ridges, or else the peaks cannot be gotten, we must use the parameters of its neighborhoods.

2.3   Enhancement of Ridges

After estimating the two essential parameters of a fingerprint image, we can design an appropriate band-pass filter to emphasis the structure of ridges and eliminate the noise among the ridges. Gabor function has the optimal joint resolution in both spatial and frequency domain, so we use 2-dimension even Gabor function to design the enhancement filter. By applying properly tuned Gabor filters to a fingerprint image, the

ridge and valley structures can be right accentuated. In practice, we should change the Gabor function as a digital filter:

$$h(m,n) = \exp(-(x^2/\delta_x^2 + y^2/\delta_y^2)/2)\cos(2\pi fx)$$

$$x = (n-j)\cos\theta + (m-i)\sin\theta$$

$$y = (j-n)\sin\theta + (m-i)\cos\theta \qquad\qquad (3)$$

$$G'(i,j) = \sum_{m=0}^{w'-1}\sum_{n=0}^{w'-1} h(m,n)G(i-m,j-n)$$

where θ and f are gradient orientation and frequency of ridges in local regions, δ_x, δ_y are the Gaussian parameters along and vertically along the axis of ridge orientations and $\delta_x \le \delta_y$, here we set $\delta_x = 3, \delta_y = 4$; x, y are new coordinates after rotating angle θ of each pixel (m, n) in local regions. G and G' are images before and after filtering, respectively. The last one is the size of filter window W'. Many references often set it as same as the block size, while in our approach, W' can be automatically changed according to the quality variety.

2.4   Quantization of Filter

The Gabor filter is observed not very sensitive to the orientation of ridges in many experiments. In order to improve filter efficiency, we quantitate the filter into 8 discrete orientations and the corresponding index is:

$$a = \lfloor \theta \bmod \pi \rfloor \qquad\qquad (4)$$

where $\lfloor \cdot \rfloor$ is a floor integer operation and $a = 0,1,2,...,7$, at the same time, the average distance of ridges W' is among 5~20, furthermore, the size of filter mask B is among 5~24, so we can design the filter into a 4-dimension array and the filter is replaced as:

$$h(m,n) = \{g_{w'}^a(m,n)\}_{w'=5,a=0,m=0,n=0}^{w'=20,a=7,m=B-1,n=B-1} \qquad\qquad (5)$$

2.5   Adaptive Enhancement of Ridges

According to the quality variety in different regions of a same fingerprint image, we set the following rules during the filtering process:

IF $Q(i,j) < Q_0$ and $C(i,j) < C_0$

$G'(i,j) = 255$

ELSE IF $Q(i,j) < Q_0$ and $C(i,j) > C_0$

$$G'(i,j) = \sum_{m=0}^{B1-1}\sum_{n=0}^{B1-1} g_{w1}^a(m,n)G(i-m,j-n)$$

ELSE IF $Q(i,j) \in [Q_0,Q_1]$ and $C(i,j) \in [C_0,C_1]$

$$G'(i,j) = \sum_{m=0}^{B2-1}\sum_{n=0}^{B2-1} g_{w2}^a(m,n)G(i-m,j-n)$$

ELSE

$G'(i,j) = G(i,j)$

where Q_0 , Q_1 , C_0 , C_1 are presetting thresholds and $Q_0 < Q_1$, $C_0 < C_1$. Here we let $Q_0 = 0.015, Q_1 = 0.985$, $C_0 = 0.15, C_1 = 0.85$. $g_{w1}^a(m,n), g_{w2}^a(m,n)$ are template masks of Gabor digital filter after quantization with different parameters and the filter size is changeable ($B1 > B2$). When the quality of local region is high, the small filter size is helpful to grasp the local ridges and deduce the computational cost, and when the quality is low, the filter size tunes up to large and uses the neighbor ridges to eliminate the disturbance of noise and restore the ridges structure. When the case $Q(i,j) > Q_0 \text{ and } C(i,j) < C_0$ happens, it means that the enhancement has gotten clear ridge structures and needn't to filter any more. So the procedure of enhancement is adaptive and refers to quality status of different regions.

When the frequency or the orientation of a block is invalid, we estimate them not by the values of 3×3 or 5×5 neighborhoods, but by a fused filter, and the formula is as follows. Let $V(r,s)$ be the filter mask vector representing that 4-dimension array in block (r,s), and $V(i,j)$ be the filter mask vector of invalid block (i,j), then:

$$V(i,j) = \sum_{r=i-k}^{i+k} \sum_{s=j-k}^{j+k} T(r,s)V(i-r,j-s); k = 1,2$$

$$V(i-r,j-s) \in \{g_{w'}^a(m,n)\}_{w'=5,a=0}^{w'=20,a=7}$$

(6)

where $T(r,s)$ is a mean or Guassian smoothing filter mask, each element in $V(i,j)$ is estimated by the right elements in the blocks nearby. The reason of taking this way is that although Gabor function can reach the low boundary of resolution in spatial-frequency domain, its character of selecting orientation and frequency (especially frequency) is easy to destroy the ridges hidden in noisy pixels. As shown in Fig.2, undiscriminating repeated filter causes bifurcations of ridges and hardly to be remedied. We want to avoid these cases, so the aim in the first steps of enhancement is prior to smooth the low-quality region as possible, in the last steps is to sharpen the structure. We can say the fused filter intends to de-Gaborlized in some sense. It seems that in Fig.2c the contrast between ridges and valleys is not very evident in low quality regions, in fact, the following filtering can provide clearer and better ridges than Fig.2b.

(a)   (b)   (c)

Fig. 2. Effect of the fused filter. (a)original image with sweat blur and great pressure; (b) image enhanced twice by Gabor-based method; (c) image adaptively enhanced by Eq.(6) where maximum time of filtering is twice. The low-quality region in (a) has been sharpened in (b), while the ridge structure with sweat blur is destroyed much more remarkable than in (c). Priority of weighted smoothing in (c) has protected the ridges in low-quality regions

3   Experimental Results and Discussion

Our adaptive enhancement algorithm was implemented in Microsoft Visual C++ 6.0 in Microsoft Windows XP SP2 platform with Intel Celeron 1.7GHz   CPU. Fig.3 gives an example of an input fingerprint image and two output images with our method and traditional Gabor-based enhancement method. It can be found from the figures (c) and (d) that in the high-quality regions, our method is similar to the traditional method with more protection of endings and bifurcations of ridges, while in the low-quality regions, traditional method still need more repeated filtering after 4 times and our method has stopped with a satisfaction.
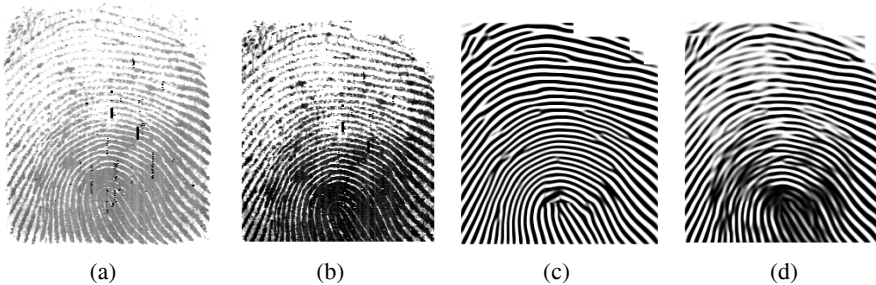

| (a) | (b) | (c) | (d) |

Fig. 3. An example of enhancement result. (a)original input image. (b)normalized and segmented image. (c)enhanced image with our proposed algorithm and it takes 0.19 seconds, and (d) enhanced image with Gabor-based filter by 4 repeated times and it takes 0.54 seconds

   Apparently, the figures argue that our algorithm have made evident improvements both in filter quality and time consumption. In order to further evaluate the performance, we select eight low-quality images in four data sets of FVC2004[6], they are 107-8,109-6 in DB1, 106-8-L,107-7-W in DB2, 103_8_L, 104-8 in DB3, 101-1,104-5 in DB4, and use Good Index (GI) proposed in [4][5] to evaluate the performances. Fig.4 shows an example of experimental results, in the high-quality regions, our algorithm is still similar to the traditional Gabor-based method, while in low-quality regions, the performance of the traditional method is relatively bad and it still needs more filtering. Although our algorithm extracts some false minutiae, the number is far less than those extracted by traditional method. The test strongly favors our algorithm from another point.

   Table 1 gives the GI and time consumption of each image by the two algorithms. Our algorithm has two advantages than the latter:

 (i) GI is higher 0.11, in other words, higher 23.7%. Because the eight images are extremely bad, a little improvement will be very meaningful for fingerprint matching such as shown in Fig.4(c) and (e).

(ii) Time consumption decreases 0.36625 second, is less than half of the traditional method, in other words, 54.06% of computation time are saved. The variance of time consumption in traditional algorithm is 0.1997, higher 0.11768 than our algorithm, this fact shows that the traditional method heavily relies on the quality of input fingerprint image and exposes the disadvantage of indistinctively repeated filtering, and it also proves the correctness of way of adaptive enhancement.

Table 1. Confused matrix of GI and time consumptions in eight low-quality images selected from four data sets of FVC2004

Image	Our algorithm		Traditional algorithm	
	GI	Time(s)	GI	Time(s)
1	0.76	0.38	0.64	0.86
2	0.73	0.21	0.67	0.55
3	0.39	0.37	0.34	0.72
4	0.41	0.35	0.40	0.69
5	0.63	0.32	0.39	0.67
6	0.59	0.41	0.31	1.02
7	0.57	0.26	0.55	0.53
8	0.52	0.19	0.42	0.38
avg	0.575	0.31125	0.465	0.6775
var	0.13416	0.08202	0.137009	0.1997

(a)   (b)   (c)

(d)   (e)
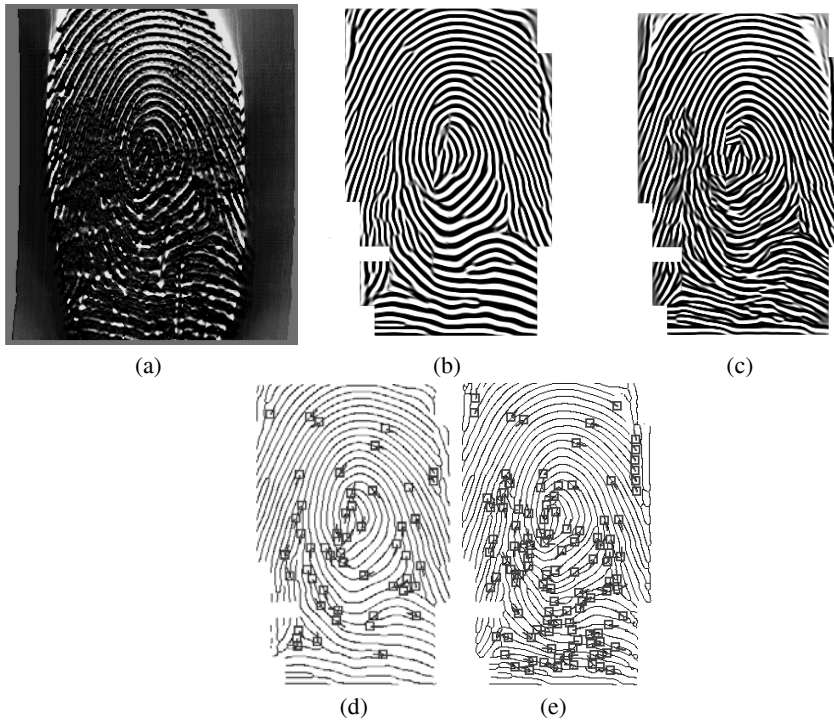
Fig. 4. Result of low-quality images enhancement. (a) input image whose main part is from 103-8-L in FVC2004 DB3. (b) and (d) are enhanced image and minutiae image by our proposed algorithm. (c) and (e) are by traditional Gabor-based algorithm

4   Conclusions

We have presented a new adaptive enhancement algorithm for fingerprint images. The algorithm automatically selects parameters of filter and times of filters in different regions of an image. The experiments shows that our algorithm is much more effective

than traditional Gabor-based methods and it can not only accentuate the ridges, but also can protect the endings and bifurcations of ridges. The test in eight low-quality images selected from FVC2004 data sets shows that the average GI of our algorithm is higher 0.11, in other words, higher 23.7%, and the time consumption is less 0.36625 second, also 54.06% time savings than the traditional Gabor-based methods. The variance of time consumption in traditional algorithm is 0.1997, higher 0.11768 than our algorithm, this fact shows that the traditional method heavily relies on the quality of input image and exposes the disadvantage of indistinctively repeated filtering, and it also proves the correctness of way of adaptive enhancement.

References

1. D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar. Handbook of Fingerprint Recognition, New York: Springer, (2003).
2. L.O'Gorman, and J.V.Nickerson. An Approach to Fingerprint Filter Design, Patter Recognition, 22(1) (1989) 29-38.
3. B.G. Sherlock, D.M. Monro, and K. Millard, Algorithm for Enhancing Fingerprint Images, Electronics Letters, 28(18) (1992) 1720-1721.
4. L.Hong,Y.F.Wang, and A.K.Jain, Fingerprint Image Enhancement: Algorithm and Performance Evaluation, IEEE Trans. PAMI, 20(8) (1998) 777-789.
5. N.Ratha, S.Chen, and A.K.Jain, Adaptive Flow Orientation Based Feature Extraction in Fingerprint Images. Pattern Recognition, 28(11) (1995) 1657-1672.
6. http://bias.csr.unibo.it/fvc2004/
7. X.C. Wang, J.W. Li, and Y.M. Niu: Fingerprint Classification Based on Curvature Sampling and RBF Neural Networks, Lecture Notes in Computer Science, 3497(2005)171-176

Fingerprint Classification Based on Statistical Features and Singular Point Information

Zhi Han[1,2] and Chang-Ping Liu[1]

[1] Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China
[2] Graduate School of Chinese Academy of Sciences, Beijing 100080, China
`zhi.han@mail.ia.ac.cn`

Abstract. Automatic fingerprint classification is an effective means to increase the matching speed of an Automatic Fingerprint Identification System with a large-scale fingerprint database. In this paper, an automatic fingerprint classification method is proposed to classify the fingerprint image into one of five classes: Arch, Left loop, Right Loop, Whorl and Tented Arch. First, the information of core points, which is detected with a two-stage method, is applied to determine the reference point in fingerprint image. Then three different features based on statistical properties of small image blocks, which are likely to degrade with image quality deterioration, are calculated from the region of interest and form a 300-dimension feature vector. The feature vector is inputted into a three-layer Back Propagation Network (BPN) classifier and a 5-dimension vector is outputted, each dimension of which corresponds to one of 5 fingerprint classes. Finally, the fingerprints are classified with integrate analysis of the BPN classifier output and singular point information. The accuracy of 93.23% with no rejection is achieved on NIST-4 database and experimental results show that the proposed method is feasible and reliable for fingerprint classification.

1 Introduction

Fingerprints have been considered as one of the most reliable biometric features applied to personal identification for their uniqueness and unchangeableness [1]. In an Automatic Fingerprint Identification System, an incoming fingerprint is matched against all the fingerprints in the database to find out whether an individual represented by the incoming fingerprint is the same as an individual represented by one of fingerprints in the database. However, such one-by-one matching may require very much computation when the fingerprint database is of huge size. Fingerprint classification refers to partition of all fingerprint images in database into several smaller exclusive sets according to some predefined classification criteria. An accurate and reliable fingerprint classification method can greatly reduce the size of the search space and accordingly increase the matching speed of a large-scale fingerprint database. The classification criteria of most automatic fingerprint classification are the modification or extension of the standard Henry's classification system [2] and most widely used criteria classify the fingerprint images into 5 classes: Arch (A), Left loop (L), Right Loop (R), Whorl (W), Tented Arch (T). Fig.1 shows the example fingerprints of the five classes.

Many fingerprint classification methods have been proposed in literature. In general, these methods can be categorized into four approaches: syntactic-based, singularity-based, structure-based and frequency-based. In syntactic approach [3], finger-

print images are represented and classified with a formal grammar. Singularity-based approach uses the information such as number, type and location of singular points to classify fingerprint images [4-5]. Structure-based methods [6-13] use structure information such as orientation field or ridge distribution in fingerprint image to make classification with different classifiers, such as neural network, HMM, SVM etc. Frequency-based methods use the features in frequency spectrum of fingerprints in classification [14-15]. In addition, some methods combine two or more approaches for classification [16-17].
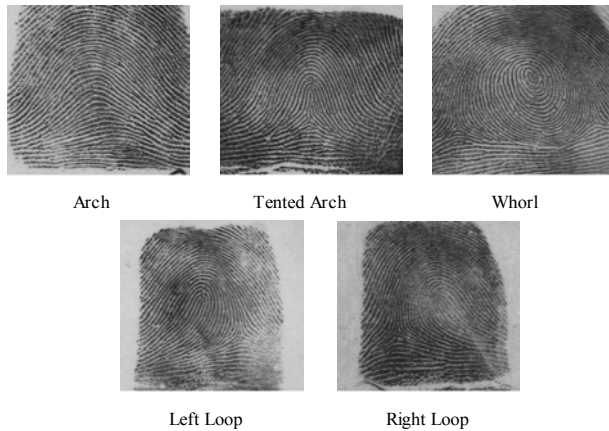

<div align="center">

Arch   Tented Arch   Whorl

Left Loop   Right Loop

</div>

Fig. 1. Example images of 5 major fingerprint classes from NIST-4 database

In this paper, an automatic fingerprint classification method is proposed to classify fingerprints into five classes shown in Fig. 1. First, the information of core points, which is detected with a two-stage method, is applied to determine the reference point in fingerprint. Then three different features based on statistical properties of small image blocks, which are likely to degrade with image quality deterioration, are calculated from the region of interest and form a 300-dimension feature vector. Then feature vector is inputted into a three-layer BPN classifier and a 5-dimension vector is outputted, each dimension of which corresponds to one of 5 fingerprint classes. Finally, the fingerprints are classified with integrate analysis of the BPN classifier output and singular point information.

The rest of this paper is organized as follows: Section 2 describes the details of the algorithm. Section 3 gives experimental results and Section 4 draws a conclusion.

2   Fingerprint Classification Algorithm

The main steps of proposed method are: (i) singular points extraction with a two-stage method; (ii) reference point determination (iii) statistical features extraction from the region of interest centered on the reference point; (iv) training of a BPN classifier; (v) classifying fingerprints based on integrate analysis of the BPN classifier output and singular point information.

2.1 Singular Points (SP) Extraction

The singular points (SP), namely core points (CP) and delta points (DP), are the points where the sharpest change of ridge orientation occurs in fingerprint images. The information of SPs in a fingerprint image is used to determine the reference point and classify the fingerprints in proposed fingerprint classification method. In this paper, a two-stage SP detecting method as follows is adopted:

(1) The fingerprint image is divided into several non-overlapping blocks first. Then the orientation of each block is represented by the orientation of the center pixel in that block and is computed with the method proposed in [18] and then is normalized to 12 fixed integer value $\{0, 1,…, 11\}$. The block orientation image composed with orientations of all the blocks is denoted as O_B.

(2) For each pixel in O_B, the number of different orientation values in the 8-neighboring pixel is computed and denoted as $N_{Direction}$. If $N_{Direction} < 4$, then this block is considered as including no SP.

D_6	D_5	D_4	D_3	D_2
D_7	d_3	d_2	d_1	D_1
D_8	d_4	(i,j)	d_0	D_0
D_9	d_5	d_6	d_7	D_{15}
D_{10}	D_{11}	D_{12}	D_{13}	D_{14}

Fig. 2. Two selected closed paths for the given pixel to calculate the Poincaré Index

(3) Let $S = \{c \mid N_{Direction}(c) \geq 4, c \in O_B\}$. For a given pixel (i, j) in S, the Poincaré Index (PI) values along two closed paths $D_0 D_1...D_{15} D_0$ and $d_0 d_1...d_7 d_0$, which are shown in Fig.2, are computed respectively in order to decrease the noise influence. The PI for the given pixel (i, j) along the path $D_0 D_1...D_{15} D_0$, named $PI_1(i, j)$, can be computed as follows:

$$PI_1(i,j) = \sum_{k=0}^{15} \Delta(k) \text{, where } \Delta(k) = \begin{cases} \delta(k) & |\delta(k)| < 6 \\ 12 + \delta(k) & \delta(k) < -6 \\ 12 - \delta(k) & otherwise \end{cases}$$

$\delta(k) = o(D_k) - o(D_{(k+1)\bmod 16})$, $o(D_k)$ represents the orientation of the pixel D_k.

The PI for the given pixel (i, j) along the path $d_0 d_1...d_7 d_0$, named $PI_2(i, j)$ can be computed in the same way.

(4) Denote S_{sp} as the set of candidate SPs in O_B, then S_{sp} can be calculated according to the PI values as follows:

$$S_{Core} = \{(i, j) \mid PI_1(i, j) \geq 10 \,\&\, PI_2(i, j) \geq 10, (i, j) \in O_B\}$$
$$S_{Delta} = \{(i, j) \mid PI_1(i, j) \leq -10 \,\&\, PI_2(i, j) \leq -10, (i, j) \in O_B\}$$
$$S_{sp} = S_{Core} \bigcup S_{Delta}$$

where S_{Core} and S_{Delta} are the candidate CPs set and candidate DPs set respectively. The candidate singular point blocks are those corresponding to the pixels that belong to the set S_{sp}. The candidate region of SPs is the set of all the candidate singular point image blocks and is represented as S_{CBR}.

(5) For each pixel in each candidate block, calculate the local orientation variance (LOV) with the equation $V(P) = \sum_{k=0}^{7} |o(P) - o(NP(k))|$, where $V(P)$ denotes LOV value at pixel P, $o(P)$ denotes the orientation at pixel P, $NP(k)$ refers to the k th 8-neighboring pixel of the given pixel. In each candidate block, the pixel with the maximum LOV value is considered as one candidate SP. For one candidate SP P, assume the block containing P is B_P. If $B_P \in S_{Core}$, then P is considered as a CP; if $B_P \in S_{Delta}$, then P is considered as a DP.

(6) Considering that LOV of the point near the SP is also very great and the point may be in the different block from the SP, if the distance between two candidate SPs is small than a threshold, delete the point with smaller LOV value.

(7) Check all candidate SPs. If the number of candidate CPs is more than 2, the two points with larger LOV are selected as the CPs; similarly, if the number of DPs is more than 2, the two points with larger LOV are selected as the DPs.

2.2   Reference Point Determination

Reference point is used to align the fingerprint images before statistical features are extracted. The information of core points, which are detected with the method described in section 2.1, is used here to determine the reference point. If only one CP is detected in the image, then this CP is considered as the reference point; if 2 CPs are detected, the upper one is selected as the reference point. In the case no CP is detected, the point with maximum curvature of the concave ridges in fingerprint image is detected with the method in [19] and is considered as the reference point.

2.3   Statistical Features Extraction

After the reference point is determined, features need to be extracted for fingerprint classification. Considering the features should be applied to both good quality and poor quality fingerprint images, features based on statistical properties of image are selected in this paper because they are likely to degrade with image quality deterioration. The steps of feature extraction are as follows:

(1) The region centered on reference point with size $R \times R$ is defined as the region of interest for feature extraction and is denoted as ROI. In this paper, $R = 160$.

(2) Extract features: Divide ROI into non-overlapping blocks of size $w_B \times w_B$. We select $w_B = 16$ in this paper. Therefore, there are 100 blocks in ROI. For each block $B_i (i = 1,...,100)$, the mean M_i and variance V_i of pixel values in the block are calculated with the following equations respectively:

$$M_i = \frac{1}{n_i} \sum_{(k,l) \in B_i} f(k,l)$$

$$V_i = \frac{1}{n_i} \sum_{(k,l) \in B_i} (f(k,l) - M_i)^2$$

where n_i refers to the number of pixels in block B_i, $f(k,l)$ refers to the gray value of pixel (k,l).

In addition, the number of black pixels in each block B_i, which is selected as another class of feature, is calculated and denoted as C_i. Before C_i is calculated, the gray-level image needs to be binarized first. An adaptive threshold method, where the average gray value of the pixels in a 5x5 neighborhood is set as the threshold for that pixel, is adopted to make binarization in this paper[20].

(3) Create the feature vector: 3 statistical features in each block define the components of 300-dimension feature vector. Assume F denotes the feature vector and then F can be represented as $F = [M_1, V_1, C_1, ..., M_{100}, V_{100}, C_{100}]^T$.

2.4   Back Propagation Network (BPN) Classifier

After the feature is extracted, a 3-layer BPN is trained to classify the input feature vector into 5 classes. The neural network has one hidden layer with 20 neurons, 300 input neurons corresponding to the 300-dimension statistical feature vector, and 5 output neurons corresponding to the five fingerprint classes. The weight matrix of neural network is trained with a quick back propagation-training algorithm [21]. The training set includes 2000 fingerprint images, each of which is represented with a 300-dimension feature vector. Before the feature vector is inputted into BPN classifier, both in training stage and testing stage, the feature vector needs to be normalized and the value of each dimension is normalize to [0, 1].

The 5 output neurons of the classifier compose a 5-dimension vector $C_{out} = [c_1, c_2, c_3, c_4, c_5]$, where c_i is the output value of each output neuron and the range of c_i is [0,1]. Each output neuron corresponds to one fingerprint class, namely the i th neuron corresponds to class i. We predefine that class 1 refers to Arch, class 2 refers to Left Loop, class 3 refers to Right Loop, class 4 refers to Whorl and class 5 refers to Tented Arch. Then, the expected output for each class are: Arch: [1,0,0,0,0]; Left Loop: [0,1,0,0,0]; Right Loop: [0,0,1,0,0]; Whorl [0,0,0,1,0]; Tent Arch [0,0,0,0,1]. If a fingerprint is classified only with the output of the trained BPN classifier, the fingerprint will be classified into the class corresponding to the neuron with maximum value among all the output neurons.

2.5   Fingerprint Classification Based on BPN Classifier Output
and Singular Points Information

In this paper, fingerprint images are classified based on integrate analysis of the BPN classifier output and SPs information in fingerprint image. $C_{out} = \{c_1, c_2, c_3, c_4, c_5\}$ denotes the BPN classifier output. Assume c_k and c_l are the maximum and second maximum among $c_i, i = 1, 2, ..., 5$, respectively. In addition, the number of CPs is denoted as N_C and the number of DPs as N_D. Accordingly, the classification decision can be made with following rules:

(1) If c_k meets $c_k > T_1$ and $c_k - c_l > T_2$, the fingerprint is classified into class k; or go to (2). T_1 and T_2 are predefined threshold. In this paper, $T_1 = 0.6$ and $T_2 = 0.2$.

(2) Else, if $Nc = 0$ and $N_D = 0$, compare c_1 and c_5: If $c_1 > c_5$ then the fingerprint is classified into Arch; else it is classified into Tented Arch.

(3) Else, if $Nc = 2$ and $N_D = 2$, the fingerprint is classified into Whorl

(4) Else, if $Nc = 1$ and $N_D = 1$ and $c_k < T_3$, calculate the distance between core point and delta point in horizontal and vertical direction respectively and denote these 2 values as d_x and d_y. Compare d_x and d_y:

   (a) If $d_x < d_y \times T_4$, assign the fingerprint to Tented Arch

   (b) Else, compare c_2 and c_3: If $c_2 > c_3$ then the fingerprint is classified into Left Loop; else it is classified into Right Loop.
   Here, T_3 and T_4 are predefined threshold. In this paper, $T_3 = 0.3$ and $T_4 = 0.5$

(5) Else, if $Nc = 1$, compare c_2 and c_3: If $c_2 > c_3$ then the fingerprint is classified into Left Loop; else it is classified into Right Loop.

(6) Else, assign the fingerprint to class k

3   Experimental Results

The proposed fingerprint algorithm has been validated on the NIST Special Database 4 [22]. The NIST-4 database consists of 4000 fingerprint images from 2000 fingers with 2 impressions for each finger. Each image is analyzed by human experts and labeled with one or more of the five classes Arch (A), Left Loop (L), Right Loop (R), Whorl (W) and Tented Arch (T). For 17.5% of 4000 fingerprints, two labels are assigned due to a variety of ambiguities. In our experiments, a fingerprint is considered as belonging to both classes when it is labeled to multiple classes.

In our experimental condition, 2000 fingerprint images from 1000 fingers with 2 impressions were selected as the train set to train the BPN classifier and all the 4000 fingerprint images of NIST-4 were selected as the test set. The classification accuracy with BPN classifier alone achieved 91.63% on train set and 91.43% on test set. With the proposed method, which is based on integrate analysis of BPN classifier output and singular points information, accuracy of 93.23% was reached. Table 1 shows the classification results on test set. In our method, no image was rejected, in both training stage and test stage. If the class label assigned with our proposed method for the test image matches one of the multiple labels, the classification of this image is considered correct. In our test environment (P4 1.7G, 512M DDR), it takes 0.7s for our system with proposed method to classify one fingerprint image.

Table 1. Confusion matrix of the classification results on all 4000 images of NIST-4 database

	A	L	R	W	T
A	861	4	5	4	10
L	22	710	15	18	11
R	18	20	744	6	16
W	7	5	3	783	0
T	68	12	15	12	631

4   Conclusion

In this paper, we proposed a novel method for the fingerprint classification based on statistical features combined with information of singular points. Statistical features based on gray-level and binarized image blocks are extracted and applied to fingerprint classification. Compared with other filter-based methods, statistical features based on gray-level and binarized image blocks in the proposed method are easily and quickly extracted. Experiment on NIST-4 database shows that these features are effective for fingerprint classification. Combining statistical features with information of singular points further improves the performance of the method. Table 2 gives an accuracy comparison of various state-of-the-art fingerprint classification algorithms on NIST-4 database which are reported in literature. The comparison results reveal that the proposed method has a very good performance.

Table 2. A comparison of various fingerprint classification algorithms on NIST-4 database

Algorithm	5-class accuracy%(reject rate%)
K. Karu et al.[4]	85.5
L. Hong et al [5]	87.2
C.L.Wilson et al.[6]	90.2 (10)
R.Cappelli [23]	92.2
C.H.Park et al.[15]	90.7
Y.Yao et al [13]	90 (1.8)
S.Shah et al. [10]	92.8 (10)
Proposed	**93.23%**

According to the analysis of the wrongly classified cases, we found that some errors were caused by incorrect reference point location. The algorithm of reference point needs to be further improved.

References

1. A. K. Jain, L. Hong, and S. P. Bolle: An identity-authentication system using fingerprints. Proc. IEEE, vol. 85 (1997) 1365-1388
2. E.R Henry: Classification and Uses of Fingerprints. Routledge: London, 1900.
3. C.V. K. Rao, K. Black: Type Classification of Fingerprints: A Syntactic Approach. IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol.2 (3) (1980) 223-231
4. K. Karu, A.K. Jain: Fingerprint Classification. Pattern Recognition, Vol. 29 (3) (1996) 389-404
5. L. Hong, A.K. Jain: Classification of Fingerprint Images. in: 11th Scandinavian Conference on Image Analysis, Kangerlussuaq, Greenland, 1999.
6. C. L. Wilson, G.T. Candela, C.I. Watson: Neural Network Fingerprint Classification. Journal of Artificial Neural Networks, Vol. 1 (2) (1993) 203-228
7. G.T. Candela, P.J. Grother, C.I. Watson, et al: PCASYS―A Pattern-Level Classification Automation System for Fingerprints. NIST Technical Report NISTIR 5647, Aug. 1995.
8. A. Senior: A Hidden Markov Model Fingerprint Classifier. Proceedings of 31st Asilomar conference on Signals, Systems and Computers (1997) 306-310
9. M.M.S. Chong, T.H. Ngee, L.Jun et al: Geometric framework for Fingerprint Classification. Pattern Recognition, Vol.30 (9) (1997) 1475-1488

10. S. Shah, P.S. Sastry: Fingerprint Classification Using a Feedback-Based Line Detector. IEEE Trans. On System, Man and Cybernetics, Vol.34 (1) (2004) 85-94
11. J.-H. Chang, K.-C.Fan: A new model for Fingerprint Classification by Ridge Distribution Sequences. Pattern Recognition, Vol.35 (2002) 1209-1223
12. K.A. Nagaty: Fingerprints Classification using Artificial Neural Networks: a Combined Structural and Statistical Approach. Neural Networks, Vol.14 (2001) 1293-1305
13. Y. Yao, G.L. Marcialis, M. Pontil et al: Combining Flat and Structured Representations for Fingerprint Classification with Recursive Neural Networks and Support Vector Machines. Pattern Recognition, Vol. 36 (2003) 397-406
14. A. Fitz, R. Green: Fingerprint Classification Using Hexagonal Fast Fourier Transform. Pattern Recogntion, Vol.29 (10) (1996) 1587-1597
15. C.H.Park, H.Park: Fingerprint Classification using Fast Fourier Transform and Nonlinear Discriminant Analysis. Pattern Recognition, Vol. 38 (2005) 495-503
16. B.G. Sherlock and D.M. Monro: A Model for Interpreting Fingerprint Topology. Pattern Recognition, Vol. 26 (7) (1993) 1047-1055
17. M. Kawagoe, A. Tojo: Fingerprint Pattern Classification. Pattern Recognition, Vol.17 (3) (1984) 295-303
18. M. Kass and A. Witkin: Analyzing Oriented Patterns. Computer Vision, Graphics, and Image Processing, Vol 37 (3) (1987) 362-385
19. S.Prabhakar: Fingerprint Classification and Matching Using a Filterbank. Ph.D. Thesis (2001) 96-102
20. X. Sun, Z.-M. Ai: Automatic feature extraction and recognition of fingerprint images. In: 3rd International Conference on Signal Processing, Vol.2 (1996) 1086 − 1089
21. S.E. Fahlman: An Empirical Study of Learning Speed in Back-Propagation Networks. CMU-CS-88-162, 1998
22. C.I. Watson and C.L. Wilson: NIST Special Database 4, Fingerprint Database. US National Institute of Standards and Technology, 1992.
23. R. Cappelli, D. Maio, D. Maltoni: Fingerprint Classification based on Multi-space KL. Proc.of Workshop on Automatic Identification Advances Technologies, (1999) 117-120

An Iterative Algorithm for Fast Iris Detection

Topi Mäenpää

Machine Vision Group, Infotech Oulu
University of Oulu, Finland
`topiolli@ee.oulu.fi`

Abstract. Reliable detection of irises is a necessary precondition for any iris-based biometric identification system. It is also important in measuring the movement of eyes in applications such as alertness detection and eye-guided user interfaces. This paper introduces a novel iterative algorithm for the accurate and fast localization of irises in eye images.

1 Introduction

Reliable biometric identification of people has been an important research area for a long time. Due to the current political situation and new legislation, it has become increasingly important for example in access and border control. Computer vision techniques have been successfully utilized in fingerprint and iris recognition, and especially fingerprint-based systems are becoming ubiquitous. Even though iris is seen as the most reliable biometric measure, it is still not in everyday use because of the complexity of the systems.

One of the future challenges in the development of iris recognition systems is their incorporation into devices such as personal computers, mobile phones and embedded devices. In such applications, the computational complexity and noise tolerance of the recognition algorithm play an important role. This paper addresses the computational complexity of the necessary first step in iris recognition, namely iris detection.

The typical way of detecting the iris is to first detect the pupil and utilize the information in finding the iris. Even though a successful detection of the pupil constraints the search for the iris a lot, three independent parameters still need to be found. First, the concentricity of the pupil and the iris cannot be assumed. Therefore, the coordinates of the center of the iris must be found. The radius of the iris becomes the third free parameter. With transform-based techniques, such as the Hough transform, a three-dimensional transform domain must be constructed [10]. This also applies to the integrodifferential operator introduced by Daugman [3]. Some kind of coarse-to-fine strategy must be employed or the transform domain must be limited as in [4, 7] to achieve one-pixel accuracy within reasonable time. Furthermore, reliable edge detection is difficult to implement because the outer boundary of the iris tends to be extremely soft in near infra-red illumination commonly used in iris recognition.

Despite the difficulties, transform-based techniques have been applied to iris detection in recognition [8, 9]. Even though they can sometimes be quite efficiently implemented, a novel iterative approach is introduced to overcome the

afroementioned difficulties. The algorithm is based on the fact that relatively accurate initial guesses for the unknown parameters can be obtained based on the location of the pupil. Thus, an iterative algorithm only needs to run a few rounds to converge to the solution. Another advantage of the algorithm is that eyelid detection is incorporated into the detection of the iris. This either makes the task of an eyelid detector, such as the one introduced in [11], easier or renders it unnecessary.

2   Detecting the Pupil

In the literature, multiple ways of detecting the pupil have been presented. Some use edge information [2] whereas others utilize thresholding [8]. Since the problem is trivial, most methods work well enough. In this work, a simple yet effient way that utilizes global thresholding is used. In the CASIA database used in the experiments, the method works well, but may need to be replaced with another or enhanced to remove specular higlights if such artifacts exists in images.

In an image containing only an eye and some of its surroundings, the gray level of the pupil is close to the darkest pixel in the image. A gray level that is certainly above that of any of the pixels in the pupil area is selected as a threshold value. The thresholded binary image is inverted and processed with morphological opening to remove any elements smaller than the pupil. Let $(x_i \ y_i), i = 1 \ldots N$ denote the image-domain coordinates of the remaining pixels. The center (x_c, y_c) of the pupil is obtained by calculating the center of mass of the pixels:

$$(x_c \ y_c) = \frac{1}{N} \sum_{i=1}^{N} (x_i \ y_i),\qquad(1)$$

where N is the total number of non-zero pixels. The radius r is directly obtained by recalling that the number of pixels is equal to the area of the circular pupil. Thus, $r = \sqrt{N/\pi}$.

3   Detecting the Iris

The proposed algorithm remotely resembles active contour models (snakes) [5], but the constraints imposed on the boundary curve are treated in a different way. There is no "internal energy" that restricts the shape of the boundary. Instead, a circular boundary is assumed.

The algorithm is initialized with a circle whose center is at the center of the pupil. The radius of the circle is chosen to be "somewhat larger" than that of the pupil. The exact value affects practically nothing but the speed of convergence. The idea is to impose a force to selected points on the circle. The magnitude and the direction of the force are determined by two properties of the iris and its outer boundary:

– The iris is (on average) darker than the sclera.
– The transition from dark to light is steepest on the exact location of the boundary.

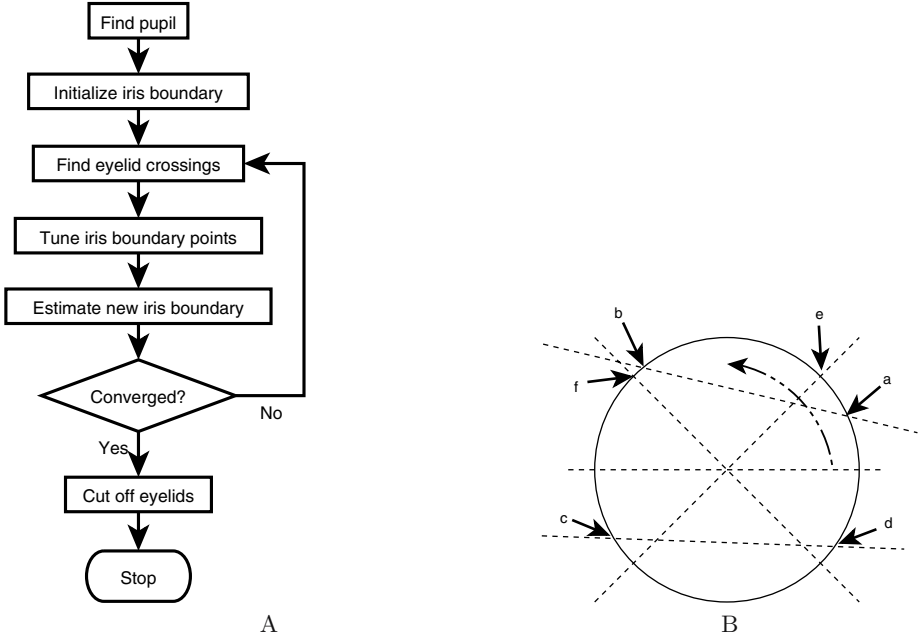A flow chart representation of the algorithm is shown in Fig. 1 A.

A

B

Fig. 1. (A) The iris detection algorithm as a flow chart. (B) Significant points on the iris boundary

Each iteration of the algorithm starts with N points initialized on a circular boundary curve. These points are denoted by $(x_i\ y_i)$, where $i = 0 \ldots N-1$. $N/2$ points are uniformly distributed on both sides of the center of the circle within a 90° sector:

$$(x_i\ y_i) = (x_c\ y_c) + r(\cos\alpha\ \sin\alpha), \tag{2}$$

where $(x_c\ y_c)$ and r denote the center and the radius of the circle and

$$\begin{aligned}\alpha &= -\tfrac{\pi}{4} + \tfrac{i\pi}{2(N/2-1)},\ i = 0 \ldots N/2 - 1 \\ &= \tfrac{3\pi}{4} + \tfrac{(i-N/2)\pi}{2(N/2-1)},\ \ i = N/2 \ldots N-1.\end{aligned} \tag{3}$$

Angles are measured couter-clockwise starting at the positive x axis direction. In fact, the full 90° sector is not necessarily used due to eyelids. Eyelid handling is explained in Sect. 4.

Once the points have been initialized, a force $\boldsymbol{f}$ is independently applied to each:

$$(x_i\ y_i) \leftarrow (x_i\ y_i) + \boldsymbol{f}_i. \tag{4}$$

The force is composed of two components, whose purpose is to drive the boundary 1) to the edge of the dark and bright area ($\boldsymbol{f}_{i,1}$) 2) to the location of the steepest transition ($\boldsymbol{f}_{i,2}$). Each of these components is weighted by a learning constant:

$$\boldsymbol{f}_i = \mu_1 \boldsymbol{f}_{i,1} + \mu_2 \boldsymbol{f}_{i,2}. \tag{5}$$

The learning constants are updated so that greater emphasis is put on $\boldsymbol{f}_1$ in the beginning. The first component, $\boldsymbol{f}_1$ drives the boundary to dark areas by utilizing the gradient $\boldsymbol{G} = \nabla \boldsymbol{I}_s$ of a smoothed image $\boldsymbol{I}_s = \boldsymbol{K} * \boldsymbol{I}$ (s for *smooth*), where $\boldsymbol{I}$ denotes the two-dimensional image domain and $\boldsymbol{K}$ is a Gaussian convolution kernel. The gradient in horizontal and vertical direction is estimated with Sobel masks. The equation for $\boldsymbol{f}_{i,1}$ is simply:

$$\boldsymbol{f}_{i,1} = -\boldsymbol{G}(x_i, y_i). \tag{6}$$

The second component, $\boldsymbol{f}_2$ moves the boundary towards the steepest gradient:

$$\boldsymbol{f}_{i,2} = (\nabla|\boldsymbol{G}|)(x_i, y_i). \tag{7}$$

Once the force has been applied to all points, a new location for the center of the iris is estimated. The equation of a circle is used as a starting point in building a linear estimation model:

$$\sqrt{(x_c - x_i)^2 + (y_c - y_i)^2} = r \tag{8}$$

$$(x_c - x_i)^2 + (y_c - y_i)^2 = r^2 \tag{9}$$

$$x_i^2 + y_i^2 = r^2 - x^2 - y^2 + 2x_i x_c + 2y_i y_c \tag{10}$$

The linear estimation model for $(x_c \; y_c)$ becomes:

$$\begin{pmatrix} x_0^2 + y_0^2 \\ \vdots \\ x_{N-1}^2 + y_{N-1}^2 \end{pmatrix} = \begin{pmatrix} 2x_0 & 2y_0 & 1 & -1 \\ \vdots & \vdots & \vdots & \vdots \\ 2x_{N-1} & 2y_{N-1} & 1 & -1 \end{pmatrix} \begin{pmatrix} x_c \\ y_c \\ r^2 \\ x_c^2 + y_c^2 \end{pmatrix} \tag{11}$$

The terms r^2 and $x_c^2 + y_c^2$ are non-linear, but cannot be removed from the estimation. They are just forgotten in the result. Eq. 11 follows the standard pattern of linear estimation problems:

$$\boldsymbol{Z} = \boldsymbol{H}\boldsymbol{\Theta}, \tag{12}$$

where $\boldsymbol{Z}$, $\boldsymbol{H}$ and $\boldsymbol{\Theta}$ are matrices that correspond to the terms in Eq. 11. An LS estimate of $\boldsymbol{\Theta}$ is obtained by using the Moore-Penrose pseudoinverse [6]:

$$\widehat{\boldsymbol{\Theta}} = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{Z}. \tag{13}$$

In $\widehat{\boldsymbol{\Theta}}$, the two topmost rows contain a linear estimate of the iris' center, denoted by $(\widehat{x}_c \; \widehat{y}_c)$. The radius is estimated by finding an estimate $\widehat{r}$ that minimizes the mean square error distance between the boundary of the circle and the N points. It is simply the mean distance from the estimated center to the measured boundary points:

$$\widehat{r} = \frac{1}{N} \sum_{i=1}^{N} \sqrt{(\widehat{x}_c - x_i)^2 + (\widehat{y}_c - y_i)^2}. \tag{14}$$

The parameters of the circle are finally updated with the new estimates:

$$\begin{aligned} (x_c \; y_c) &\leftarrow (\widehat{x}_c \; \widehat{y}_c) \\ r &\leftarrow \widehat{r} \end{aligned} \tag{15}$$

The algorithm is stopped once the change on both the radius and the center is less than one pixel. Since convergence cannot be guaranteed in the general case, a predefined number of iterations is used as a second stopping criterion.

4   Detecting Eyelids

Eyelid detection is incorporated into the iris finding algorithm so that pixels on the eyelids are ignored. The locations of eyelid boundaries are estimated on each iteration. This approach has the advantage that no separate algorithm is needed in finding the eyelid. Furthermore, a more accurate esimate of the iris location can be obtained by limiting the search to the sclera.

To detect the eyelid, one only needs to inspect the gray profile of the current estimate of the iris boundary. The gray profile is obtained from the smoothed image:

$$g(\gamma) = \boldsymbol{I}_s(x_c + r\cos(\gamma), y_c + r\sin(\gamma)), \qquad (16)$$

where $\gamma \in [0, 2\pi)$ is the rotation angle, and $\boldsymbol{I}_s(x, y)$ represents the gray level at (x, y). Eyelids boundaries are detected by utilizing the derivative of a Gaussian-smoothed gray profile $g_s = k * g$, where k represents a one-dimensional Gaussian kernel. The profile is relatively smooth on the iris or the sclera because it cannot steeply cross the boundary of the iris. If the profile crosses the boundary of either eyelid, a rapid change is seen in g_s. The exact location of the eyelid boundary is assumed to be at the point of steepest descent of ascent.

Fig. 1 B shows a schematic picture of the iris boundary. Since γ is measured counter-clockwise as shown by the curved arrow, the rightmost crossing of the upper eyelid (a) is found first in $g(\gamma)$. To qualify as an eyelid crossing, $g_s(\gamma)$ must satisfy the following conditions:

1. $\frac{d}{d\gamma} g_s(\gamma)$ must be greater than a threshold, whose value is experimentally found.
2. $\frac{d^2}{d\gamma^2} g_s(\gamma)$ must be zero. This ensures a local maximum/minimum.
3. $\frac{d}{d\gamma} g_s(\gamma)$ must be positive for points a and c, and negative for points b and d. This is because eyelids cause a shadow on the sclera.

Once the iris boundary has converged to its final state, the last estimates of the iris boundaries can be used in removing the eyelids. A simple way of doing this is to cut off the circle segment bounded by the boundary points (e.g. the segment between a and b in Fig. 1 B). Another option is to collect all estimated eyelid boundary points $(x_i \; y_i)$, $i = 0 \ldots M - 1$ found during the iteration. Two sets are collected, one for each eyelid. Finally, a linear estimation problem is built to fit a parabola to the points in a similar manner as in Eq. 11:

$$\begin{pmatrix} y_1 \\ \vdots \\ y_{M-1} \end{pmatrix} = \begin{pmatrix} x_0^2 & x_0 & 1 \\ \vdots & \vdots & \vdots \\ x_{M-1}^2 & x_{M-1} & 1 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix}, \qquad (17)$$
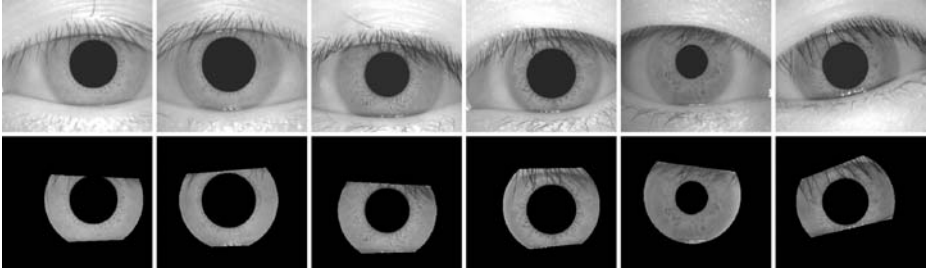
Fig. 2. Experimental results. The top row shows eye images and the bottom row the irises extracted from the images

where a, b, and c are the parameters of a parabola described by $y = ax^2 + bx + c$. The solution is analogous to Eq. 13.

The former of these two methods is faster, but also more sensitive to measurement noise. Another disadvantage is that whenever the eyelid is highly curved, cutting off a segment of the iris wastes information useful for recognition. In many cases, however, this segment is covered by eyelashes anyway.

5   Experimental Results

To evaluate the algorithm, experiments were performed on the 108 different irises of the CASIA iris image database collected by Institute of Automation, Chinese Academy of Sciences [1]. Since there is no ground truth to compare against, it is difficult to give a quantitative figure of the accuracy of the algorithm. It however seems that correct results are obtained in almost all cases, even when the border between the iris and the sclera is faint, or when the iris is occluded by eyelids.

Fig. 2 shows six images of eyes and the corresponding irises. Since the algorithm works faultlessly with all cases in which the iris not significantly occluded or has a prominent boundary, the figure shows only more challenging problems. Images 1-3 show successful detections with faint iris borders and eyelid occlusion. The three last ones show some failures. In images 4 and 5, the boundary of the upper eyelid is detected incorrectly, even though the parameters of the iris are correct. In these examples, the steepest change in the circular gray profile is not at the exact location of the eyelid boundary. In image 6, the very smooth transition from iris to sclera causes a misalignment of the estimated boundary. It remains to be seen how these errors affect identification accuracy.

The algorithm converges very fast. In 59% of cases, only three iterations or less are needed. 93% of the irises are found with six iterations or less. The most difficult cases (2) take 11 iterations. On average, 3.4 iterations are needed.

6   Performance Considerations

The proposed algorithm has a number of parameters that affect its computational performance. Some of them also affect the accuracy. The optimal trade-off may depend on application, but the following matters should be considered:

- Initial guess for the iris boundary. The better the guess the faster the convergence.
- Number of samples (N) used in tuning the iris boundary. This value is affected by image resolution. The theoretical lower bound is three.
- Width of the sector (α) the samples are placed on. Not the whole circumfence of the iris needs to be taken into account. For example, in Fig. 1 B, everything in the sector between e and f is ignored.
- Eyelid detection. Since circle parameters of the iris boundary change relatively little in the final phase, eyelid detection can be sometimes skipped. The sampling frequency of the gray profile $g(\alpha)$ can also be reduced.
- Learning constants. A large learning constant provides fast reaction but also increases noise sensitivity. The magnitude of the learning constants is also affected by the type of the gradient estimator.

The algorithm requires three matrix multiplications and an inversion of a symmetric 4-by-4 matrix on each iteration. Alternatively, these can be replaced with QR decomposition, after which one matrix-vector multiplication followed by back-substitution must be performed. Furthermore, a few squares and a square root are needed for the estimation of the iris center. For maximum computational performance, the calculations should be implemented with integer arithmetic. In addition to the arithmetic performed on each iteration, the input image needs to be convolved with a Gaussian mask and two edge detection masks once in the beginning. For eyelid detection, one-dimensional convolution is needed. Since there is no transform domain to be built, the memory requirements are quite small.

7   Discussion and Conclusions

The goal of the author is to develop algorithms that can be used with cheap hardware so that iris recognition could one day be easily applicable to any application. As a first step, a novel iterative algorithm for the fast detection of irises was introduced. The algorithm gives faultless results for almost all of the iris images in the CASIA database and converges to the solution in a few iterations.

As a next step, the performance of the algorithm must be measured in a recognition application. Furthermore, the algorithm needs to be evaluated with images containing artifacts such as specular highlights and eyeglasses.

Acknowledgement

References

1. Casia iris image database. http://www.sinobiometrics.com.
2. W.W. Boles and B. Boashash. A human identification technique using images of the iris and wavelet transform. *IEEE Transactions on Signal Processing*, 46(4):1185–1188, 1998.
3. J. Daugman. How iris recognition works. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):21–30, 2004.
4. J. Huang, Y. Wang, T. Tan, and J. Cui. A new iris segmentation method for recognition. In *19th International Conference on Pattern Recognition*, volume 3, pages 554–557, 2004.
5. M. Kass, A. Witkin, and D. Terzopoulo. Snakes: active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1987.
6. C. Lawson and R. Hanson. *Solving Least Squares Problems*. Prentice-Hall, 1974.
7. L. Ma, T. Tan, Y. Wang, and D. Zhang. Personal identification based on iris texture analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1519–1533, 2003.
8. L. Ma, T. Tan, Y. Wang, and D. Zhang. Efficient iris recognition by characterizing key local variations. *IEEE Transactions on Image Processing*, 13(6):739–750, 2004.
9. M. Vatsa, R. Singh, and P. Gupta. Comparison of iris recognition algorithms. In *International Conference on Intelligent Sensing and Information Processing*, pages 354–358, Chennai, India, 2004.
10. R. Wildes, J. Asmuth, G. Green, S. Hsu, R. Kolczynski, J. Matey, and S. McBride. A machine-vision system for iris recognition. *Machine Vision and Applications*, 9:1–8, 1996.
11. Y. Wu, H. Liu, and H. Zha. A new method of detecting human eyelids based on deformable templates. In *Systems, Man and Cybernetics*, volume 1, pages 604–609, 2004.

A Non-linear Normalization Model for Iris Recognition

Xiaoyan Yuan and Pengfei Shi

Institute of Image Processing and Pattern Recognition,
Shanghai Jiaotong University, Shanghai, P.R. China, 200030
{yanr,pfshi}@sjtu.edu.cn

Abstract. Iris-based biometric recognition outperforms other biometric methods in terms of accuracy. In this paper an iris normalization model for iris recognition is proposed, which combines linear and non-linear methods to unwrap the iris region. First, non-linearly transform all iris patterns to a reference annular zone with a predefined λ, which is the ratio of the radii of inner and outer boundaries of the iris. Then linearly unwrap this reference annular zone to a fix-sized rectangle block for subsequence processing. Our iris normalization model is illuminated by the 'minimum-wear-and-tear' meshwork of the iris and it is simplified for iris recognition. This model explicitly shows the non-linear property of iris deformation when pupil size changes. And experiments show that it does better than the over-simplified linear normalization model and will improve the iris recognition performance.

1 Introduction

People put more and more attention on biometrics because of the urgent requirements to solve the security problem in this information society. Instead of the traditional token-based method (key, card etc.) and knowledge-based method (password etc.), biometrics identify people by what they 'are'. Commonly used biometric features include face, fingerprints, voice, iris, gait, palm-print, hand geometry, etc. Among them, iris-based biometric outperforms other biometric methods in terms of accuracy because its richness of unique and time-invariant features. Moreover, the ceaselessly pupillary motion of the subject can be measured to detect counterfeit.

The most famous and widely-used iris recognition system is developed by J. Daugman [1] [2]. First, he employs Integro-differential operator to localize the iris region, and then normalize this region with linear 'rubber-sheet' model. Complex 2-D Gabor wavelets are constructs to extract iris features and a 0-1 bit sequence, or Iris-Code, is generated for any given iris. Finally, Hamming distance is used as the matching measure.

There are also many other famous iris recognition methods, such as Wildes' [3], Boles' [4], Ma's [5] [6] etc. They all yield very good recognition performances.

In this paper, we focus on the normalization method for iris recognition. Section 2 introduces the related work on iris deformation and normalization method. Section 3 is the detailed introduction of the proposed non-linear normalization model. Some experiments results are presented in section 4. Finally, conclusions are drawn in section 5.

2   Related Work on Iris Deformation and Normalization

Normalization means to compensate for the elastic deformation of iris texture, such as translation, overall scaling and angular scaling (pupil size changes) etc. to make the iris pattern to be 'aligned' and 'comparable'.

In Daugman's normalization method, the deforming of the meshwork of the iris is modeled as the linear stretching of a homogenous 'rubber sheet'. Thus, each point in the iris is remapped in a 'double dimensionless projected polar coordinate system' [1] [2] with:

$$I\big(x(r,\theta), y(r,\theta)\big) \to I(r,\theta) \tag{1}$$

By this means, the annular iris region is unwrapped to a fix-sized rectangle block (shown in Fig. 1) regardless of pupillary change and overall iris image size. This iris normalization model is most frequently used by various researchers.
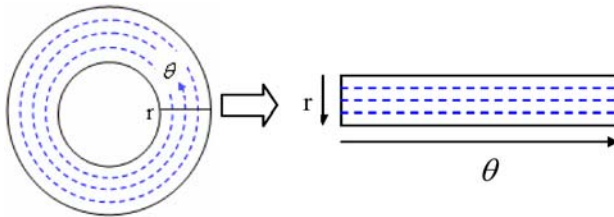
Fig. 1. Daugman's 'rubber-sheet' normalization method. The annular iris zone is 'stretched' to a rectangle block. The dashed lines are sampling circles

Instead of unwrapping the iris region, Wildes [3] applies an image-registration technique to geometrically warp a new iris image into alignment with a selected image in database. Thus, the iris scaling and rotation are compensated. The iris texture deformation caused by pupillary variation is also processed linearly in his method.

In order to analysis the binocular torsion, J. P. Ivins [7] introduces a five-parameter deformable model that can translate (horizontal and vertical eye motion), rotate (torsion), and scale both uniformly and radially (pupil size changes). To normalize the iris pattern for iris-based recognition, this five-parameter deformable model is simplified in [8].

Harry J. Wyatt's work [9] focuses on construction of a meshwork 'skeleton' that can minimize 'wear-and-tear' of iris when pupil size changes. His basic 'skeleton' is a double mesh of collagen 'fibers', which refer to the arcs inside the iris region of Fig. 2. It is a computational model that used to determine the optimal properties of such a mesh. The form of the fiber arcs was optimized across a set of pupil sizes by minimizing strain of the fibers, and constraining each point on a fiber to move only in the radial direction. A logarithmic spiral was used as an initial solution for optimization, and a twenty term polynomial was added to it.

In the viewpoint of physiology, pupil diameter may range from a minimum of about 1.5 mm to a maximum of over 7 mm due to the external variation, such as illumination. The iris is anchored to the globe at its 'root' (the outer circle of the iris), with an outer diameter of about 12 mm. In consequence, iris tissue may vary more than 5 times in circumferential extent (for tissue near the pupil). The structure of the

iris undergoes frequent and substantial deformation. But most iris systems assume that the capture environments are controlled to be invariant and the pupil size change very little. Supposed in the scenario of the distributed structure of iris system as Fig. 3 shows, it is necessary to consider the iris deformation caused by the external variation. And a simple linear stretching of the iris texture seems not enough to catch the complex physical nature of iris deformation. More research should be done on it.
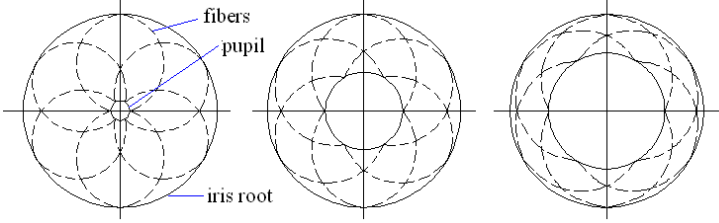
Fig. 2. The 'minimum-wear-and-tear' meshwork of iris, and pupil diameter is 1.5, 4.0, 7.0 mm. The dashed arcs are 'fibers', which is the basic structure of the iris
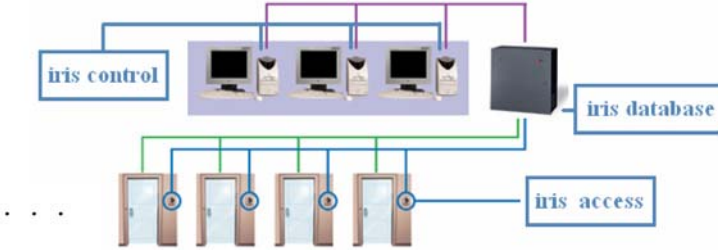
Fig. 3. The distributed structure of iris recognition system. In this scenario, problems caused by the external variation should not be ignored and over-simplified

3   The Propose Non-linear Iris Normalization Method

Illuminated by Harry J.Wyatt's skeleton of the iris [9], we propose our non-linear normalization model for iris recognition. To make the illustration of the normalization model more explicit and also to make things easier, we make such assumptions:

1) The pupil margin and iris root (which correspond to the inner and outer boundaries of the iris) are concentric circles. The radii of the two circles that have been detected are r and R respectively. And the center of the circle is O, which is regarded as the origin of the coordinates in Fig. 4;
2) The margin of the pupil does not rotate significantly during pupil size changes;
3) The pupil shape doesn't change and remain circular when pupil size changes.

   The size of the iris outer boundary is almost unchanged for an eye. Suppose the pupil size is also unchanged, thus $\lambda = r/R$ will be a constant regardless the overall iris size in the captured image. In our research, we first 'make' all pupils zoom to a radius of $r_{ref} = \lambda_{ref} \cdot R$ according to our non-linear iris transform model (λ_{ref} is pre-defined with a priori knowledge, for example, the mean value of all λ). Then this

reference annular zone with λ_{ref} is linearly mapped to a fix-sized rectangle zone of $m \times n$ by equally sampling m points in each virtual concentric sampling circle (the dashed circles in Fig. 4) with fixed radial partition of $\Delta r = (R - r_{ref})/(m-1)$. This is mean to solve the scaling problem caused by distance variation or video zoom problem, and also facilitate the subsequent processing of iris recognition.

Our non-linear iris normalization model is illustrated as Fig. 4. A virtual arc ($arc(PI')$) which connects one point on the pupil margin and another point on the iris root is supposed to exist in the iris region. The polar angle traversed by this arc between these two points is $\pi/2$. This arc is named 'a fiber' following Wyatt's work [9]. When the pupil radius changes from r to r_{ref}, the fiber connects the same two points is moved to the location of $arc(P'I')$.

$arc(PI')$ is part of the circle of:

$$x^2 + (y - y_1)^2 = r_1^2$$

with
$$y_1 = (R^2 - r^2)/2R \tag{2}$$

and
$$r_1 = (R^2 + r^2)/2R$$

And $arc(P'I')$ is part of the circle of:

$$x^2 + (y - y_2)^2 = r_2^2$$

with
$$y_2 = (R^2 - r_{ref}^2)/2R \tag{3}$$

and
$$r_2 = (R^2 + r_{ref}^2)/2R$$

The length of the fiber $arc(PI')$ is:

$$s_1 = r_1 \cdot (\pi - \arcsin(r/r_1)) \tag{4}$$

The length of the fiber $arc(P'I')$ is:

$$s_2 = r_2 \cdot (\pi - \arcsin(r_{ref}/r_2)) \tag{5}$$

The length of the fiber is changed from s_1 to s_2 when pupil radius change from r to r_{ref}, and s_1 is not equal to s_2, that is to indicate that a point in this fiber move not only in radial direction, but also in angular direction. But its angular movement is relatively small as can be shown by (4) (5), so we ignore it and approximate the fiber to be a resilient structure that each point on it moves only in radial direction as pupil size varies.

Suppose A' is on the ith sampling circle. Then A' ($x_{A'}, y_{A'}$) can be solved by:

$$\begin{cases} x^2 + y^2 = (r_{ref} + i \cdot \Delta r)^2 \\ x^2 + (y - y_2)^2 = r_2^2 \end{cases} \tag{6}$$
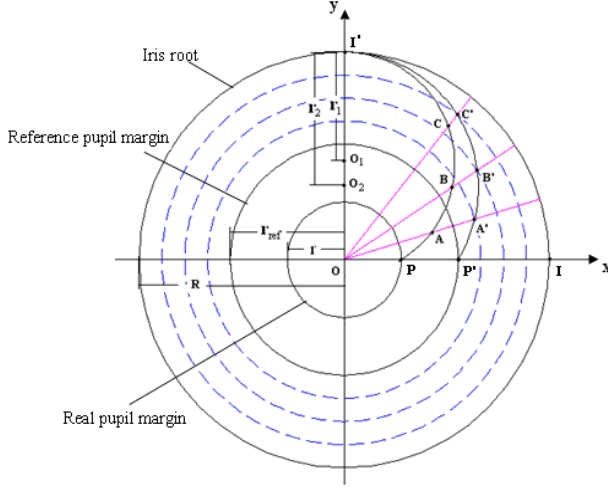
Fig. 4. The proposed normalization model. All iris patterns are first 'made' to deform non-linearly to a reference annular zone with a fixed $\lambda = r/R$. Then, this annular zone is stretched linearly to a fix-sized rectangle block for further processing

Its corresponding point $A(x_A, y_A)$ on the $arc(PI')$ can be solved by:

$$\begin{cases} y - y_{A'} = k\left(x - x_{A'}\right) \\ x^2 + \left(y - y_1\right)^2 = r_1^2 \end{cases} \tag{7}$$

with $k = y_A/x_{A'}$.

If there are 'parallel' arcs, that is a single such arc as $arc(P'I')$ in Fig. 4 replicated at a series of polar position around the pupil center, the ith sampling circle in the reference annual zone corresponds to a concentric circle that passes A in the original iris region. By this means, the linearly radial sampling of the reference annular zone corresponds to the non-linear sampling of the original iris region.

4   Experiments

The recognition results depend much on the dataset used. Most researches are tested on the somewhat 'ideal' iris images captured in controlled conditions. But in practical applications, the external conditions are more casual, so there are still many problems should be considered. Iris deformation due to pupil size variation is one of them.

The dataset we applied to test our method contains 1000 iris images of 200 classes (SJTU-IDB v2: Iris Database of Shanghai JiaoTong University, version 2), which are captured with our sensor specifically designed to capture iris images. And the images are captured in different times under different conditions (different locations, day, night, light on, light off, curtain up, curtain down etc.) from people of various occupations and ages. Some iris samples change much due to different external conditions, shown as fig. 5. Pupil size variation is indicated by $\lambda = r/R$. The value of λ_{ref} is 0.4505 in our experiments.
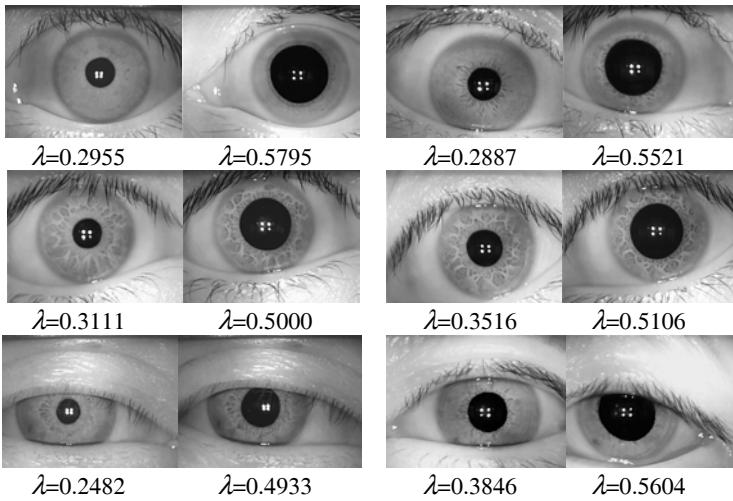
Fig. 5. Some samples in our dataset. Pupil size variation is indicated by λ shown under the iris image, which is the ratio of the radii of inner and outer circles of the iris

To compare the results of linear normalization method and the proposed non-linear normalization methods, the distribution of the within-class matching distance is estimated with 4000 comparisons and the distribution of between-class matching distance is estimated with 995000 comparisons using this two normalization methods. Besides the normalization, other steps of the iris recognition procedure are all same, that is to localize the iris region with Integro-differential operator, encode it with complex Gabor filters, and with Hamming distance as the matching measure. It should be noted that only iris textures in two rectangular 'wedges' of right and left sides instead of the whole annular zone are processed for recognition to avoid the false iris information caused by occlusion of eyelids and eyelashes. It can be seen that the latter has a better discriminability.

5   Conclusion

In this paper an iris normalization method is proposed which combines the linear and non-linear method to unwrap the iris region. First, non-linearly transform all iris image to a reference annular zone with a predefined λ, which is the ratio of the radii of inner and outer boundaries of the iris. Then linearly unwrap this reference annular zone to a fix-sized rectangle block for subsequence processing.

This model still simplifies the real physiological mechanism of iris deformation for efficiency purpose in practical application and some assumptions and approximations are still needed to be made to support this model. But our iris normalization model explicit show the non-linear property of iris deformation. And experiments show that it did better than the over-simplified linear normalization model and will improve the iris recognition performance. More effort should be made to make a more precise iris deformation model based on physiological properties of the iris to further improve the iris recognition performance in our future work.
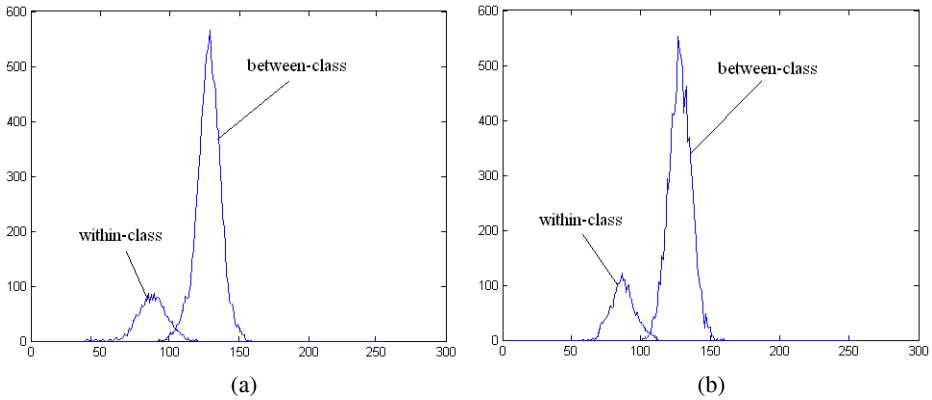
Fig. 6. The distributions of within-class and between-class matching distances with linear normalization (a) and the proposed normalization model (b). It can be seen that the latter has a better discriminability

Acknowledgement

References

1. Daugman J.: How Iris Recognition Works. IEEE Transactions on Circuits and Systems for Video Technology. 14 (2004) 21- 30
2. Daugman J.: High confidence personal identification by rapid video analysis of iris texture. Proceedings of the IEEE, International Carnahan conference on security technology (1992)
3. Wildes R.: Iris Recognition: An Emerging Biometric Technology. Proceedings of the IEEE, vol.85 (1997)
4. W.W. Boles and B. Boashash.: A Human Identification Technique Using Images of the Iris and Wavelet Transform. IEEE Transaction on Signal Processing. 46 (1998)
5. Li Ma, Tieniu Tan, Yunhong Wang, Dexin Zhang: Personal Identification Based on Iris Texture Analysis. IEEE Transaction on Pattern Analysis and Machine Intelligence. 25 (2003) 1519-1533
6. Li Ma, Tieniu Tan, Yunhong Wang, Dexin Zhang: Efficient Iris Recognition by Characterizing Key Local Variations. IEEE Transaction on Image Processing. 13 (2004) 739-750
7. Ivins J. P, Porrill J., Frisby J. P.: A Deformable Model of the Human Iris Driven by Non-linear Least-squares Minimization. In: The 6th International Conference on Image Processing and its Applications Dublin, Ireland. (1997) 234-238
8. Ming Xing, Xu Tao, Wang Zheng-xuan: Using Multi-matching System Based on a Simplified Deformable Model of the Human Iris for Iris Recognition. Journal of Bionics Engineering. 1 (2004) 183-190
9. Harry J. Wyatt: A 'Minimum-wear-and-tear' Meshwork for the Iris. In: Vision Research. 40 (2000) 2167-2176

A New Feature Extraction Method
Using the ICA Filters
for Iris Recognition System

Seung-In Noh[1], Kwanghyuk Bae[1], Kang Ryoung Park[2], and Jaihie Kim[1]

[1] Department of Electrical and Electronic Engineering, Yonsei University
Biometrics Engineering Research Center, Seoul, Korea
{sinoh,paero,jhkim}@yonsei.ac.kr
[2] Division of Media Technology, Sangmyung University, Seoul, Korea
parkgr@smu.ac.kr

Abstract. In this paper, we propose a new feature extraction method based on independent component analysis (ICA) for iris recognition, which is known as the most reliable biometric system. We extract iris features using a bank of filters which are selected from the ICA basis functions. The ICA basis functions themselves are sufficient to be used as filter kernels for extracting iris features because they are estimated by training iris signals. Using techniques of the ICA estimation, we generate many kinds of candidates ICA filters. To select the ICA filters for extracting salient features efficiently, we introduce the requirements of the ICA filter. Each ICA filter has a different filter size and a good discrimination power to identify iris pattern. Also, the correlation between bandwidths of the ICA filters is minimized. Experimental results show that the EER of proposed ICA filter bank is better than those of existing methods in both the Yonsei iris database and CASIA iris database.

1 Introduction

An iris is the physiological feature containing unique texture. Also it remains stable and fixed from about six months of age throughout life. An iris recognition system is known as the most reliable system among all biometrics - fingerprint, face, voice, and so on [1]. In general, the iris recognition system consists of four steps [2]: image acquisition, iris localization, feature extraction, and identification. First, an eye image is obtained at a distance from a CCD camera without any physical contact to the apparatus. Second, localizing the inner and outer boundary of the iris is performed. Third, Gabor or wavelet transform is typically used to extract the iris features, and identification or verification is followed. In this paper, we focus on the feature extraction and identification.

This paper presents a new method to extract iris features using independent component analysis (ICA). The concept of the ICA can be seen as a generalization of PCA. While the PCA is optimal transformation in the mean square sense (first- and second-order statistics), the ICA is an unsupervised learning algorithm using high-order statistics. The major characteristic of second-order

statistics techniques like PCA is that they produce spatially global features. They are not adequate to represent locally structured texture. The most common method for generating spatially localized features is to apply ICA to produce basis functions[3, 4].

We propose a method for designing a bank of filters using ICA technique. From the viewpoint of filter design, the ICA basis functions themselves are sufficient to be used as filter kernels for extracting iris features because they are estimated by training iris signals. Thus the ICA filter bank which is composed of the ICA basis functions is able to capture the local structure of iris signal efficiently, and can achieve the good recognition performance.

In Section 2, we will explain the preprocessing methods. In Section 3, the proposed methods will be introduced. Experimental results will be presented in Section 4, and conclusions will be discussed in Section 5.

2   Iris Signal

It is necessary to preprocess a captured eye image for extracting iris features. Preprocessing consists of interior boundary (between pupil and iris) detection and exterior boundary (between iris and sclera) detection using edge detection algorithm[5]. Localized iris region is transformed into a polar coordinate using rubber sheet model[1]. In a normalized iris region at a polar coordinate, we generate iris signals to estimate the ICA basis functions or to extract iris features. Here an iris signal is defined as a one-dimensional iris signature as the following equation:

$$S_{ij}(\theta) = \sum_{r=-M}^{r=M} \frac{1}{\sqrt{2\pi}\sigma} \exp\{-\frac{r^2}{2\sigma^2}\}I(r+i, \theta+j), \quad \theta = 1, 2, \cdots, N \qquad (1)$$

where $I(r, \theta)$ denotes gray information in the polar coordinate image of $P \times Q(96 \times 768)$ in our experiment). (i, j) is the starting position of the small image patch (height$\times$ width is $(2M+1) \times N$) which is converted to an iris signal,
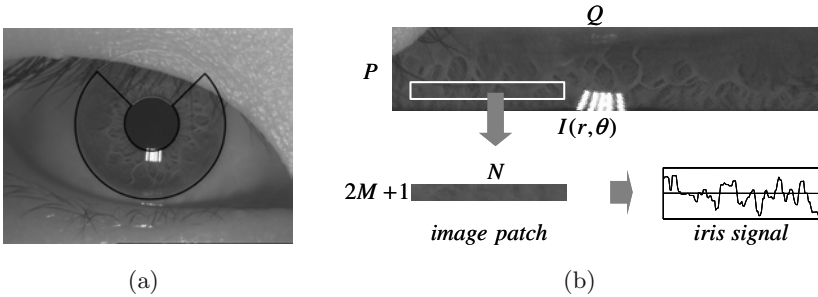

Fig. 1. Generating an iris signal: (a) localized eye image (b) an iris signal extracted from an iris image patch in normalized iris region at a polar coordinate

$S_{ij}(\theta)$. As each one-dimensional column signal of image patch is averaged with the Gaussian function, 2D image patch is converted to 1D iris signal which can be also considered as the N-dimension vector as shown in Fig. 1. Here, σ is determined by evaluating a number of variances and selecting the best candidate [6].

3   Feature Extraction by Filter Bank

3.1   Filter Bank Approaches in Iris Recognition System

In general, a filter bank is a structure that decomposes a signal into a collection of subsignals. Depending on the application, these subsignals (subband signal) help emphasize specific aspects of the original signal or may be easier to work with than the original signal. In an application of pattern recognition, the purpose of filter banks is to extract the discriminant features. In other words, a filtering should be considered with respect to the feature separation for good classification performance, thus does not have to consider the perfect reconstruction.

Here we focus on iris recognition system based on the filter bank approach. A large percentage of existing iris recognition system makes use of the filter banks to extract the local characteristic features. Moreover, most systems almost used heuristically designed filters which were already designed for the applications of image processing or compression [6]. Because these functions are fixed independent of input data, the parameters (e.g. spatial location, orientation, and frequency) of functions should be empirically selected to extract feature vectors. For example, there are the Gabor wavelets [1], the discrete wavelet transforms [7][8], the discrete wavelet frame [9], circular symmetric Gabor filters [10], Laplacian of Gaussian filters [2] and others.

3.2   Iris Feature Extraction by ICA Filters

In this approach, we propose a method for designing a bank of filters using ICA technique. From the viewpoint of filter design, the ICA basis functions themselves are sufficient to be used as filter kernels for extracting iris features because they are estimated by training iris signal as describe in [6]. Namely, the ICA basis functions can capture the inherent characteristics such as local structure and frequency of iris texture. Thus the ICA filter bank which is composed of the ICA basis functions is able to be classified into optimized filters and filter banks. This paper shows that these ICA filters can capture the local structure of iris signal efficiently, and can achieve the good recognition performance.

Here, two problems are considered to design the ICA filters for good recognition performance. The first is which filters are to choose and how many filters are to use. And the second is which features are to use.

Now let's reconsider the process to estimate the ICA basis functions before finding the solution of the first problem. As shown in Fig. 2, training iris signals are put to the preprocessing in which the dimension of iris signal are reduced.
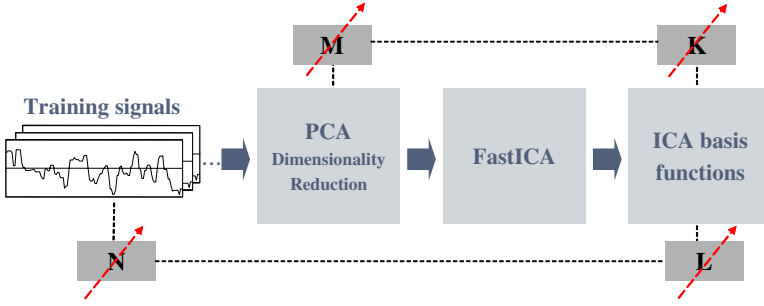
Fig. 2. The flow of ICA estimation

Using the preprocessed signals, the FastICA[3] algorithm estimates the ICA basis functions. Here, we defined that the ICA bases set is a set whose elements are the ICA basis functions learned by FastICA.

Note that the dimension (size) of iris signal is N, the dimension of preprocessed signal is M, the number of ICA basis functions is K, and support (size) of ICA basis function is L. Among four parameters, N (dimension of iris signal) and M (dimension of preprocessed signal) are only adjustable parameters. In contrast, K (the number of ICA basis functions) is same to M (dimension of preprocessed signal) and L (support of ICA basis function) is same to N (dimension of iris signal) at all times. Therefore, K and L are dependent parameters. According to the selection of N (dimension of iris signal) and M (dimension of preprocessed signal), many kinds of the ICA bases set can be generated by FastICA algorithm. Each ICA bases set estimated by the change of N and M is composed of the number of ICA basis functions which have different properties of spatial location and frequency. Namely, there are a lot of candidates for choosing ICA filters. If the degree of freedom of M and N is not limited, it is not possible to find the combination of M and N for designing optimal filter bank. In the case of existing ICA filter bank technique for texture segmentation [4], after M and N is fixed, the ICA basis functions are estimated and filter selection method is introduced.

How Many Filters to Use. In proposed method, two channel filter bank is assumed for performing the comparative experiments fairly since representative methods such as Daugman's system [1] and Li ma et al.'s system [10] use the two channels. In practical application, however, we can easily modify the channel of filter bank using more than two kinds of ICA filters.

Which Filters to Choose. Hereafter, we focus on the problem how to choose two filters from a lot of candidates which can be generated by choosing M and N. The selection of N (dimension of iris signal) means the determination of the ICA filter size. In the application of pattern recognition, the size of filter mask is related to the effective localization range of filtering method when a filtering is viewed as finding the frequency content of a signal locally in time or space.

To achieve the good local frequency analysis, it is necessary to use the wavelet based approaches. Namely, if we want to extract the low frequency information of input signal we allow the use of long finite support of filter, in the other hand, if high frequency information should be analyzed we make a use of short support of filter. We call this approach the optimal time-frequency analysis. To make use of two channels for extracting frequency information effectively, we choose two kinds of N (or L, in our experiments, $L = 32, 64$) so that the ICA filter bank can have the different size of filter mask. The ICA filter with short support must have a discrimination power for relatively high frequency information of an iris signal while the ICA filter with long support must have a discrimination power for relatively low frequency information.

Now, the consideration for choosing filters is limited to the problem to find the appropriate ICA function adequate to the size of filter mask. Because the size of ICA filter is determined, the size of the iris signal is also fixed and the parameter M is only adjustable. The ICA bases sets consist of many kinds of the ICA basis functions as the adjustment of M. All of the ICA functions are the candidates of the ICA filter with size L. To select the optimal ICA filter from the candidate ICA functions, we introduce a criterion which is assessed by the EER when the candidate ICA function is only used as a filter kernel for iris feature extraction. We call this criterion the discrimination power of a ICA function to identify iris signal.

Lastly, we consider the interaction of the ICA filters even though each ICA filter has a good discrimination power. If two ICA filters of ICA filter bank are similar property, it is hard to expect the improvement of recognition performance. Therefore, we need a criterion on the combination of two filters. To make efficient criterion, we make use of a prior knowledge of the iris texture property. It is that the majority of useful information of the iris pattern is in a specific frequency band [1][10]. Based on a prior knowledge, the bandwidths of two filters should be in the specific frequency band. And two filters do not have the same passband in the specific frequency as possible as. In other words, one of filter with short support must have a relatively high frequency passband in the specific frequency band while the ICA filter with long support must have a relatively low frequency passband in the specific frequency band.

In detail, we introduce a cascade method for choosing ICA filters. First, we choose the ICA functions with short support using the criterion of discrimination power. Naturally, its bandwidth is in the relatively high frequency passband. Second, among the candidate ICA functions with long support, we select the best one which satisfies the follow criterion,

$$C(h_i) = \int_{\alpha}^{\beta} |H_i(w)|^2 dw \qquad (2)$$

where h_i is candidate ICA filter with long support, $H_i(w)$ is a magnitude spectrum of h_i, α is a low frequency of passband in the specific frequency band with useful information of iris texture, β is a peak frequency of the passband of the ICA filter with short support. We call this criterion the ability of interactivity

between two filters. Finally, we construct optimal ICA filter bank using two ICA filters.

In brief, the requirements of the ICA filters are as follow. Each ICA filter of filter bank has a different filter size, and a good discrimination power to identify iris pattern. Also, the correlation between bandwidths of the ICA filters is minimized. Following is the overall procedure to make a bank of filters:

1. determine the support of ICA filters ($L = 32, 64$).
2. At $L = 32$.
   - the size of training signal is naturally set to L because of $N = L$.
   - estimate the ICA bases sets as the change of M (dimension of reduced iris signal).
   - evaluate the EERs as the change of the ICA bases set.
   - find the M or the ICA basis set which has the smallest EER.
   - compute the discrimination power of each ICA basis function in the ICA bases set when it is only used as filter kernel.
   - select the ICA basis function of best performance as the ICA filter with size $L = 32$
3. At L=64.
   - the size of training signal is naturally set to L because of $N = L$.
   - estimate the ICA bases sets as the change of M (dimension of reduced iris signal).
   - evaluate the EERs as the change of the ICA bases set.
   - find the candidate ICA basis sets whose EERs are smaller than predefined threshold.
   - compute the discrimination power of each ICA basis function in the ICA bases sets when it is only used as filter kernel
   - select the k-candidate ICA basis functions whose discrimination power are in the k smallest EER.
   - calculate Eq. 2 (the ability of interactivity between two filters) of the candidate k-ICA basis functions
   - select the ICA basis function with maximum value of Eq. 2.

Which Features to Use. After the ICA filter bank is generated by proposed method, an iris feature vector is easily generated by filtering the ROI [6] with the ICA filter bank as following

$$y_i(n) = \sum h_k(l - n)x_i(l) \tag{3}$$

where $x_i(n)$ is the iris signals and $h_k(n)$ is the filters of ICA filter bank. Once iris features are obtained by filtering the input signal in a ROI, we generate the iris code for storing the iris feature compactly and comparing the iris features fast. The encoding method of the iris code is to assign a value 0 or 1 to each element of feature vectors depending on the sign.
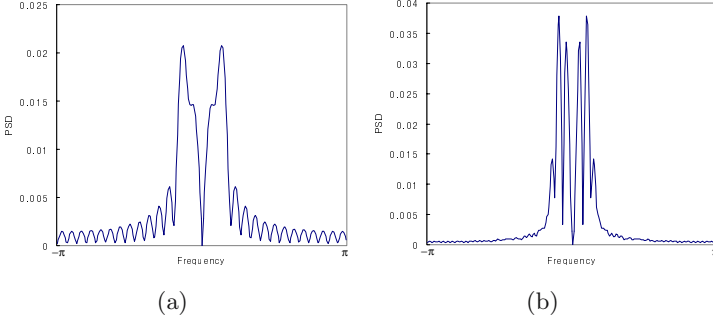
Fig. 3. The power spectral density of ICA filters: (a) a filter at $L = 32$ (b) a filter at $L = 64$

4   Experimental Results

Our algorithm was evaluated on the Yonsei iris database and the CASIA iris database [11]. The Yonsei iris database consists of 990 images with 10 images for each of 99 individuals. The Yonsei iris database samples have a resolution of 640 × 480 and a gray information of 8 bits, which are captured by our iris recognition camera. The iris diameter is approximately 320 pixels. The CASIA iris database contains 756 images from 108 eyes of 80 subjects. The iris images are 8-bit gray images with a resolution of 320 × 280.

As described in the Section 3, we assume two channel filter bank whose filters has a different size (in our experiments, $L = 32, 64$) and then construct the ICA filters of filter bank one by one. Finally, the ICA filter bank can be constructed by selected two filters. Fig. 3 shows the frequency properties of two filters.

Using proposed ICA filter bank, we evaluate the recognition performance. Table. 1 shows the experimental results from the two databases. In both case, recognition performance is better than other filtering based methods such as Gabor wavelet[1] and circular symmetric Gabor filters[10]. This result shows that optimized filter bank by ICA technique is able to improve the recognition performance. Moreover, since, in our experiments, we fixed the many of parameters, i.e., the number of channel, the effective size of filter, if we will study on

Table 1. Comparing the EERs of Gabor wavelets[1], circular symmetric Gabor filters[10], and ICA filter bank

	EER(%)		
	Gabor wavelets	Circular symmetric Gabor filters	ICA filter bank
Yonsei iris database	0.045	0.054	0.038
CASIA iris database	0.290	0.361	0.275

the properties of these parameters more, we make the many kinds of ICA filter banks to improve the recognition performance.

5   Conclusions

A person identification technique using human iris recognition is presented. In this paper, we extract the iris features using a bank of filters which are selected from the ICA basis functions. Experimental results show that the EER of proposed ICA filter bank is better than those of existing methods in both the Yonsei iris database and CASIA iris database. To improve the performance of iris recognition, we will further study on how to extract features efficiently in noisy environments. To be widely used in more reliable biometrics, we will especially consider several conditions like iris image rotation, sensor noise, and poor focus image.

Acknowledgements

References

1. John G. Daugman: High confidence visual recognition of personals by a test of statistical independence. IEEE Transactions on PAMI, vol. 15, no. 11, pp.1148-1160, November 1993.
2. Richard P. Wildes: Iris Recognition: An Emerging Biometric Technology. Proceedings of The IEEE, vol. 85, no. 9, pp.1348-1363, September 1997.
3. A. Hyvärinen, J. Karhunen, and E. Oja: Independent Component Analysis, Wiley, 2001.
4. R. Jenssen and T. Eltoft: Independent Component Analysis for Texture Segmentation, Pattern Recognition, vol.36, no.10, pp.2301-2315, 2003.
5. ChulHan Lee, Seung-In Noh, and Jaihie Kim: The Method of Iris Region Extration for Iris Recognition, The 2003 ITC-CSCC, vol.3, pp.1658-1661, 2003.
6. Kwanghyuk Bae, Seung-In Noh, Jaihie Kim: Iris Feature Extration Using Independent Component Analysis, LNCS, vol.2688, pp.838-844, 2003.
7. W.W.Boles and B.Boashash: A human Identification Technique Using Images of the Iris and Wavelet Transform. IEEE Transactions On Signal Processing, vol. 46, no. 4, pp.1185-1188, April 1998.
8. L.Ma, T. Tan, Y.Wang, and Dexin Zhang: Efficient Iris Recognition by Characterizing Key Local Variations, IEEE Transactions on Image Processing, vol.12, no.6, pp.739-750, 2004.
9. Seung-In Noh, Kwanghyuk Bae, Jaihie Kim: A Novel Method to Extract Features for Iris Recognition System, LNCS, vol.2688, pp.862-868, 2003.
10. L.Ma, T. Tan, Y.Wang, and Dexin Zhang: Personal Identification Based on Iris Texture Analysis, IEEE Transactions on PAMI, vol.25, no.12, pp.1519-1533, 2003.
11. http://www.sinobiometrics.com.

Iris Recognition Against Counterfeit Attack Using Gradient Based Fusion of Multi-spectral Images

Jong Hyun Park and Moon Gi Kang⋆

Biometrics Engineering Research Center
Department of Electrical and Electronic Engineering, Yonsei University
B727 2nd Engineering Hall, 134 Shinchon-Dong, Seodaemun-Gu, Seoul, Korea
{letslove,mkang}@yonsei.ac.kr

Abstract. In this paper, we present an iris recognition system considering counterfeit attacks. The proposed system takes multi-spectral images instead of one infrared iris image. The energy of the multi-spectral images is checked and the authentication is failed if the amount of the energy is not in the proper range. Then the images are normalized and merged into a grayscale image by using a gradient-based image fusion algorithm. In the fusion process, the images considered to be from a counterfeited iris are merged into a poor-quality image which successively generates poor matching score. We show that the proposed scheme successfully maintains the performance of real iris images preventing the counterfeit attacks with experimental results.

1 Introduction

Biometrics identification and recognition systems authenticate a person by analyzing the unique characteristics of human physiological or behavioral biometric traits such as fingerprint, iris, face, hand vain, and speech. Recently, iris recognition has been studied as one of the emerging biometrics [1]-[3]. Since the iris pattern contains much more complex features than the other biometric, the iris recognition technology makes it possible for the user to get reliable performance with relatively vast amount of population. Therefore, it is widely used for border management systems in airports or entrance control of a large company.

Although the authentication system based on biometrics provides higher security and convenience, the system can be still attacked by malign users in various ways [4]-[6]. The ways of the malign attacks can be roughly categorized into counterfeit attacks on the sensor, attacks on the the channels between various system components, and the attacks on the database. The counterfeit attacks refer to that the malign attacker intentionally submits a counterfeited iris pattern to the sensor to deceive the authentication system. In this paper, we focus on rejecting the counterfeited irises during the recognition process. Preventing

---

the counterfeit attacks is usually done by analyzing the liveness of the iris [7]. The liveness of the iris is usually checked by using the biological characteristics of the iris such as the eye and the pupil movement, reaction to the illumination change, and spectral characteristics of the iris image. Another category of the methods is to make it more difficult to counterfeit the iris pattern which can pass the recognition system. A typical example is utilizing multi-modal biometrics.

In this paper, an iris recognition method resistant to the forgery attacks is proposed. It utilizes the iris patterns which vary according to the wavelength of light. It takes multi-band infrared iris images instead of taking one infrared image which is generally used in most of the iris recognition systems. Then, a gradient-based image fusion algorithm is used to merge multi-band iris images into a grayscale iris image. The merging process generates a fused iris image which contains the pattern of the enrolled iris only when it is considered to be genuine. It produces a noisy image if the multi-band images contain the characteristics of the fake iris images. The proposed method, utilizing a couple of iris images on different wavelengths, contains the characteristics of both categories of countermeasures described in the previous paragraph.

This paper is organized as follows. In Section 2, the gradient-based fusion algorithm is briefly reviewed. The proposed algorithm is described in detail in Section 3. The performance of the proposed algorithm is analyzed with the experimental results in Section 4. Finally, we conclude this paper in Section 5.

2   Background

We use a gradient-based image fusion algorithm as the iris image merging process. Gradient-based image fusion is based on the concept of multi-valued edge representation proposed by Zenzo [8]. The differential of a multi-valued image $\boldsymbol{f}$ in the Euclidean space R^N is given by

$$d\boldsymbol{f} = \frac{\partial \boldsymbol{f}}{\partial x} dx + \frac{\partial \boldsymbol{f}}{\partial y} dy. \tag{1}$$

The squared norm of the differential is

$$
\begin{aligned}
&||d\boldsymbol{f}||^2 \\
&= \begin{pmatrix} dx \\ dy \end{pmatrix}^T \begin{pmatrix} ||\frac{\partial \boldsymbol{f}}{\partial x}||^2 & \frac{\partial \boldsymbol{f}}{\partial x} \cdot \frac{\partial \boldsymbol{f}}{\partial y} \\ \frac{\partial \boldsymbol{f}}{\partial x} \cdot \frac{\partial \boldsymbol{f}}{\partial y} & ||\frac{\partial \boldsymbol{f}}{\partial y}||^2 \end{pmatrix} \begin{pmatrix} dx \\ dy \end{pmatrix} \\
&= \begin{pmatrix} dx \\ dy \end{pmatrix}^T \begin{pmatrix} \sum_k \left(\frac{\partial f_k}{\partial x}\right)^2 & \sum_k \frac{\partial f_k}{\partial x} \frac{\partial f_k}{\partial y} \\ \sum_k \frac{\partial f_k}{\partial x} \frac{\partial f_k}{\partial y} & \sum_k \left(\frac{\partial f_k}{\partial y}\right)^2 \end{pmatrix} \begin{pmatrix} dx \\ dy \end{pmatrix} \\
&= v^T G v, \tag{2}
\end{aligned}
$$

which is considered to be the amount of multi-valued variation, or the contrast of the pixel in the direction of v. The 2×2 matrix G is called the first fundamental form. The eigenvectors of G describe the direction of the maximum

or minimum variation of (2). The corresponding eigenvalues refer to the rate of change along the eigenvector directions. However, the eigenvectors don't give the orientations but just the directions of the spectral variation. Actually, there is no general definition of the gradient orientation of a multi-valued image. It is different from the grayscale case in that the direction of increasing intensity is usually understood to be the gradient direction. Thus, the orientation should be determined by some criteria. The easiest example of the criteria is following the gradient orientation of the averaged image [9]. Then, a vector field composed of vectors of which the directions are the same as the major eigenvectors and the magnitudes are the functions of the two corresponding eigenvalues is made through the whole image plane. The fused image which has maximum contrast information of the source images can be obtained by finding the grayscale image of which the gradient is similar to the vector field. We recommend to refer to the literature on the gradient-based image fusion for more detailed information [9]-[11].

3   Iris Recognition Against Counterfeit Attack

Figure 1 shows the schematic of the proposed iris recognition system. At first, multi-spectral iris images are taken by multi-spectral infrared imaging system. Generally, two types of multi-spectral imaging systems are possible. One type is composed of multiple cameras with narrow band infrared optical filters whose wavelengths are different from each other attached in front of the cameras. Another type needs just a camera and a filter or a set of filters whose wavelengths can be automatically changed mechanically or electrically. The former type captures the images simultaneously, while the latter type grabs them sequentially. When the images are captured sequentially, we should set the capture speed so fast that the captured iris images do not need registration. Using multi-spectral images has two major advantages. The first one is that it makes the attackers more difficult to make a fake iris since it should have spectral response very close to that of the original iris. The second one is that we can utilize the abundant information to increase the recognition efficiency or to detect or prevent the counterfeit attacks. Actually, all the biometric systems using multi-modal biometrics hold these advantages in common.

After the multi-spectral iris images are captured, we analyze the energy of each band and reject the enrolled iris if the energy is larger or smaller than a pre-specified threshold. Usually, the response curve of a digital camera is not uniform but has a convex shape. In addition, the pass band gain of each filter is not the same. Therefore, the captured images should be modified according to the pre-specified settings of the digitizer so that the result images can have pertinent brightness and contrast. The digitizer is set to be adapted to real irises, and if simple fake irises are used for the replay attack, all the multi-spectral images can hardly maintain proper brightness and contrast. Therefore, just using multi-spectral iris images can elevate the security of the overall biometric system.
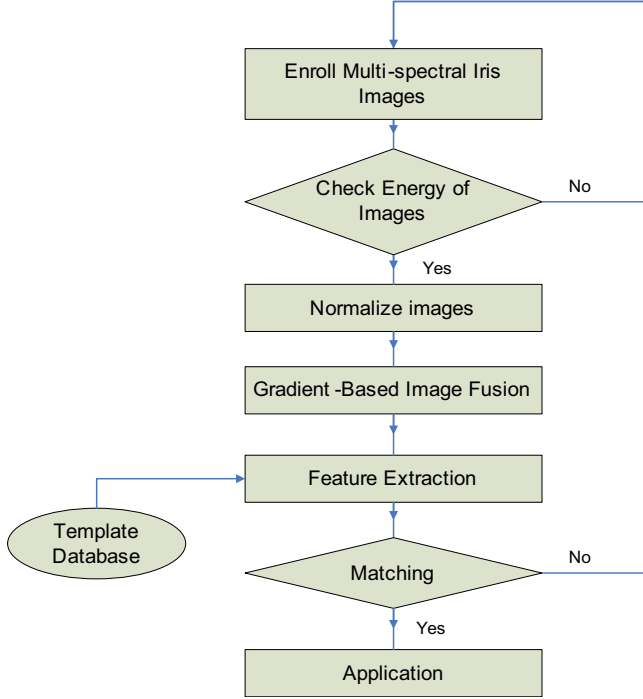
Fig. 1. The block diagram of the proposed iris recognition system

If the iris images are considered to have adequate energy, the images are normalized and fused into a grayscale iris image. The gradient-based image fusion makes a merged image in which the first-order gradient information of the multi-spectral images is maximized. Thus, the various iris features in the multi-spectral images can be effectively integrated in one image. In addition to this advantage, we utilize the phenomenon that the fake iris cannot generate multi-spectral images containing spectral variations in order to reduce the feature information of the fake iris images. Specifically, we use a criterion for gradient direction determination as

$$C = \sum_k w_k f_k Row_k(G) \cdot \hat{v}, \tag{3}$$

where w_k is a weight whose summation over the index k equals to zero, $Row_k(G)$ is k-th row vector of the first fundamental form G, $\hat{v}$ is the initial fused gradient whose orientation is the same as the eigenvector of maximum eigenvalue, and '·' means an inner product. If C is larger than zero, the initial orientation of $\hat{v}$ is maintained. However, if C is smaller than zero, the orientation is changed to the opposite. In case of the iris images from a real person, this function does not make serious confusion to the determination process since the responses of the bands are different from each other. However, in case of the fake iris, the value of C becomes very small overall since f_k values as well as the dot product values

following f_k become similar to each other after the normalization process. Thus, the estimated orientation becomes very unstable and the quality of the fused image becomes very poor.

After the multi-spectral images are fused into one iris image, the authentication is performed using the fused iris image. The authentication process is the same as general iris recognition systems. Note that the iris images in the database should be grayscale images that are fused from multi-band images during the first enrollment. Since the fake iris usually generates a fused result of poor quality, the matching score is far from the acceptance threshold and the attack cannot success.

4   Experimental Results

In this experiment, we used an electronically tunable liquid crystal filter which has the bandwidth of 10 nm (FWHM) and the tunable pass-band range from about 650 nm to 1100 nm. The digital camera used in the experiment has 640 × 480 resolution and has sensitivity enough to sense infrared light in the pass-band range of the filter. The number of spectral bands and the wavelength of each band should be pre-specified. Each wavelength and the interval between two successive bands should be selected experimentally so that the features of one band can be distinguished with those of the other bands and the inner and the outer iris boundaries can be detected easily. Figure 2 shows an example of iris images on four different wavelengths, that is 780, 810, 850, and 900 nm. The overall tone of the iris becomes brighter as the wavelength becomes higher. The radial feature of the region near the iris-pupil boundary is vivid in the low wavelength image, while the furrows near the outer boundary are more prominent in the high wavelength band images. The images of the printed iris cannot show any various characteristics of real iris images described above and just the brightness
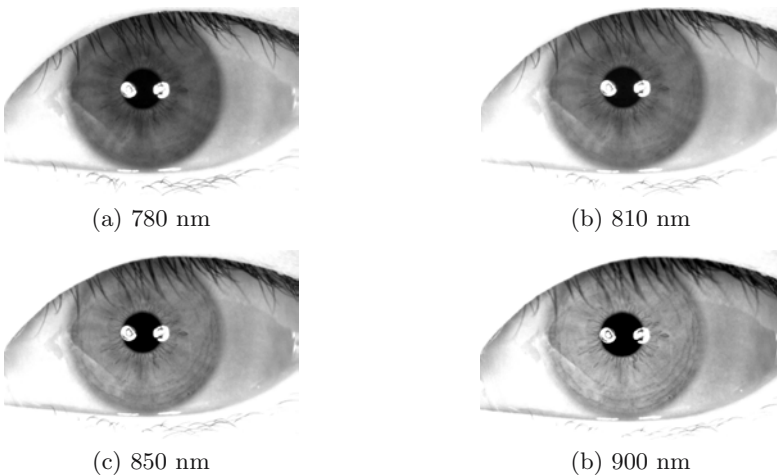

(a) 780 nm   (b) 810 nm

(c) 850 nm   (b) 900 nm

Fig. 2. Example of multi-spectral iris images of a person

and the contrast become different. We chose 780 nm and 900 nm from the four wavelengths as the two spectral bands to be used.

In this experiment, the multi-spectral fake iris images are artificially generated using the multi-spectral images of real irises instead of taking the images of counterfeited irises. It is due to the fact that if the quality of the fake irises affect the performance of the recognition, the ability of destroying the fused image of fake iris images cannot be assessed accurately. The artificial images are made by changing the brightness and the contrast of a real iris image containing enough iris pattern. We took multi-spectral images of four persons and generated the artificial multi-spectral images of fake irises. The images were taken several times and the fused image using one set of the images were used as the template data. Figure 3 shows the fused images of a real iris and a fake iris. The major features are maintained in the result of the real iris, while the features of the fused image using the artificial images are not clear and many major features are already ruined. The matching process based on Daugman's algorithm [2] is performed and the average hamming distance values of both the real irises and the fake irises of each person are shown in Table 1. Most of the results from fake irises show very high hamming distance values above 4.0, while all the results from real irises show acceptable match results.
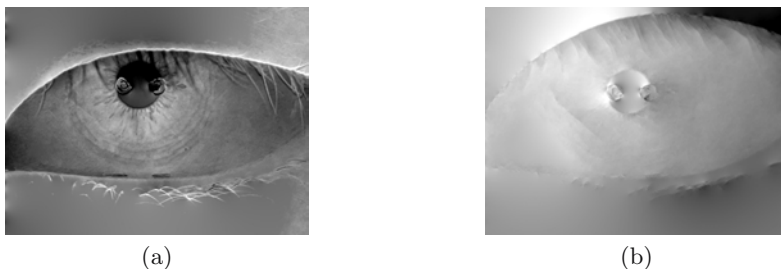

(a)   (b)

Fig. 3. Example of the fusion results from (a) real iris images (b) artificial iris images

5   Conclusions

In this paper, we introduced an iris recognition scheme against the counterfeit attacks. Multi-spectral iris images are taken and merged into a grayscale image.

Table 1. Average hamming distance values of real and fake irises

	Hamming Distance Values	
	Real	Fake
Iris 1	0.258	0.463
Iris 2	0.213	0.475
Iris 3	0.249	0.473
Iris 4	0.262	0.453

The fake iris is blocked in two processes. Firstly, if the images contain energy above or below certain amount, they are discarded. Secondly, if the images are taken from a fake iris, the gradient-based image fusion process generates a poor image since the gradient orientation is confused in the fusion process. The experimental results showed that the proposed scheme successfully rejected the fake inputs. This algorithm has an advantage that it can be applied to any iris recognition algorithms using one sheet of image for matching.

References

1. Wildes, R.P.: Iris recognition: An emerging biometric technology. Proceedings of the IEEE **85** (1997) 1347–1363
2. Daugman, J.: How iris recognition works. IEEE Transactions on Circuits and Systems for Video Technology **14** (2004) 21–30
3. Boles, W.W., Boashash, B.: A human identification technique using images of the iris and wavelet transform. IEEE Transactions on Signal Processing **46** (1998) 1185–1188
4. Ruud M. Bolle, J.H.C., Ratha, N.K.: Biometric perils and patches. Pattern Recognition **35** (2002) 2727–2738
5. Ernst, J.: http://www.iris-recognition.org/counterfeit.htm. (2005)
6. Jain, A.K., Uludag, U.: Hiding biometric data. IEEE Transactions on Pattern Analysis and Machine Intelligence **25** (2003) 1494–1498
7. Li Ma, Tieniu Tan, Y.W., Zhang, D.: Personal identification based on iris texture analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence **25** (2003) 1519–1533
8. Zenzo, S.D.: A note on the gradient of a multi-image. Computer Vision, Graphics, and Image Processing **33** (1986) 116–125
9. Scheunders, P.: A multivalued image wavelet representation based on multiscale fundamental forms. IEEE Transactions on Image Processing **10** (2002) 568–575
10. Chung, D.H., Sapiro, G.: On the level lines and geometry of vector-valued images. IEEE Signal Processing Letters **7** (2000) 241–243
11. Socolinsky, D.A., Wolff, L.B.: Multispectral image visualization through first-order fusion. IEEE Transactions on Image Processing **11** (2002) 923–931

An Iris Detection Method
Based on Structure Information

Jiali Cui[1], Tieniu Tan[1,*], Xinwen Hou[1], Yunhong Wang[1,2], and Zhuoshi Wei[1]

[1] Center for Biometrics and Security Research
National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, P.O. Box 2728, Beijing, P.R. China, 100080
{jlcui,tnt,xwhou,wangyh,zswei}@nlpr.ia.ac.cn
[2] School of Computer Science and Engineering
Beihang University, Beijing, P.R. China, 100083
yhwang@buaa.edu.cn

Abstract. In this paper, we propose an iris detection method to determine iris existence. The method extracts 4 types of features, i.e., contrast feature, symmetric feature, isotropy feature and disconnectedness feature. Adaboost is adopted to combine these features to build a strong cascaded classifier. Experiments show that the performance of the method is promising in terms of high speed, accuracy and device independence.

1   Introduction

Biometrics [1] has become a promising topic in the last decade and has many applications such as access control, network security, etc. Iris recognition, as a typical biometric technique, has been receiving more and more attention [2-5] due to its high accuracy. The idea of iris recognition goes back to the 19th century, when a French doctor recognized prisoners by their iris color. In the last decade, many researchers were interested in it. Daugman realized an automatic recognition system in 1993 [2]. Wildes developed a device to get iris images from distance and used super-resolution to obtain clear images [3]. Others [4, 5] used wavelet and Gabor filters to extract iris features, respectively.

   Iris detection is an important procedure in preprocessing. We must emphasize that the term iris detection in this paper is defined as determining whether there is an iris in an input image. Although 'iris detection' has been used in other papers, its meaning is similar to eye detection for gaze evaluation, expression synthesis, face recognition, etc. Iris detection is a necessary procedure in an iris recognition system and its need comes mainly from the following reasons. First, it can improve the performance of iris recognition systems. Most of the systems assume that there is an iris in the input image. Since many images in the input sequence of an iris sensor do not include iris, it is very necessary to separate iris images (images with iris) from non-iris images (images without iris) to avoid unnecessary computation. It can also avoid matches between non-iris images and iris templates to give wrong decision. Sometimes, these non-iris images cannot be excluded by high-light blocks, circle verification and liveness detection. Another reason is that it is a general and fast coarse local-

---

*   Corresponding author. Tel: 86-10-62616658, Fax: 86-10-62551993

ization method and important for iris recognition. As we know, many iris localization methods, such as Daugman's circle-finding method, take a coarse-to-fine strategy. Then, the speed and accuracy of iris localization is influenced by coarse localization results. Moreover, it is a method independent of device. Some systems find high-light blocks (papillary reflection of infrared light) as coarse localization, but the number and size of high-light blocks is different from device to device. A general coarse localization method should be device independent.

However, there seems no fast iris detection method up to now. Although our previous work [6] detects iris using an appearance-based method, it is not fast. Inspired by work described by Viola and Jones [7], we propose a fast iris detection method based on structure information. The algorithm extracts 4 types of features, namely contrast feature, symmetric feature, isotropy feature and disconnectedness feature. Then Adaboost is adopted to select useful features from the feature set to build a cascaded classifier.

The remainder of the paper is organized as follows. The algorithm of iris detection is described in detail in Section 2. Section 3 gives the experimental results of the method. The paper is concluded in Section 4.

2   Proposed Method

The task of iris detection is to determine iris existence in an image and, if existence, to locate its position and size. This task involves three major processes. The first process is iris modeling, which characterizes the iris structure properties. The second is feature extraction, which extracts iris features to represent iris structure. The last is feature selection and classifier design, which computes the classification ability of each feature and combines the features of good classification ability to build a classifier.

2.1   Iris Modeling

Iris modeling is to character iris properties and provide cues to feature extraction. To the best of our knowledge, although color is important for iris patterns of westerners, most personal identification systems based on iris recognition use intensity images. Moreover, gray value of iris images captured via different devices is different. Although texture plays an important role, texture feature extraction is computationally complex. So structure (gray level distribution) is most important for iris modeling.

If an iris image is seen as a surface, it can be found that iris (including pupil) is like an inverse hat (See Fig.1). Thus, an inverse-hat model is adopted. It indicates that the center of an iris is of low gray value and the surrounding parts are of high gray value.

2.2   Feature Extraction

Feature extraction is to extract features to represent iris patterns. As we know, spatial gray level co-occurrence matrix (GLCM) [8] is a widely used feature for structural patterns. However, GLCM is a statistical feature and computationally complex. Based on the proposed inverse-hat model, four types of features are introduced and extracted as follows.
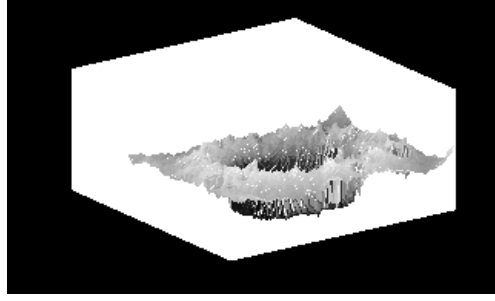
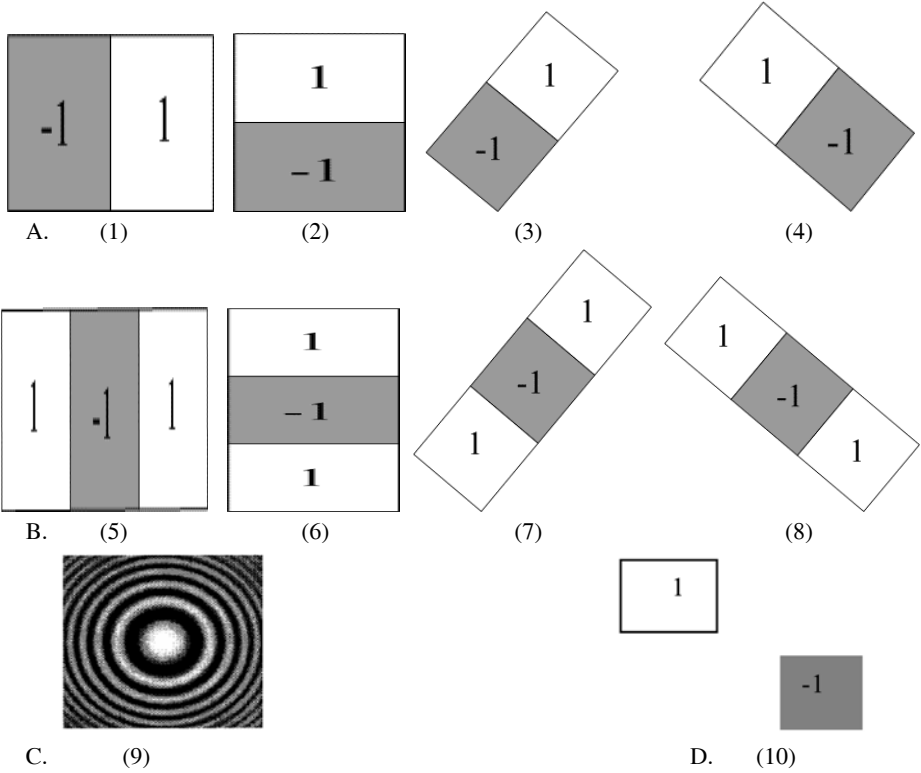**Fig. 1.** 3D surface representation of an iris region

Fig. 2. Four types of features

In Fig.2, four types of features are contrast feature (denoted by A, including features 1, 2, 3, 4 shown in Fig.2), symmetric feature (denoted by B, including features 5, 6, 7, 8), isotropy feature (denoted by C, including feature 9) and disconnectedness feature (denoted by D, including feature 10), respectively. They are computed as follows.

A feature, denoted by f, is the difference (d) of two regions normalized by the variance (v) of the total image and is computed as follows.

(1) Compute d. For type A, B and D, $d =$ (sum of gray values in region denoted by 1) $-$ (sum of gray values in region denoted by -1). For type C, d is difference of two rings and computed as follows.

$$S_i = \int_{r_i}^{r_{i+1}} \int_0^{2\pi} I(\rho,\theta) d\rho d\theta \qquad (1)$$

$$d_{i,j} = s_i - s_j \qquad (2)$$

Circle generation used in Eqn. (1) is computed using Bresenham algorithm [9] (a famous method in computer graphics).

(2) Computer the variance v of the image. For the sake of speed, we introduce a fast algorithm for variance computation. Let $h(i)$ be the histogram of the images and I be an image, then

$$Var(I) = E(I)^2 - (EI)^2$$
$$= \frac{1}{N} \sum_{i,j} I(i,j)^2 - (\frac{1}{N} \sum_{i,j} I(i,j))^2 \qquad (3)$$
$$= \frac{1}{N} \sum_{i=1}^{255} h(i)i^2 - \frac{1}{N^2} (\sum_{i,j} I(i,j))^2$$

Here M and N are the size of an image. $E(I)$ and EI are the mean of the total image. Because i^2 ($i = 0,1,...,255$) can be made a lookup table, Eqn (3) needs (at most) only 2*M*N+256 integer additions, 256 integer multiplications and 2 divisions.

(3) The feature is $f = d/v$.

Finally, a feature set is built though changing the size of the regions shown in Fig.2.

2.3 Feature Selection and Classifier Design

Our method is inspired by Viola and Jones' work [7], which obtains high speed. Therefore, Adaboost is adopted to select features to build a cascaded classifier. Feature selection procedure based on Adaboost is introduced briefly because it is not the focus of the paper. Adaboost takes a rejection strategy to eliminate negative samples and puts big weight on the outlier step by step (or layer by layer). If a sample passes all the layers, it can be seen as a positive sample. Its procedures are shown in Fig.3.

3 Experimental Results and Analysis

Experiments were done on a PC with PIV 2.4GHz processor and 256M DRAM. The code was written in VC++6.0.
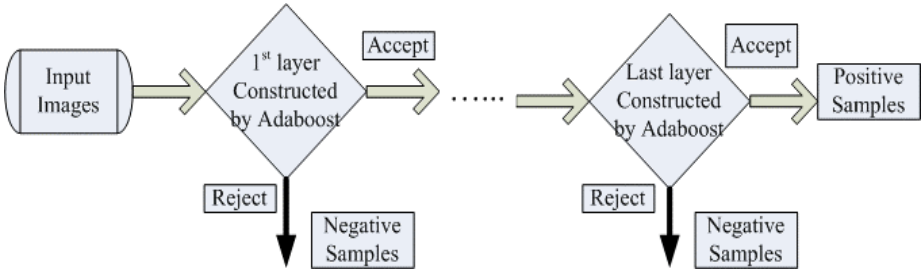
Fig. 3. Flowchart of iris detection

3.1 Training Database

The training database comprises of 12735 positive samples and 13924 negative samples. Positive samples include segmented iris images with rotation, scale and transition transform. We have two iris sensors denoted by sensor 1 and 2, respectively. Although samples based on different sensors are available, only one database (captured by sensor 1) is used in training in order to verify the generality and robustness of our method. Negative samples include 3 categories, namely natural images (1700 images including cars, persons, vessels, scenes, animals, etc. A database used for image retrieval is adopted here), face images (7035 images including only closed eyes) and background images (5189 images of the application background, such as wall, device, etc). Three categories of negative samples are selected to make the method both general and specific. More face images with closed eyes are selected because those images are very similar to iris images and they occur very frequently in real applications. Some examples of the training database are shown in Fig.4.
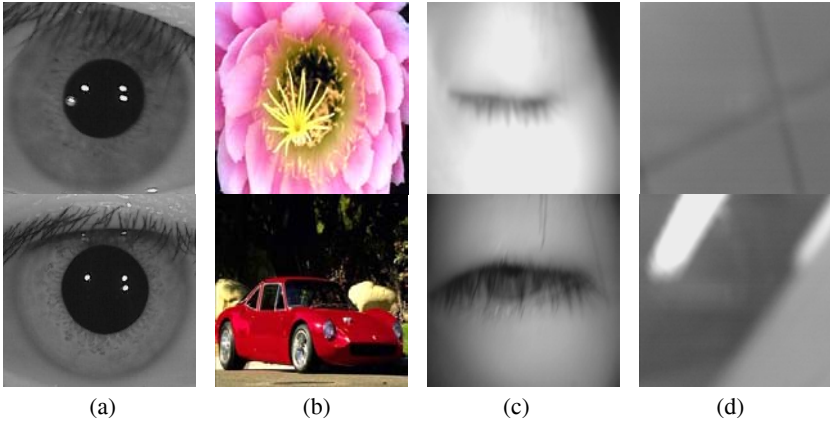

(a)   (b)   (c)   (d)

Fig. 4. Examples from training database (a) positive samples (b) natural images (c) face images with closed eyes (d) background images

3.2 Test Database

The test database includes 3000 iris images from 100 subjects and 3135 non-iris images. Iris images include 3 categories, (A) 1000 images captured by sensor 1, (B)

1000 images captured by sensor 2, (C) 1000 images captured by either sensor 1 or 2, but users wear glasses during data acquisition. There is no overlap between training database and test database.

3.3   Results and Analysis

In training process, a cascaded classifier with 24 layers is obtained. 437 features were selected in the classifier and the number of each feature in the classifier is listed in Table 1.

Table 1. Number of each feature selected in cascaded classifier

Feature Index	1	2	3	4	5	6	7	8	9	10
Number	18	62	32	36	87	104	20	21	30	27

It can be seen that feature 6 is selected 104 times and it indicates that vertical symmetry is very important to characterize iris structure.

Experiments also show that the method has promising performance in terms of speed and accuracy. Averagely, the detection process costs only 16ms. High speed makes it possible to search in an image exhaustively. The detection accuracy is listed in Table 2.

Table 2. Detection accuracy

Category	# of Correctly classified images	Recognition Rate
Non-iris Images	3087	98.4%
Iris Images (A)	980	98%
Iris Images (B)	965	96.5%
Iris Images (C)	880	88%

Table 2 indicates that the method is nearly device independent, which profits from feature normalization and down sampling. The recognition rate decreased sharply when users wear glasses and the reason is reflection. When users wear glasses, there may be much reflection in images and the reflection changes the gray level distribution. Indeed, the accuracy is measured in comparison with human visual assessment, so the results may be subjective. Some examples are shown in Fig.5, where white box indicates the position of an iris.
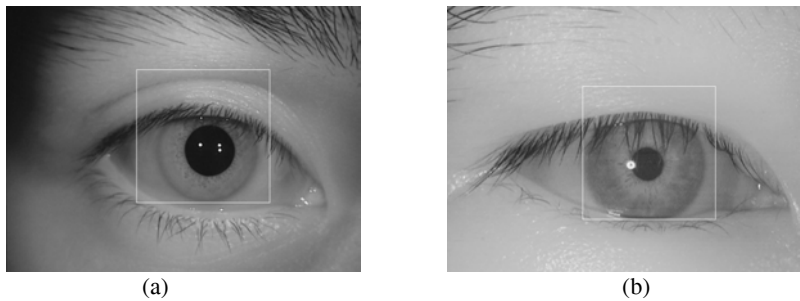

(a)   (b)

Fig. 5. Detection results (a) an iris image captured by sensor 1 (b) an iris image captured by sensor 2

To select a proper number of layers, the following experiment is done. We count the number (percentage p) of images that can pass (at most) x layers and draw the histogram p vs x in Fig.6. In Fig.6, x-axis and y-axis denote layer number and image percentage, respectively. Black bar denotes positive samples and gray bar denotes negative samples. It can be seen that most negative samples can be rejected using 16 layers; however, in real application we use 24 layers because there are many face images with closed or half-closed eyes that can pass 23 or 24 layers.
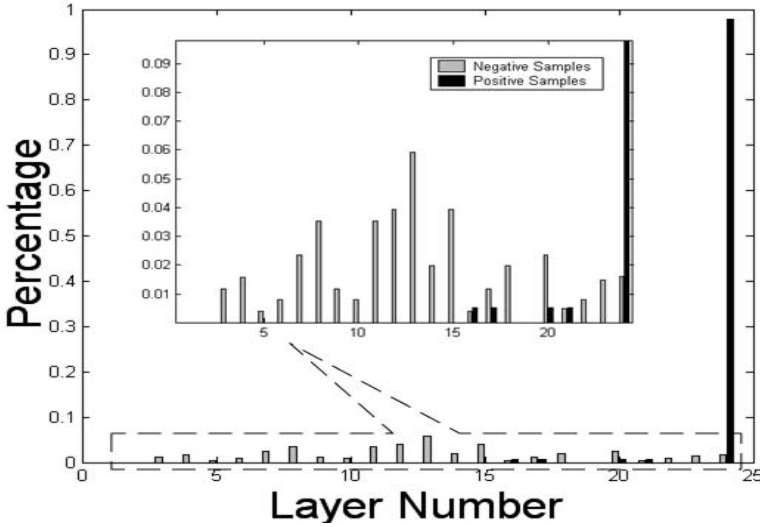
Fig. 6. Sample distribution vs layer number

4   Conclusions

This paper has proposed an iris detection method based on iris structure characterized by gray value distribution. The method first extracts 4 types of features to character-ize iris structure. The performance of the proposed method is promising in terms of high speed, accuracy and device independence. The method is fast and, therefore, it can be used in real time systems and can search an image exhaustively. The high recognition rate indicates that structure information is very efficient in iris detection. Moreover, the method is device independent, so when it is used in a new device, it does not need new training. In the future, we will focus on overcoming glasses reflec-tion and collecting more negative samples.

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. J.Daugman, "High Confidence Visual Recognition of Persons by a Test of Statistical Independence", IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 15, No.11, pp.1148-1161,1993.
3. R.Wildes, J.Asmuth, et al., "A Machine-vision System for Iris Recognition", Machine Vision and Applications, Vol.9, pp.1-8, 1996.
4. Seung-In Noh, Kwanghyuk Bae, Yeunggyu Park, Jaihie Kim, "A Novel Method to Extract Features for Iris Recognition System", LNCS, Volume 2688 / 2003, pp. 862 – 868, August 2003.
5. L. Ma, T. Tan, Y. Wang and D. Zhang, "Personal Identification Based on Iris Texture Analysis", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 25, No. 12, pp.1519-1533, 2003.
6. Jiali Cui, Li Ma, Yunhong Wang, Tieniu Tan and Zhenan Sun, "An Appearance-based Method for Iris Detection", the 6th Asian Conference on Computer Vision (ACCV), Vol. 2, pp. 1091-1096, 2004, Jeju Korea.
7. Paul Viola and Michael Jones, "Robust Real-time Object Detection", International Journal of Computer Vision 57(2), 137–154, 2004.
8. David A. Clausi, "An analysis of co-occurrence texture statistics as a function of grey-level quantization", Canadian Journal of remote sensing, vol. 28 no. 1 pp. 45-62, 2002.
9. W.E. Wright, "Parallelization of Bresenham's Line and Circle Algorithms," IEEE CG&A, Vol. 10, No. 5, 1990, pp. 60-67.

Constructing the Discriminative Kernels Using GMM for Text-Independent Speaker Identification

Zhenchun Lei, Yingchun Yang, and Zhaohui Wu

College of Computer Science and Technology, Zhejiang University
310027 Hangzhou, P.R. China
{leizhch,yyc,wzh}@zju.edu.cn

Abstract. In this paper, a class of GMM-based discriminative kernels is proposed for speaker identification. We map an utterance vector into a matrix by finding the sequence of components, which have the maximum likelihood in the GMM for the all frame vectors. And the weights matrix was used, which were got by the GMM's parameters. Then the SVMs are used for classification. A one-versus-rest fashion is used for the c class problem. Results on YOHO in text-independent case show that the method can improve the performance greatly compared with the basic GMM.

1 Introduction

The support vector machine (SVM) [1] is based on the principle of structural risk minimization, which has got more attention in speaker identification recently for its superior performance [2-5]. The methods using SVMs in speaker verification and identification can be divided into frame-based and utterance-based. In the former, every frame is scored by the SVMs and the decision is made based on the accumulated score over the entire utterance [2]. Another is utterance-based approaches which map an utterance into a vector as SVM's input, and the researchers focus on how to construct a better kernel dealing with utterances having different lengths, such as fisher kernel [3] and dynamic time-alignment kernel [4], etc.

In this paper, we construct a class of GMM-based kernels for text-independent speaker identification. Our experiments were run on the YOHO database and in the text-independent speaker identification case.

This paper is organized in the following way: section 2 is the GMM's brief introduction. In section 3 we review the SVMs theory briefly and some apply approaches in speech recognition and speaker recognition using SVMs. The new discriminative kernels will be described in section 4. Section 5 presents the experimental results on YOHO. Finally, section 6 is devoted to the main conclusions.

2 Gaussian Mixture Model

One of the most successful text-independent recognition methods is the Gaussian mixture model [6, 13]. It provides an effective way to describe the personal speech characters, and one of its powerful attributes is the capability to form smooth approximations to arbitrarily shaped densities.

For a D-dimensional feature vector, x, the mixture density used for the likelihood function is defined as:

$$p(x \mid \lambda) = \sum_{i=1}^{M} w_i p_i(x) \tag{1}$$

The density is a weighted linear combination of M unimodal Gaussian densities, $p_i(x)$, each parameterized by a mean $D \times 1$ vector, μ_i, and a $D \times D$ covariance matrix, Σ_i:

$$p_i(x) = \frac{1}{(2\pi)^{D/2} |\Sigma_i|^{1/2}} \exp\{-\frac{1}{2}(x - \mu_i)'(\Sigma_i)^{-1}(x - \mu_i)\} \tag{2}$$

The mixture weights, w_i, furthermore satisfy the constraint:

$$\sum_{i=1}^{M} w_i = 1 \tag{3}$$

The parameters of the density model are denoted as $\lambda = \{w_i, \mu_i, \Sigma_i\}$, where i=1,...,M.

For reason of both modeling and estimation, it is usual to employ GMMs consisting of components with diagonal covariance matrices. And we also use only diagonal covariance matrices in this paper. Given a collection of training vectors, maximum likelihood model parameters are estimated using the iterative expectation-maximization (EM) algorithm.

The feature vectors of X are assumed independent, and the log-likelihood of a model λ for a sequence of feature vectors, $X = \{x_1, ..., x_T\}$, is computed as:

$$S(X) = \log p(X \mid \lambda) = \sum_{t=1}^{T} \log p(x_t \mid \lambda) \tag{4}$$

The utterance score is used to make a decision by comparing it against a threshold that has been chosen for a desired tradeoff between detection error types.

3 Support Vector Machine

SVM [1, 12] is a binary classifier, which implement the structural risk minimization principle in statistical learning theory by generalizing optimal hyper-plane with maximum margin in two classes of data.

Given a data set $\{x_i, y_i\}, i = 1, ..., n, y_i \in \{-1, +1\}, x_i \in R^d$, we try to find a hyperplane to separate positive and negative data. The hyperplane is defined as following:

$$x \cdot w + b = 0 \tag{5}$$

In linear non-separable case, SVM searches for the maximum margin and the minimum error. The slack variables, ξ_i, are introduced and the optimization problem is:

$$\min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \tag{6}$$

subject to

$$y_i(w^T x_i + b) \geq 1 - \xi_i, i = 1,...,n$$
$$\xi_i \geq 0, i = 1,...,n \tag{7}$$

C is the misclassification penalty parameter and controls the degree of penalty to the misclassification data.

The decision function can be shown to have the form:

$$f(x) = w^T x + b = \sum_{i=1}^{n_s}\alpha_i y_i s_i^T x + b \tag{8}$$

where n_s is the number of support vectors, $s_i, i = 1,...,n_s$ are the corresponding support vectors and α_i are Lagrange multipliers which are found by solving the following quadratic programming (QP) problem.

$$\max_{\alpha}(\sum_i \alpha_i + \sum_i \sum_j \alpha_i \alpha_j y_i y_j x_i \cdot x_j) \tag{9}$$

subject to:

$$\sum_i \alpha_i y_i = 0$$
$$0 \leq \alpha_i \leq C \tag{10}$$

The main feature of the SVM is that its target functions attempts to minimize the number of errors made on the training set while simultaneously maximizing the margin between the individual classes. This is an effective prior for avoiding over-fitting, which results in a sparse model dependent only on a subset of kernel functions.

The extension to non-linear boundaries is acquired through the use of kernels that satisfy Mercer's condition. The kernels map the original input vector x into a high dimension space of features and then compute a linear separating surface in this new feature space. In practice, the mapping is achieved by replacing the value of dot production between two data points in input space with the value that results when the same dot product is carried out in the feature space. The following is formations:

$$\max_{\alpha}(\sum_i \alpha_i + \sum_i \sum_j \alpha_i \alpha_j y_i y_j K(x_i, x_j)) \tag{11}$$

The kernel function K defines the type of decision surface that the machines will build. In our experiments, the radial basis function (RBF) kernel and the polynomial kernel are used and they take the following forms:

$$k_{rbf}(x_i, x_j) = \exp\left[-\frac{1}{2}(\frac{\|x_i - x_j\|}{\sigma})^2\right] \tag{12}$$

$$k_{poly}(x_i, x_j) = (x_i \cdot x_j + 1)^n$$

where σ is the width of the radial basis function and n is the order of the polynomial. The use of kernels means that an explicit transformation of the data into the feature space is not required.

Recent several approaches using SVMs have been proposed in speech applications and speaker recognition. Constructing a superior kernel function for utterances can be difficult and still a challenge. A well-known kernel is the fisher kernel made by Jaakkula and Haussler [3], which has been explored for speech recognition in [8] and speaker recognition in [10]. The dynamic time alignment kernel (DTAK) is developed by incorporating an idea of non-linear time alignment into the kernel function [9]. The pair HMM a kernel is similar to DTAK, which may be more suited to a text dependent speaker verification task where the model topology is predefined [10]. Campbell also introduced the sequence kernel derived from generalized linear discriminates in [11].

Like them, we will construct a new class of kernels to deal with the variable length utterances in this paper. In the GMM, the score for a frame vector is the mixture of all component scores, and the total score for an utterance is accumulated on all frame vectors. But the score source and distribution on components are discarded. So a class of kernels can be constructed by using the parameters in the GMM.

4   GMM-Based Discriminative Kernels

The utterance, X, is denoted as a sequence of acoustic feature vectors $X = \{x_1, ..., x_n\}$, and the vector x_i have d components. A GMM $\lambda = \{w_i, \mu_i, \Sigma_i\}$ has been got from EM algorithm. Now we construct a new kernel.

Campbell [11] brought a generalized linear discriminate sequence (GLDS) kernel based on Taylor approximate, which defined the mapping $X \to \overline{b}$ as:

$$X \to \frac{1}{n} \sum_{i=1}^{n} b(x_i) \tag{13}$$

We also adopted this idea and extended to combine the GMM. We mapped an utterance to multiple $\overline{b}$ by mapping every frame to its nearest component, so the result is a matrix rather than a vector and then be translated to a vector by expanding simply. We simply set $b(x)$ to be the difference vector to a mean in the component in our current experiments.

For x_i, we can find the component from all components in the GMM which has the maximal likelihood:

$$w_t p_t(x_i) = \arg \min_{j=1...M} \{w_j p_j(x_i)\} \tag{14}$$

then map x_i to an matrix:

$$Z(x_i) = [v_1 \cdot z_1, ..., v_M \cdot z_M] \tag{15}$$

where

$$z_k = \begin{cases} x_i - \mu_t, k = t \\ \quad 0, else \end{cases} \tag{16}$$

The coefficients v_i are the weights. We constructed the weight coefficients using the GMM's parameters, covariance matrix Σ_i and the mixture weights w_i, which are following:

$$v_i = w_i \cdot sqrt(diag(\Sigma_i)^{-1}) \tag{17}$$

For Σ_i is a diagonal covariance matrix, there is nothing lost. The matrix Z has d*M size and it reflect the weighted distance between x_i and the nearest component at dimension-level.

After compute the mapped matrix $Z(x_i)$ for every frame vector, we can get the mapped matrix for the whole utterance X by using the mean:

$$\Phi(X) = \frac{1}{n}\sum_{i=1}^{n} Z(x_i) \tag{18}$$

And after expanding $\Phi(X)$ to a one-dimension vector simply, a d*M size vector can be obtained. Also we can describe the linear kernel between $\Phi(X)$ and $\Phi(Y)$ directly:

$$K_{linear}(X,Y) = \sum_{i=1}^{d}\sum_{j=1}^{M} \Phi(X)_{ij} \cdot \Phi(Y)_{ij} \tag{19}$$

And the polynomial and RBF kernel can also be constructed in the same way.

$$K_{poly}(X,Y) = (\sum_{i=1}^{d}\sum_{j=1}^{M} (\Phi(X)_{ij} \cdot \Phi(Y)_{ij}) + 1)^n \tag{20}$$

$$K_{rbf}(X,Y) = \exp\left[-\frac{1}{2} \cdot \frac{\sum_{i=1}^{d}\sum_{j=1}^{M} (\Phi(X)_{ij} - \Phi(Y)_{ij})^2}{\sigma^2} \right] \tag{21}$$

where σ is the width of the radial basis function and n is the order of the polynomial.

5 Experiments

Our experiments were performed using the YOHO database. This database consists of 138 speaker prompted to read combination lock phrases, for example, "29_84_47". Every speaker has four enrollment sessions with 24 phrases per session and 10 verify sessions with 4 phrases per session. The features are derived from the waveforms

using 12th order LPC analysis on a 30 millisecond frame every 10 milliseconds and deltas computed making up a twenty four dimensional feature vector. Mean removal, preemphasis and a hamming window were applied. Energy-based end pointing eliminated non-speech frames.

The 50 speakers, labeled 101 to 154, were used in our experiments and 50 SVMs were trained one speaker from all other 49 speakers. For every speaker, the EM algorithm was used on its 96 enroll utterances, and a GMM could be got. Then all speakers' utterances were been mapped based on the models, and the support vector machines were training using the vectors as inputs.

The SVM is constructed to solve the problem of binary classification. For N-class, the general method is to construct N SVMs [12]. The ith SVM will be trained with all of the examples in the ith class with positive labels, and all other examples with negative labels. We refer to SVMs trained in this way as 1-vs-r (one-versus-rest) SVMs. Another method is 1-vs-1 (one-versus-one) SVMs, which construct $K=N(N-1)/2$ classifiers and each classifier be trained on only two out of N classes. In our experiments, the 1-vs-r method was adopted.

In training phase, 50 SVMs were trained discriminating one speaker from all other 49 speakers. Training SVMs rely on quadratic programming optimizers, so it is not easily to large problems. There are some algorithms for this problem and the SMO [14] is used.

In our experiments, we used two component numbers, 64 and 128, which generated the mapped vectors having 24*64=1536 and 24*128=3072 dimensional space respectively. The SVM with linear kernel, polynomial kernel and radial basis function kernel were trained.

Table 1 shows the results.

Table 1. Performance of each model for text independent speaker identification experiments on YOHO database

Models	Error rate on 64 components (%)	Error rate on 128 components (%)
SVM/GMM(linear kernel)	3.5	2.8
SVM/GMM(rbf kernel)	2.9	2.5
SVM/GMM(polynomial kernel)	2.5	2.2
Basic GMM	5.4	3.2

Table 1 show that the linear kernel, rbf kernel and polynomial kernel of SVM/GMM improve 35.2%, 46.3% and 53.7% compare to the GMM respectively. And when 128 codebook size, they are 12.5%, 21.9% and 31.3%.

6   Conclusions

A class of discriminative kernels for text independent speaker identification was presented. The basic idea is mapping the variable length utterances to the fixed size vectors by the GMM. The mapping procedure is running on the maximum likelihood, and the weights using GMM's parameters were taken into account. The experiments on YOHO show that our method can get the better performance than the basic GMM.

Acknowledgments

References

1. V.Vapnik. Statistical Learning Theory. Johm Wiley and Sons, New York, (1998)
2. V.Wan, W.M.Campbell, Support Vector Machines for Speaker Verification and Identification. in Proc. Neural Nerworks for Signal Processing X, (2000) 775-784
3. T.S.Jakkola and D.Haussler. Exploiting generative models in discriminative classifiers. In Advances in Neural Information Processing System 11, M.S.Kearns, S.A.Solla, and D.A.Cohn, Eds. MTT Press, (1999)
4. Nathan Smith, Mark Gales, and Mahesan Niranjan, Data-dependent kernel in SVM classification of speech patterns. Tech.Rep. CUED/F-INFENG/TR.387, Cambridge University Engineering Department, (2001)
5. M.Schmidt and H.Gish, Speaker Identification via Support Vector Machines, in ICASSP, (1996) 105-108
6. D.A.Reynolds, T.Quatieri, and R.Dunn, Speaker verification using adapted Gaussian mixture models, Digital Signal Processing, vol.10, no.1-3, (2000)
7. W.M.Campbell, A SVM/HMM SYSTEM FOR SPEAKER RECOGNITION, in Proc. ICASSP, (2003)
8. Shai Fine, Jiri Navratil, and Ramesh A.Gopinath, A hybrid GMM/SVM approach to speaker recognition. in Proc. ICASSP (2001)
9. Hiroshi Shimodaira, Kenichi Noma, Mitsuru Nakai and Shigeki Sagayama, Dynamic Time-Alignment Kernel in Support Vector Machine, NIPS, (2001) 921-928
10. Vincent Wan, Steve Renals, Valuaution of Kernel Methods for Speaker and Identification. in Proc. ICASSP, (2002)
11. W.M.Campbell, GENERALIZED LINEAR DISCRIMINANT SEQUENCE KERNEL FOR SPEAKER RECOGNITION, in Proc. ICASSP, (2002) 161-164
12. C.J.C.Burges, A tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, vol.2, no.2, (1998) 1-47
13. D.A.Reynolds and R.C.Rose, Robust text-independent speaker identification using Gaussian mixture speaker models, IEEE Trans. Speech Audio Processing, vol.3, (1995) 72-83
14. J.Platt.Fast training of SVMs using sequential minimal optimisation. Advances in Kernel Methods: Support Vector Learing, MIT press, Cambridge, MA, (1999) 185-208

Individual Dimension Gaussian Mixture Model for Speaker Identification

Chao Wang, Li Ming Hou, and Yong Fang

School of Communication and Information Engineering, Shanghai, China
lmhou@staff.shu.edu.cn

Abstract. In this paper, Individual Dimension Gaussian Mixture Model (IDGMM) is proposed for speaker identification. As to the training-purpose feature vector series of a certain register, its joint probability distribution function (PDF) of is modeled by the product of the PDF of each dimension (marginal PDF), the scalar-based Gaussian Mixture Model (GMM) serving as the marginal PDF. For a good discriminative capability, the decorrelation by Schmidt orthogonalization and the Mixture Component Number (MCN) decision are adopted during the train. A close-set text-independent speaker identification experiment is also given. The simulation result shows that the IDGMM accelerates the training process remarkably and maintains the discriminative capability in testing process. …

1 Introduction

Speaker Identification (SI) [1] is a process of finding the identity of a client without a speaker first declaring his/her identity. It's composed of 3 parts: feature coefficient extraction, modeling (training) and scoring (testing). In general, Mel-Frequency Cepstrum Coefficient (MFCC), Gaussian Mixture Model (GMM) and Bayes score is the representative one respectively.

Over the past several years, GMM, a joint probability distribution function (PDF) described by the weighted sum of several multivariate Gaussian PDFs, has become the prevailing approach for modeling in SI. Each multivariate Gaussian PDF is termed as a Mixture Component (MC), whose parameters (weighted coefficient, mean vector and covariance matrix) are usually obtained by the Expectation-Maximum (EM) iterative algorithm [2]. The Mixture Component Number (MCN) is an experiential solution obtained after numerous simulations, 16/32/64 (especially 32) being widely used.

Effective in modeling as GMM is, there still exists deficiency (heavy computation burden especially in training) related to the dimension number of feature vector. The traditional way of simplification is to assume that the covariance matrix of Gaussian PDF is not a full matrix but a diagonal matrix, which facilitates the covariance matrix estimation and the probability computation of Gaussian PDF [2]. However, such simplification accelerates the modeling moderately and discards some identity information useful in speaker discrimination.

Individual Dimension Model (IDM) that models each dimension series individually is an intuitive way to accelerate the training remarkably. Frederic [3] initiates such research by the histogram–based modeling on each dimension series of training-purpose feature vector series. However, its discriminative capability degrades re-

markably. The partial reason steps from the following facts: the identity information embedded in the training-purpose feature vector series is composed of the inter-dimension one and the intra-dimension one. It's obviously that Frederic's method only use the intra-dimension one and omit the inter-dimension one.

In this paper, on the basis of GMM and IDM, we propose an Individual Dimension Gaussian Mixture Model (IDGMM) as the model of speaker identification. As to the training-purpose feature vector series of a certain register, its joint probability distribution function (PDF) of is modeled by the product of the PDF of each dimension (marginal PDF), the scalar-based Gaussian Mixture Model (GMM) serving as the marginal PDF. For a good discriminative capability, the decorrelation by Schmidt orthogonalization and the Mixture Component Number (MCN) decision are adopted during the train. A close-set text-independent speaker identification experiment is given. The simulation result shows the proposed IDGMM accelerates the modeling remarkably and maintain recognition performance.

2   Individual Dimension Gaussian Mixture Model (IDGMM)

As for the IDM, it's only independence between dimensions of training purpose vector that might make the discriminative capability of IDM close to that of GMM. Such equality (the equality between the joint PDF and the product of marginal PDF) means that the identity information can only be represented by the intra-dimension identity information. From such fact, we induce the modeling and scoring algorithm of IDGMM concerned with independence as follows:

2.1   The Modeling Algorithm of IDGMM

Figure 1 is the flow chart of IDGMM modeling algorithm in which the feature coefficient is exemplified by MFCC. It can be detailed as follows:

 1) The correlation between each two dimensions in the original feature vector of train speech is removed by Schmidt orthogonalization. The total mean vector and multiplication matrix of Schmidt orthogonalization are also included in IDGMM as decorrelation parameter.
   In fact, this decorrelation serves as the replacement of independence transform.
 2) For each decrorelated dimension series, the histogram should be generated and input to a Mathematic Morphology (MM) filter subsequently. It's peak number should be detected and would be used as the MCN of the corresponding scalar-based GMM, which steps from the fact that for a dimension series the histogram is a coarse estimation of the marginal PDF.
    It's intuitively that when independence exists the total MCN number of joint PDF equals to the product of that of marginal PDF. Although independence has been replaced by decorrelation, such method (the total MCN decision by histogram) can describe the joint PDF more precisely than the traditional experiential-solution-based way. MM filter contributes to controlling the total MCN that provide a good tradeoff between the speed and the discriminative capability.
 3) Each decrorelated dimension series is modeled by a scalar-based GMM according to the MCN mentioned above. The scalar-based GMM is still modeled by EM algorithm.

4) Since the decorrelation is a superset of independence, the joint PDF can be approximated by the product of marginal PDF (the scalar-based GMM of each dimension series).
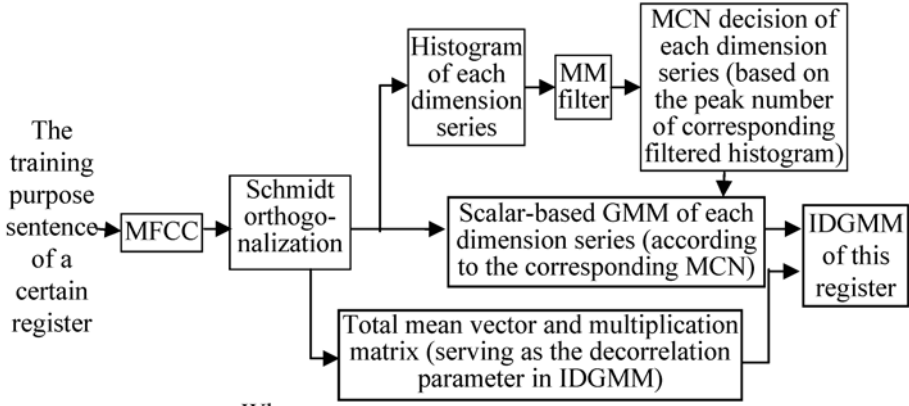
Fig. 1. The modeling (training) algorithm of IDGMM

2.2 The Scoring Algorithm of IDGMM

Figure 2 is the flow chart of IDGMM scoring algorithm in which the feature coefficient is exemplified by MFCC. It can be detailed as follows:

1) The feature vector series of test speech should be transformed by the decorrelation parameter of IDGMM, i.e. subtracted by the total mean vector and multiplied by the multiplication matrix on the left.
2) Each transformed dimension series (composed of samples) should be input to the corresponding scalar GMM. The probability of each sample in scalar GMM is computed, which forms the probability sequence of this transformed dimension series.
3) The total probability series is generated by the dot product of probability sequences of dimension series, and its temporal expectation is the final score.

2.3 The Decorrelation by Schmidt Orthogonalization

Suppose $X_i(i=1,2,\ldots,D)$be the ith dimension series of feature vector series which represents a row vector. The total mean vector is $[E(X_1),E(X_2),\ldots,E(X_D)]^T$ and $Y_i=X_i-E(X_i)$. Let Z be the decorrelated feature vector series whose column vector is the decorrelated feature vector, Z_i being the ith dimension series of Z. Then:

$$\mathbf{Z}_i = \mathbf{Y}_i - \sum_{k=1}^{i-1} \frac{<\mathbf{Y}_i,\mathbf{Z}_k>}{<\mathbf{Z}_k,\mathbf{Z}_k>} \mathbf{Z}_k . \tag{1}$$

By supposing $Y=[Y_1^T,Y_2^T,\ldots Y_D^T]^T$ $Z=[Z_1^T,Z_2^T,\ldots Z_D^T]^T$, the above equation could be expressed in matrix form:
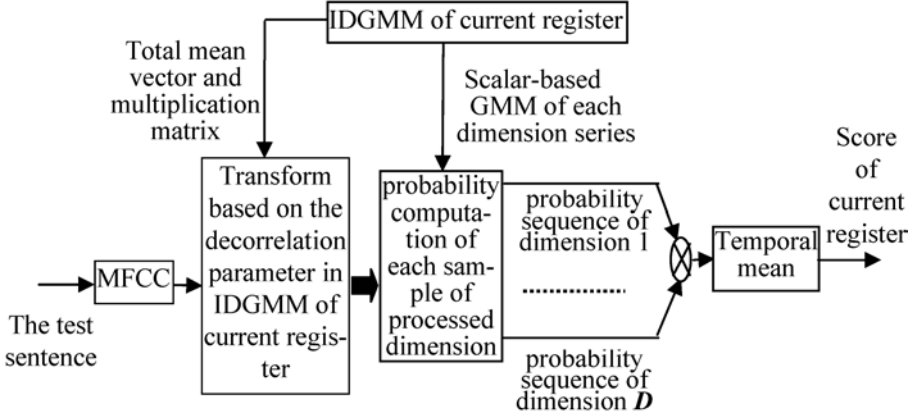
Fig. 2. The scoring algorithm of IDGMM in testing

$$Z = B^{-1} * Y . \tag{2}$$

$$B = \begin{pmatrix} 1 & 0 & 0 & ... & 0 \\ \dfrac{<Y_2,Z_1>}{<Z_1,Z_1>} & 1 & 0 & ... & 0 \\ \dfrac{<Y_3,Z_1>}{<Z_1,Z_1>} & \dfrac{<Y_3,Z_2>}{<Z_2,Z_2>} & 1 & ... & 0 \\ ... & ... & ... & ... & 0 \\ \dfrac{<Y_D,Z_1>}{<Z_1,Z_1>} & \dfrac{<Y_D,Z_2>}{<Z_2,Z_2>} & ... & \dfrac{<Y_D,Z_{D-1}>}{<Z_{D-1},Z_{D-1}>} & 1 \end{pmatrix} . \tag{3}$$

The vector $[E(X_1),E(X_2),…,E(X_D)]^T$ and matrix B^{-1} is the total mean vector and multiplication matrix of decorelation parameter in IDGMM, respectively.

By mathematical induction, it's easy to be proved that $E(Z_i)=0$ and $<Z_i-E(Z_i), Z_j-E(Z_j)>=<Z_i, Z_j>=0$ $(i ,j=1,2,…,D, i\neq j)$. According to estimation theory, it suggests that the covariance matrix must be diagonal, which means the realization of the decorrealtion.

As the replacement of the traditional decorrelation method, PCA, Schmidt orthogonalization is much easier in realization. In comparison to the complicate computation in finding the eigenvalue and the eigenvector of PCA, the inversion computation on the lower triplet matrix of Schmidt orthogonalization is easier for realization.

2.4 Mathematic Morphology (MM) Filter for MCN Decision

It's known that the model duration is in proportion to the MCN. To accelerate the model, we suggest that MCN should be decided only by the major peak number in the original histogram, i.e. the peak number of the histogram filtered by MM filter. The reason for such process is that most peaks and valleys of the original histogram play trivial roles in describing the histogram's major trend.

The trivial peak and valley (positive and negative burr with a width of 3 dots) is better to be filtered by MM, a nonlinear filter specialized in dot processing.

Suppose $f(n)$ be the source signal, $k(n)$ be the structuring function of MM. The basic MM operator [4] with idempotence are *Opening*:

$$f \circ k = (f \Theta k) \oplus k . \tag{4}$$

and *Closing*:

$$f \bullet k = (f \oplus k)\Theta k \quad \cdot \tag{5}$$

Where:

$$(f \oplus k)(m) = \max_{n=m-M+1,\cdots,m} (f(n)+k(m-n)) \qquad (m = M-1, M,\cdots, N-1) \quad \cdot \tag{6}$$

$$(f\Theta k)(m) = \min_{n=0,\cdots,M-1} (f(m+n)-k(n)) \qquad (m = 0,1,\cdots, N-M) \quad \cdot \tag{7}$$

When $k(n)$ is a flat signal with M width, the peaks or valleys with width less than M in $f(n)$ will be filtered after *Opening* or *Closing* respectively. To extract the major trend (removing the peak and the valley meanwhile) we suggest that the MM filter should be the mean of the output of "*Closing* after *Opening*" and "*Opening* after *Closing*".

In our algorithm, $f(n)$ should be the histogram, $k(n)$ should be a flat series with 3 width. As the width of filtered signal is shorter than the original one for M dots, it's better to repeat periodically at the both endpoints in advance [5]. In addition, N (the number of interval between the maximum and minimum of the histogram) should be about one of hundred of the number of sample dots (i.e. the number of feature vectors), which results form the experience from the digital image processing.

For a decorrelated dimension series of a given register, the original histogram, the histogram filtered by MM and the PDF fitted by the scalar-based GMM are all illustrated in the figure 3. Its horizontal and vertical axis scale is the medium of each interval of histogram and the occurrence frequency respectively. The major peak is only 2 of 9 peaks in histogram. The scalar-based GMM according to the 2 mixture components fits the original histogram fairly well.
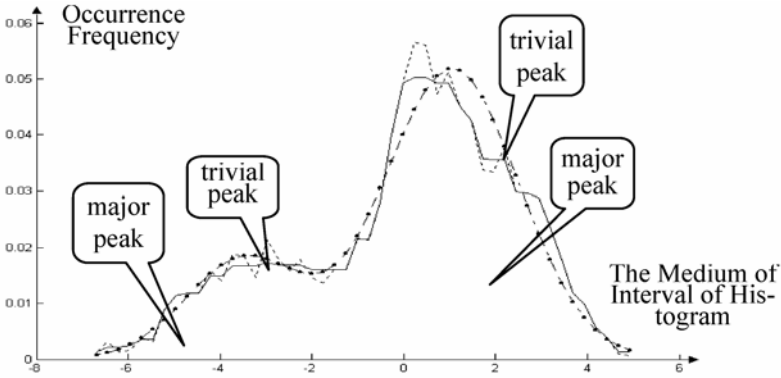

Fig. 3. The histogram(*broken line*), the histogram filtered by MM(*real line*) and the PDF expressed by scalar GMM(*dot-dashed line*)

3   Simulation Results

Our algorithm is applied to a close-set text-independent SI with a corpus of 48 registers in mandarin. The corpus is recorded by us and each sentence is sampled at 8kHZ.

Each register records 11 sentences, the sentence 1 being figure series of "1234567890" while the other sentences being the paragraph selected from the news-papers randomly. The average length of each sentence is 3.9 seconds. For each register, the test sentence is sentences from sentence 1 to sentence 4. The train (model-purpose) sentences with different length are the connected sentences from sentence 8 to sentence 8/9/10/11/12/13/14 respectively corresponding to an average length of 10/14/18/22/26/30 seconds.

The train speech is pre-emphasized by $1-0.95z^{-1}$ firstly. The feature vector is the MFCC of dimension 16, representing the feature coefficient of a hamming-weighted speech frame with 20ms length and 10ms overlap. The MCN of GMM is 32. The EM iterative algorithm is set to terminate when the relative improvement of likelihood amounts to 10^{-6} or the iterative times amounts to 40. The same EM iterative algorithm and its parameter is used to the scalar-based GMM modeling.

The recognition rate of GMM, IDGMM without decorrelation (Schmidt orthogo-nalization), IDGMM without MM filter and IDGMM are shown in figure 4.

At the same time, we check the modeling speed of the 4 algorithms. It's coincident that no matter how long the duration of model-purpose speech is, the time consuming is about 1:0.25:1.1:0.25 corresponding to the GMM, IDGMM, IDGMM without MM filter and IDGMM without decorrelation (Schmidt orthogonalization).

Finally, we investigate the issue when the register number increases. For the case of training speech with 22-seconds length, the recognition rate of 16/24/32/40/48 register are shown in table 1.

Table 1. The recognition rate when the number of register increase

register number	recognition rate(%)
16	92.188%
24	92.708%
32	89.844%
40	90.625%
48	91.146%

It can be included as follows:

1) Decorrelation plays an important role. Without it, the recognition rate decreases remarkably for the omitting of inter-dimension identity information, which is il-lustrated by the fact that the graph of IDGMM without decorrelation is much lower than that of GMM. Moreover, when the train speech duration changes, some undesirable phenomena will emerge since the ratio of the inter-dimension identity information to the intra-dimension identity information may change un-predictively. For instance, in the graph of IDGMM without decorrelation, the rec-ognition rate corresponding to the duration of 26 or 30 seconds is obviously lower than those corresponding to shorter duration, which arises from the discard of in-ter-dimension identity information.

Although the feature coefficient used here (MFCC) is most close to decorrelated, but decorrelation is still useful. After all, DCT (the final step of MFCC) is not a data-based transform. The fixed transform kernel matrix cannot decorrelate the data as thoroughly as data-based transforms, say Schmidt Orthogonalization.
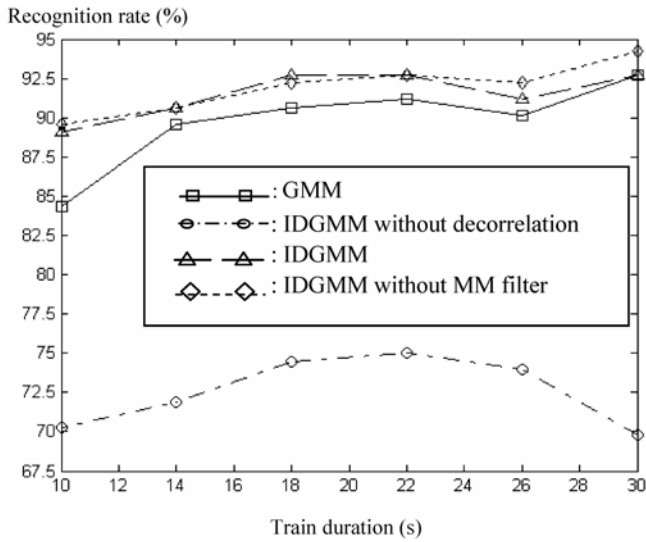
Fig. 4. Recognition rate

2) MCN decision by the peak number of histogram benefits the improvement on recognition rate whether the original one or the MM-filtered one, which validates the improvement on describe preciseness of the joint PDF mentioned above.
3) MM filter could reduce model duration remarkably (0.25:1.1) at a cost of slight declination in recognition rate in comparison to IDGMM without MM filter. So the IDGMM do make a balance between the recognition rate and the model speed. Its recognition rate is comparable to GMM while it saves a lot in the model duration.
4) The discriminative capability of IDGMM seems stable when the register number changes.

4   Conclusion

In this paper, a new algorithm named IDGMM is proposed. After the pretreatments of the decorrelation and the decision of MCN by the peak number of MM-filtered histogram, we can construct scalar-based GMM for each decorrelated dimension series independently without discarding the useful identity information. In this sense, each register can be modeled by the product of scalar-based GMMs.

IDGMM accelerates the modeling remarkably by discarding the inter-dimension information while obtains approximate recognition performance by pretreatments. When simulated, IDGMM does make a balance between the recognition rate and the model speed.

References

1. Ravi P. Ramachandran, Kevin R. Farrell, Roopashri Ramachandran, Richard J. Mammone,: Speaker recognition – general classifier approaches and data fusion methods. In: Patter Recognition(2002), Vol. 35, 2801 – 2821

2. Douglas A. Reynolds, Richard C. Rose: Robust text-independent speaker identification using Gaussian Mixture Speaker Models. In: IEEE Transactions on SPEECH AND AUDIO PROCESSING(1995), vol. 3, 72 – 83
3. Frederic Jauquet, Patrick Verlinde, Claude Vloeberghs: Histogram classifiers using vocal tract and pitch information for text independent speaker identification. In: Proceedings of the ProRISC Workshop on Circuits(1997), Systems and Signal Processing, 213 – 217
4. Chee-Hung Henry Chu, Edward J. Delp: Impulsive noise suppression and background normalization of electrocardiogram signals using morphological operators. In: IEEE Transactions on Biomedical Engineering(1989), vol. 36, 262 – 273
5. Gil, J.Y., Kimmel, R.: Efficient dilation, erosion, opening, and closing algorithms. In: IEEE Transactions on Pattern Analysis and Machine Intelligence(2002), vol.24, 1606 – 1617

Sensor Interoperability and Fusion in Signature Verification: A Case Study Using Tablet PC

Fernando Alonso-Fernandez, Julian Fierrez-Aguilar, and Javier Ortega-Garcia

Biometrics Research Lab. – ATVS, EPS, Universidad Autonoma de Madrid
Campus de Cantoblanco – C/ Francisco Tomas y Valiente 11, 28049 Madrid, Spain
{fernando.alonso,julian.fierrez,javier.ortega}@uam.es

Abstract. Several works related to information fusion for signature verification have been presented. However, few works have focused on sensor fusion and sensor interoperability. In this paper, these two topics are evaluated for signature verification using two different commercial Tablet PCs. An enrolment strategy using signatures from the two Tablet PCs is also proposed. Authentication performance experiments are reported by using a database with over 3000 signatures.

1 Introduction

Personal authentication in our networked society is becoming a crucial issue [1]. In this environment, automatic signature verification has been intensely studied due to its social and legal acceptance [2, 3]. Furthermore, the increasing use of portable devices capable of capturing signature signals (i.e. Tablet PCs, mobile telephones, etc.) is resulting in a growing demand of signature-based authentication applications.

Several works related to information fusion for signature verification have been presented [4–8]. However, few works have focused on sensor fusion and sensor interoperability [9]. In this work, we evaluate sensor interoperability and fusion using the ATVS Tablet PC signature verification system [10]. An enrolment strategy using signatures from the two Tablet PCs is also proposed.

The rest of the paper is organized as follows. The sensor interoperability and fusion topics are briefly addressed in Sects. 2 and 3, respectively. Experiments and results are described in Sect. 4. Conclusions are finally drawn in Sect. 5.

2 Sensor Interoperability

In biometrics, sensor interoperability can be defined as the capability of a recognition system to adapt to the data obtained from different sensors. Sensor interoperability has received limited attention in the literature [11]. Biometrics systems are usually designed and trained to work with data acquired using a unique sensor. As a result, changing the sensor affects the performance of the verification system. Martin et al. [12] reported a significant difference in performance when different microphones are used during the training and testing
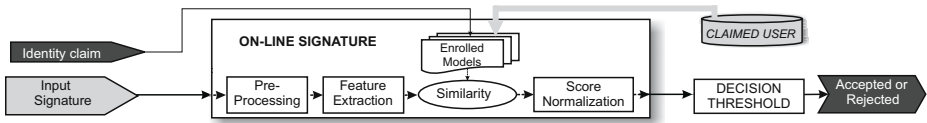
Fig. 1. System model for person authentication based on written signature

phases of a speaker recognition system. Ross et al. [11] studied the effect of matching fingerprints acquired with two different fingerprint sensors, resulting in an important drop of performance.

The sensor interoperability problem is also being addressed by standardization bodies. To standardize the content, meaning, and representation of biometric data formats, ISO and IEC have established the Sub-Committee 37 of the Joint Technical Committee 1 *JTC 1/SC 37* [13]. Different Working Groups constitute this Sub-Committee, including those related to biometric technical interfaces and data exchange formats.

3 Fusion of Sensors

Multibiometric systems refer to biometric systems based on the combination of a number of instances, sensors, representations, units and/or traits [14]. Several approaches for combining the information provided by these sources have been proposed in the literature [15, 16]. However, fusion of sensors has not been extensively analyzed. Chang et al. [17] studied the effect of combining 2D and 3D images acquired with two different cameras for face recognition. Marcialis et al. [18] reported experiments fusing the information provided by two different fingerprint sensors. No previous work on sensor fusion for signature verification has been found in the literature.

4 Experiments

4.1 On-Line Signature Verification System

The ATVS Tablet PC signature verification system is used in this work [10]. This system represents signatures as a set of 14 discrete-time functions, including coordinate trajectories, pressure and various dynamic properties. Given an enrolment set of signatures of a client, a left-to-right Hidden Markov Model (HMM) is estimated and used for characterizing the client identity. This HMM is used to compute the similarity matching score between a given test signature and a claimed identity. For more details, we refer the reader to [19, 20]. In Fig. 1, the overall system model is depicted.

4.2 Sensors

In this work, the following Tablet PCs have been used: *i*) Hewlett-Packard TC1100 with Intel Pentium Mobile 1.1 Ghz processor and 512 Mb RAM, see
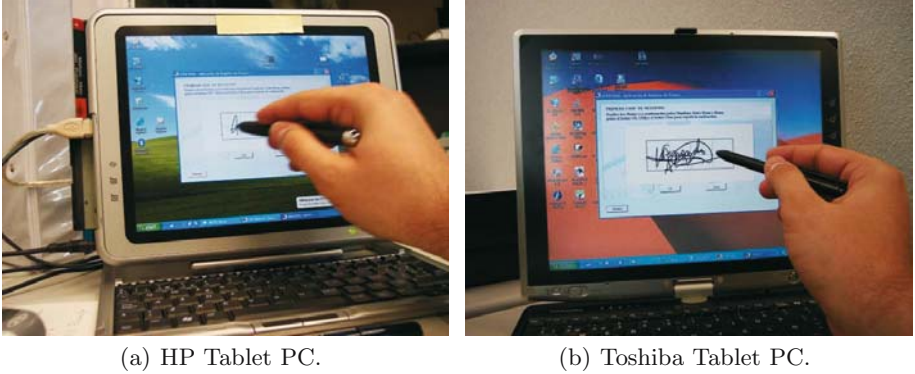
(a) HP Tablet PC.   (b) Toshiba Tablet PC.

Fig. 2. Tablet PCs considered in the experiments

Fig. 3. Signature examples of the ATVS Tablet PC Signature Database. For each row, the left half part (two signatures) corresponds to a subject captured with the HP TC 1100 Tablet PC and the right half part (two signatures) corresponds to the same subject captured with the Toshiba Portege M200 Tablet PC. For a particular Tablet, the left sample is a client signature and the right one is a skilled forgery

Fig. 2(a), and *ii*) Toshiba Portege M200 with Intel Centrino 1.6 Ghz processor and 256 Mb RAM, see Fig. 2(b). Both of them provide position in x- and y-axis and pressure p information.

Experiments reported in [10] showed that both Tablet PCs sample at a mean frequency of about 133 Hz but the instantaneous sampling period is not constant. Particularly in the HP Tablet PC, sampling period oscillates during the entire signature. To cope with this problem, the position and pressure signals have been downsampled to a constant sampling frequency of 100 Hz by using linear interpolation. Also, the two Tablet PCs provide 256 pressure values. More technical details about these sensors can be found in [10].

4.3   Database and Experimental Protocol

The ATVS Tablet PC Signature Database [10] has been used in this work. Some example signatures from this database are shown in Fig. 3. The database includes 53 users acquired with the two Tablet PCs introduced in Sect. 4.2. Each user produced 15 genuine signatures in 3 different sessions. For each user, 15 skilled forgeries were also generated by other users. Skilled forgeries were produced by observing both the static image and the dynamics of the signature to be imitated. More details regarding the acquisition protocol can be found in [10].
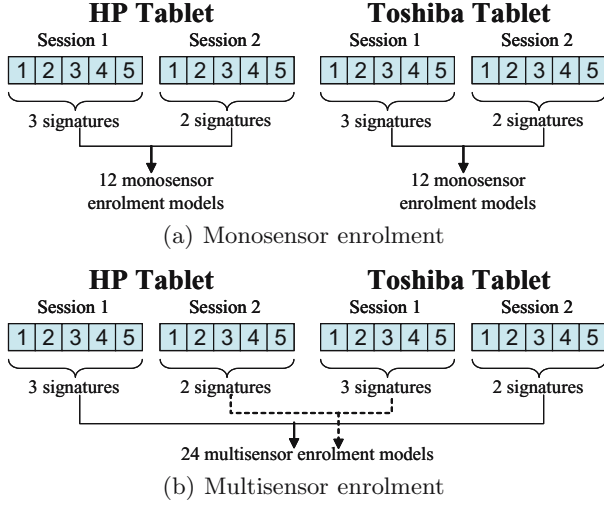
(a) Monosensor enrolment

(b) Multisensor enrolment

Fig. 4. Enrolment strategies considered in the experiments

Two different enrolment strategies are considered in this paper: *i*) *monosensor enrolment*, see Fig. 4(a), using 3 consecutive genuine signatures from the first session and 2 consecutive signatures from the second session, with both sessions from the same Tablet PC; and *ii*) *multisensor enrolment*, see Fig. 4(b), using the same scheme but taking one session from each Tablet. For each enrolment strategy, all the possible combinations are generated. The first strategy results in 12 different enrolment models per user and per Tablet. The second strategy results in 24 different enrolment models per user, all of them including information from the two sensors.

4.4   Results

Sensor Interoperability Experiments. We study the effects of sensor interoperability by considering the following experimental setup, see Fig. 5. The remaining genuine signatures of the third session of each Tablet PC have been matched against: *a*) their 12 monosensor enrolment models; *b*) the 12 monosensor enrolment models of the other Tablet PC; and *c*) the 24 multisensor enrolment models. For a specific target user, casual impostor test scores are computed by using the skilled forgeries from all the remaining targets. Real impostor test scores are computed by using the 15 skilled forgeries of each target. The first and second experiments result in $5 \times 12 \times 53 = 3180$ genuine scores, $15 \times 12 \times 53 = 9540$ real impostor scores and $15 \times 12 \times 52 \times 53 = 496080$ casual impostor scores for each Tablet PC. The third experiment results in $5 \times 24 \times 53 = 6360$ genuine scores, $15 \times 24 \times 53 = 19080$ real impostor scores and $15 \times 24 \times 52 \times 53 = 992160$ casual impostor scores for each Tablet PC.

Verification performance results are given in Fig. 6. We observe that when testing with the HP Tablet PC (Fig. 6(a)), verification performance is not much
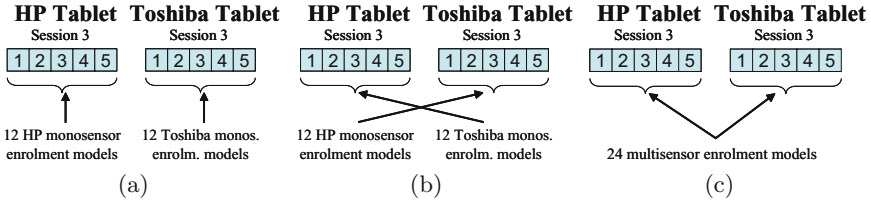
Fig. 5. Experiments evaluating interoperability

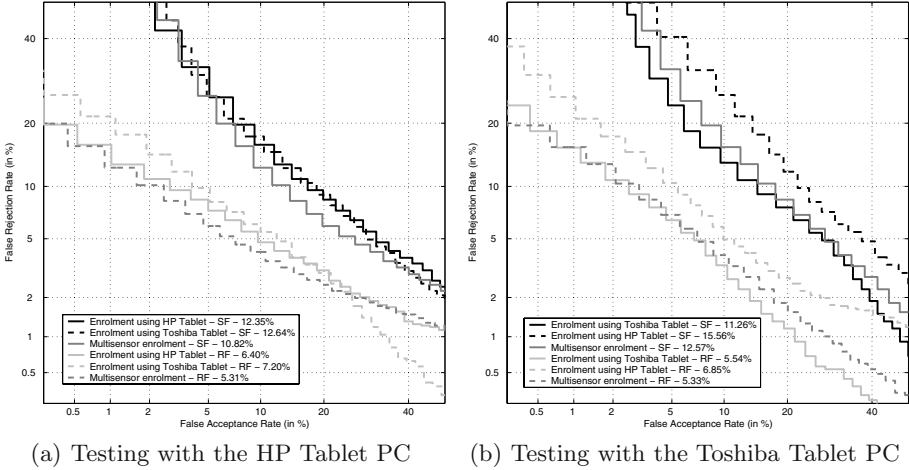(a) Testing with the HP Tablet PC   (b) Testing with the Toshiba Tablet PC

Fig. 6. Verification performance of experiments evaluating interoperability. EER values are also provided both for skilled forgeries (SF) and random forgeries (RF)

affected by the Tablet PC used for enrolment. This does not occur when testing with the Toshiba Tablet PC (Fig. 6(b)). In this case, using an enrolment model generated using the other sensor decreases the verification performance. This may be a result of the sampling frequency oscillation found in the HP Tablet PC [10]. As compared to monosensor enrolment, using multisensor enrolment results in better performance when testing with the HP Tablet PC, but worse when testing with the Toshiba Tablet PC. This may be also because enrolment models are corrupted by the *less reliable information* provided by the HP Tablet due to the above mentioned sampling frequency oscillation.

Sensor Fusion Experiments. In this case we compare fusion of two sensors with fusion of two instances of each sensor, in order to reveal the real benefits of considering information provided from different sensors [21]. Experiments using only one single instance from one sensor is also reported for comparison. The remaining five genuine signatures of the third session of each Tablet PC are used for testing. The experiments are as follows: *i) single instance from one sensor*, each one of the testing signatures of each Tablet is considered independently;
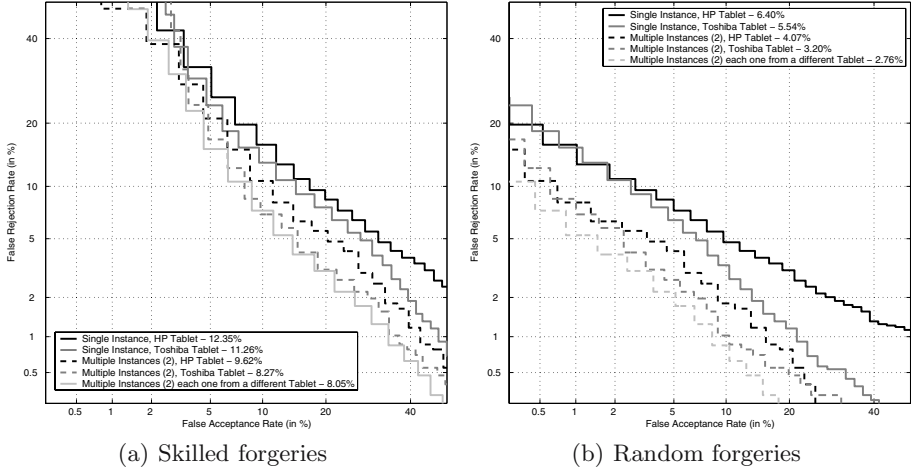
(a) Skilled forgeries (b) Random forgeries

Fig. 7. Verification performance of experiments evaluating fusion of sensors. EER values are also provided

ii) multiple instances from the same sensor, 2 groups of two different genuine signatures each are selected for each Tablet PC; and *iii) multiple instances from multiple sensors*, 5 groups of genuine signature pairs are selected, with one signature from a different Tablet for each group. In all experiments, each signature is matched against the monosensor models of its corresponding Tablet PC. The two similarity scores in each group for experiments *ii)* and *iii)* are combined using the max fusion rule. For a specific target user, a similar procedure is followed on each session of its skilled forgeries for obtaining real impostor test scores, and on each session of the skilled forgeries of the remaining targets for obtaining casual impostor test scores.

The first and third experiments result in $5 \times 12 \times 53 = 3180$ genuine scores, $15 \times 12 \times 53 = 9540$ real impostor scores and $15 \times 12 \times 52 \times 53 = 496080$ casual impostor scores for each Tablet PC. The second experiment results in $2 \times 12 \times 53 = 1272$ genuine scores, $2 \times 3 \times 12 \times 53 = 3816$ real impostor scores and $2 \times 3 \times 12 \times 52 \times 53 = 198432$ casual impostor scores for each Tablet PC.

Sensor fusion results are compared in Fig. 7. When considering only one sensor, the Toshiba Tablet results in better performance, either for single or multiple test instances. It is also remarkable the performance improvement obtained for the two Tablet PCs when using multiple instances from one sensor with respect to using a single instance. In addition, we observe that fusion of sensors outperforms the performance of the best individual sensor for random forgeries (3.20% to 2.76% EER) and an slightly improvement is also observed for skilled forgeries (8.27% to 8.05% EER). Interestingly, the relative improvement observed when fusing two sensors is not as high as the one provided by considering multiple instances in the same sensor.

5   Conclusions

Sensor interoperability and sensor fusion have been studied using the ATVS Tablet PC signature verification system. For this purpose, a database captured with the Hewlett-Packard TC1100 and Toshiba Portege M200 Tablet PCs has been used. Previous experiments have shown that instantaneous sampling period is not constant. In particular, the sampling frequency for the HP Tablet PC oscillates during the entire signature and because of that, it is referred as the *less reliable sensor* or the sensor providing the *less reliable information*.

Sensor interoperability experiments show that, when using the sensor providing *less reliable information*, verification performance is not much affected by the Tablet PC used for enrolment. This does not occur when testing with the *more reliable sensor*, where verification performance drops significantly if we use the other Tablet PC for enrolment. This results stresses the importance of having enrolment models generated with good quality data.

Regarding sensor fusion, a significant performance improvement is observed when using multiple test instances from the same sensor with respect to using a single instance. Moreover, using multiple sensors outperforms the performance of the individual sensors for random forgeries and an slightly improvement is also observed for skilled forgeries. Interestingly, this improvement observed when fusing information from two sensors is not as high as the one produced by the use of multiple instances from the same sensor. This result can be explained by the similar sensors used and the large intra-user variability found in signature verification. Therefore, the biggest improvement in performance is obtained by considering additional data from the user at hand, regardless of the sensor used for its acquisition. This result may be also extended to other behavioral biometrics, such as voice biometrics [22], and will be the source for future work.

Acknowledgments

References

1. Jain, A.K., Ross, A., Prabhakar, S.: An introduction to biometric recognition. IEEE Transactions on Circuits and Systems for Video Technology **14** (2004) 4–20
2. Plamondon, R., Lorette, G.: Automatic signature verification and writer identification - the state of the art. Pattern Recognition **22** (1989) 107–131
3. Plamondon, R., Srihari, S.N.: On-line and off-line handwriting recognition: A comprehensive survey. IEEE Trans. on PAMI **22** (2000) 63–84
4. Fierrez-Aguilar, J., Nanni, L., Lopez-Penalba, J., Ortega-Garcia, J., Maltoni, D: An on-line signature verification system based on fusion of local and global information. Proc. AVBPA, Springer LNCS, **3546** (2005) 523-532

5. Zhang, K., Nyssen, E., Sahli, H.: A multi-stage on-line signature verification system. Pattern Analysis and Applications **5** (2002) 288–295
6. Kashi, R., et al.: A Hidden Markov Model approach to online handwritten signature verification. Int. J. on Document Analysis and Recognition **1** (1998) 102–109
7. Fuentes, M., Garcia-Salicetti, S., Dorizzi, B.: On-line signature verification: Fusion of a Hidden Markov Model and a Neural Network via a Support Vector Machine. Proc. Int. Workshop on Frontiers of Handwriten Recognition **8** (2002) 253–258
8. Ly Van, B., et al.: Fusion of HMM's Likelihood and Viterbi Path for On-line Signature Verification. Proc. BioAW, Springer LNCS **3087** (2004) 318–331
9. Vielhauer, C., Basu, T., Dittmann, J., Dutta, P.K.: Finding metadata in speech and handwriting biometrics. Proc. SPIE Int. Soc. Opt. Eng. **5681** (2005) 504–515
10. Alonso-Fernandez, F., Fierrez-Aguilar, J., del-Valle, F., Ortega-Garcia, J.: On-line signature verification using Tablet PC. to appear in Proc. IEEE ISPA, Special Session on Signal and Image Processing for Biometrics (2005)
11. Ross, A., Jain, A.: Biometric sensor interoperability: A case study in fingerprints. Proc. BioAW, Springer LNCS **3087** (2004) 134–145
12. Martin, A., Przybocki, M., Doddington, G., Reynolds, D.: The NIST Speaker Recognition Evaluation - Overview, methodology, systems, results, perspectives. Speech Communication (2000) 225–254
13. ISO/IEC Joint Technical Committee 1 on Information Technology - www.jtc1.org
14. Jain, A.K., Ross, A.: Multibiometric systems. Communications of the ACM **47** (2004) 34–40
15. Kittler, J., Hatef, M., Duin, R., Matas, J.: On combining classifiers. IEEE Trans on PAMI **20** (1998) 226–239
16. Fierrez-Aguilar, J., Ortega-Garcia, J., Gonzalez-Rodriguez, J., Bigun, J.: Discriminative multimodal biometric authentication based on quality measures. Pattern Recognition **38** (2005) 777–779
17. Chang, K., Bowyer, K., Flynn, P.: An evaluation of multimodal 2D+3D face biometrics. IEEE Trans on PAMI **27** (2005) 619–624
18. Marcialis, G., Roli, F.: Fingerprint verification by fusion of optical and capacitive sensors. Pattern Recognition Letters **25** (2004) 1315–1322
19. Ortega-Garcia, J., Fierrez-Aguilar, J., Martin-Rello, J., Gonzalez-Rodriguez, J.: Complete signal modelling and score normalization for function-based dynamic signature verification. Proc. AVBPA, Springer LNCS **2688** (2003) 658–667
20. Fierrez-Aguilar, J., Ortega-Garcia, J., Gonzalez-Rodriguez, J.: Target dependent score normalization techniques and their application to signature verification. IEEE Trans. on SMC-C, Special Issue on Biometric Systems **35** (2005)
21. Fierrez-Aguilar, J., Ortega-Garcia, J., Gonzalez-Rodriguez, J., Bigun, J.: Kernel-based multimodal biometric verification using quality signals. Biometric Technologies for Human Identification, Proc. SPIE **5404** (2004) 544–554
22. Bimbot, F., Bonastre, J.-F., Fredouille, C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-Garcia, J., Petrovska-Delacretaz, D., Reynolds, D.-A.: A tutorial on text-independent speaker verification. Journal on Applied Signal Processing **2004:4** (2004) 430-451

Fusion of Local and Regional Approaches for On-Line Signature Verification

Julian Fierrez-Aguilar[1,*], Stephen Krawczyk[2,3],
Javier Ortega-Garcia[1], and Anil K. Jain[2]

[1] ATVS, Escuela Politecnica Superior, Universidad Autonoma de Madrid,
Avda. Francisco Tomas y Valiente 11, Campus de Cantoblanco, 28049 Madrid, Spain
{julian.fierrez,javier.ortega}@uam.es
[2] Department of Computer Science and Engineering, Michigan State University,
East Lansing, MI 48823, USA
jain@cse.msu.edu
[3] Information Technology Division, Naval Research Laboratory,
Washington DC, USA
stephen.krawczyk@nrl.navy.mil

Abstract. Function-based methods for on-line signature verification are studied. These methods are classified into local and regional depending on the features used for matching. One representative method of each class is selected from the literature. The selected local and regional methods are based on Dynamic Time Warping and Hidden Markov Models, respectively. Some improvements are presented for the local method aimed at strengthening the performance against skilled forgeries. The two methods are compared following the protocol defined in the Signature Verification Competition 2004. Fusion results are also provided demonstrating the complementary nature of these two approaches.

1 Introduction

The goal of biometrics is to infer the identity of people based on anatomical or behavioral data (e.g., fingerprint, face, palmprint, signature, or voice) [1]. The current interest in biometrics is due to the increasing number of important applications where an automatic assessment of identity is crucial. Within biometrics, automatic signature verification has been an intense research area because of the social and legal acceptance and widespread use of the written signature as a personal authentication method [2], and still is a challenging problem. This is mainly due to the large intra-class variations and, when considering forgeries, small inter-class variations. See Fig. 2 for one example Chinese signature where this effect is evident.

This work deals with on-line signature verification, where the time functions of the dynamic signing process are available for recognition (e.g., position trajectories, or pressure versus time). Different approaches have been considered

---

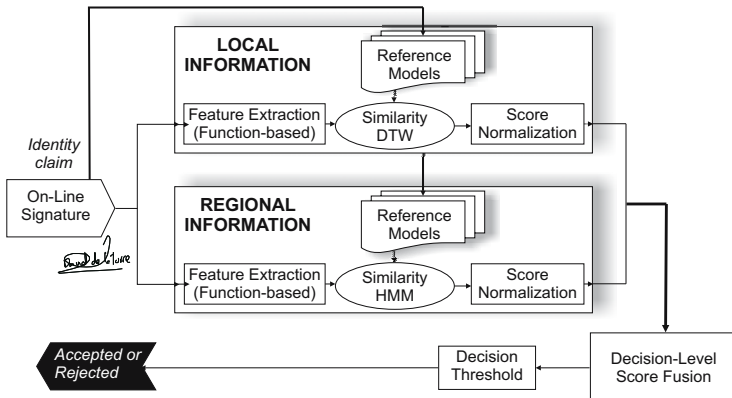* This work has been carried out while J. F.-A. was on a research stay at Michigan State University.

Fig. 1. System model of multilevel signature verification

in the literature in order to extract relevant information from on-line signature data [3]; they can broadly be divided into: *i*) feature-based approaches, in which a holistic vector representation consisting of a set of global features is derived from the signature trajectories [4–7], and *ii*) function-based approaches, in which time sequences describing local properties of the signature are used for recognition (e.g., position trajectory, velocity, acceleration, force, or pressure) [5–12]. Although recent work shows that feature-based approaches are competitive with respect to function-based methods in some conditions [7], the latter methods have traditionally yielded better results [3].

Function-based approaches can be classified into local and regional methods. In local approaches, the time functions of different signatures are directly matched by using elastic distance measures such as Dynamic Time Warping (DTW) [9–11]. In regional methods, the time functions are converted to a sequence of vectors describing regional properties. One of the most popular regional approaches is the method based on Hidden Markov Models (HMM) [5, 7, 8, 12].

One major research trend in biometric verification is the successful exploitation of the different information levels embodied in the biometric signals [13]. This is usually done by combining the confidences provided by systems that utilize the different information levels in the form of similarity scores [14, 15] (see Fig. 1 for the system model). Examples can be found regarding other biometrics like fingerprint, where the combined use of local (e.g., minutiae-based) and global (e.g, ridge-based) approaches has been shown to improve verification performance [16], or speaker verification, where multilevel approaches are currently the state-of-the-art [17]. Some work on multi-level signature verification is available in [5–7].

The purpose of this work is to compare two representative local and regional approaches from the literature and to fuse them in order to study the benefits of their combination. In order to do so, we adhere to the standard benchmark defined in the First International Signature Verification Competition 2004 (SVC2004) [18]. In the case of the local DTW system, we also present

some improvements over published work related to feature extraction and score normalization.

The paper is structured as follows. The system based on local information is described in Sect. 2 with emphasis on the improvements over published work. The regional system is briefly sketched in Sect. 3. The decision-level combination strategies are presented in Sect. 4. Experimental procedure and results are given in Sect. 5. Conclusions are finally drawn in Sect. 6.

2   System Based on Local Information (DTW)

An extension of the signature verification algorithm proposed by Jain et al. [10] was designed and developed. The original algorithm is based on the string matching technique called Dynamic Time Warping (DTW). Modifications of the original system were made to the preprocessing, feature extraction, and matching components. Each of the changes made had the intention of relying more on the temporal information as opposed to the spatial information of the signature. Here, we briefly outline the changes made to the original algorithm. For more details, we refer the reader to [19].

Preprocessing. Originally, the x and y signals of the signature were resampled in order to provide equidistant spacing between points in the signals. When resampling is performed, some temporal information is lost regarding the speed of the signature. In order to strengthen the system against skilled forgeries, we consider the speed of the signature as a feature by not resampling the input signatures. The only preprocessing steps performed are position normalization and stroke concatenation.

Feature Extraction. Three features are extracted at each point in the signature; δx, δy, and p. These features correspond to the change in the x direction, change in the y direction, and the pressure value, respectively. The feature values are normalized using signature-specific z-score normalization.

Matching. The matching algorithm is the standard DTW algorithm. The only differences with respect to the algorithm in [10] are related to the spurious and missing penalties, which we assume are equivalent. We refer to both penalties as the gap penalty G_P. Instead of imposing a constant penalty for missing or spurious points, the penalty is assigned based on the distance of the two feature vectors being compared, multiplied by a constant, i.e., $G_P(i,j) = c \cdot d_E(i,j)$, where $d_E(i,j)$ is the Euclidean distance between the feature vector of point i in the input signature and point j in the reference signature.

Score Normalization. We also added user-dependent score normalization to the new system by using the procedure described by Kholmatov et al. [11]. This method characterizes each matching with a three-dimensional score vector including maximum, minimum, and average distance information to the set of training signatures. This vector is then projected to one dimension with Principle Component Analysis (PCA).

3 System Based on Regional Information (HMM)

The regional system follows closely the system submitted by the authors to SVC 2004 [18], where it was ranked first and second for random and skilled forgeries, respectively (in Task 2, where signature data included position, pen orientation and pressure functions). The main difference of the HMM in the present work is the lack of user-dependent score normalization. A brief sketch of the system is given in this section, for more details we refer the reader to [12].

Feature Extraction. Signature trajectories are first preprocessed with position and rotation normalization. The signature is parameterized as a set of 7 discrete-time functions $\{x, y, p, \theta, v, \rho, a\}$, and first-order time derivatives of all of them, totaling 14 discrete functions; p, θ, v, ρ and a stand respectively for pressure, path tangent angle, path velocity magnitude, log curvature radius and total acceleration magnitude. A z-score transformation is finally applied to each discrete-time function.

Matching. Left-to-right Hidden Markov Models with no transition skips between states are used for identity modeling. A reduced number of states and multivariate Gaussian Mixture density for modeling the observations are used.

4 Fusion Strategies

Two theoretical frameworks for combining matchers with application to biometric verification are described in [14] and [15]. More recent approaches are reviewed in [1]. It is now generally agreed that the weighted average is a good way of combining the similarity scores provided by the different systems.

In this work, fusion strategies based on the max, product, and sum rules are compared [15]. Similarity scores given by the local and regional systems are normalized before fusion into the range $[0, 1]$ with logistic functions, in order to overcome the problem of score outliers [20].

5 Experiments

5.1 Database and Experimental Protocol

Not many signature databases are publicly available for research purposes [21]. As a result, the common practice in on-line signature recognition research is to evaluate the proposed recognition strategies on small data sets acquired at individual research laboratories. In this environment, the First International Signature Verification Competition (SVC) was organized in 2004 to provide a common reference for system comparison on the same signature data [18].

Development corpus of the Task 2 in SVC2004 is used in the experiments. The on-line signature data in this case include not only position coordinates (as in Task 1) but also pressure and pen angles at a sampling frequency equal to 100
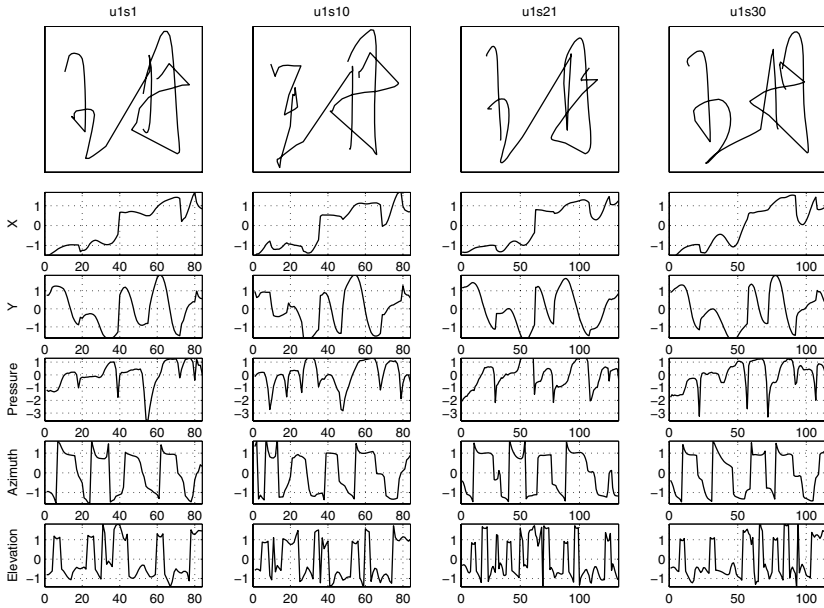
Fig. 2. Signature examples from SVC 2004 corpus. Two genuine signatures (left two columns) and two skilled forgeries (right two columns) are given along with their temporal trajectories for x and y position, pressure and pen angles

Hz. Pen angles are not used in our study, as they probed to be highly unstable in previous work [7]. The corpus consists of 40 sets of signatures. Each set contains 20 genuine signatures from one subject (acquired in two separate sessions) and 20 skilled forgeries from five other subjects. The signatures are mostly in either English or Chinese. Some examples are depicted in Fig. 2 for one specific signer.

Training signatures are randomly selected from the first acquisition session (5 in the competition, from 3 to 10 in this work). 10 runs of this random selection are conducted. For each user, the 10 genuine signatures of the second session and the 20 skilled forgeries are used for testing. In case of random forgeries, one random genuine signature from each of 20 different subjects randomly selected is used as impostor data. Whenever randomness was involved, the same random sets are used for the two systems under study.

Overall system performances using *a posteriori* user-independent decision thresholds are reported by means of DET plots [22]. Average EER tables for *a posteriori* user-dependent decision thresholds are also given. For more details on *a priori* and *a posteriori* decision thresholding techniques and their application to signature verification, we refer the reader to [12].

Finally, we must also stress the difficulty of the tasks proposed in SVC as compared to previous reported work on signature verification. This is mainly because: *i*) data was acquired by using a pen tablet (WACOM Intuos) with an inkless pen, *ii*) signers were asked to create new signatures different to the ones used in daily life in order to protect their privacy, *iii*) forgers were given not

Table 1. Comparative experiments. Verification performance with 3, 5 and 10 training signatures for *a posteriori* user-independent and user-dependent decision thresholds, both for skilled and random forgeries. Average EER is reported in %

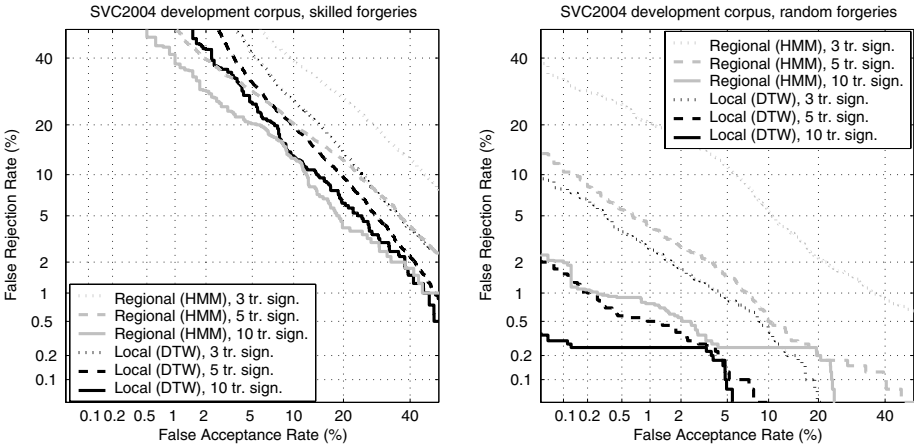|		Local (DTW)			Regional (HMM)		
		3 sign.	5 sign.	10 sign.	3 sign.	5 sign.	10 sign.
skilled	user-indep.	17.02	14.26	11.57	23.69	15.04	11.56
	user-dep.	12.98	10.20	6.76	11.56	8.04	4.50
random	user-indep.	1.83	0.57	0.24	7.64	2.41	0.69
	user-dep.	0.96	0.24	0.24	0.90	0.34	0.00

Fig. 3. Verification performance of the local and regional approaches under study on SVC2004 development corpus following SVC2004 protocol

only the shape but also the dynamics of the signatures to forge, and *iv*) training data is from one session and testing data is from a different session separated from the first session by at least one week.

5.2   Results

Verification performances of local and regional systems are given in Table 1. We observe a progressive improvement in performance as we increase the number of training signatures, especially for the HMM system. Utilizing user-dependent decision thresholds leads to significantly lower error rates than user-independent thresholds for both systems in the case of skilled forgeries. This effect is even more pronounced in the case of random forgeries for the HMM system, whereas it is not present in the DTW system, especially for large training set sizes. This can be explained by the user-dependent score normalization stage introduced in the DTW system, and the lack of user-dependent score normalization in this version of the HMM system. Comparatively, the DTW system outperforms the HMM system in almost all test conditions considering user-independent thresholds,

Table 2. Fusion experiments. Verification performance with 5 training signatures for *a posteriori* user-independent and user-dependent decision thresholds, both for skilled and random forgeries. Best SVC2004 refers to the best system competing in SVC2004 on this task and dataset. Average EER is reported in %

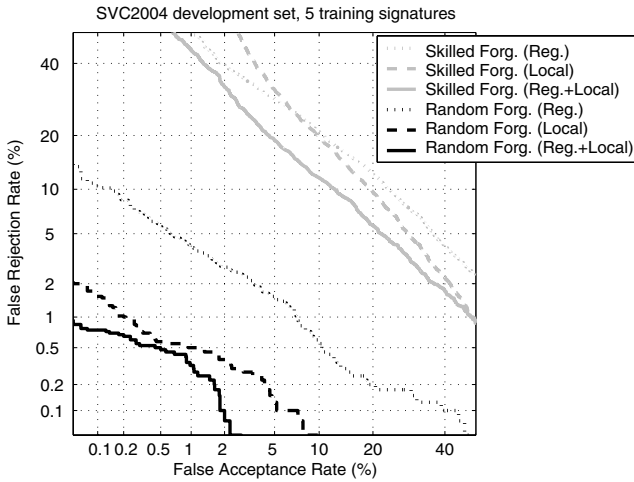|		Local	Regional	Max(L,R)	Prod(L,R)	Sum(L,R)	Best SVC2004
skilled	u-indep.	14.26	15.04	12.60	10.95	**10.91**	6.90
	u-dep.	10.20	8.04	7.69	7.60	**6.91**	
random	u-indep.	0.57	2.41	1.31	0.44	**0.49**	3.02
	u-dep.	0.24	0.34	0.19	0.14	**0.15**	

Fig. 4. Verification performance of the local and regional approaches under study as well as their combination on SVC2004 development corpus

especially for low training set sizes. The opposite occurs when employing user-dependent thresholds, especially for large training set sizes.

Detection error trade-off curves for the two competing systems are given in Fig. 3 both for skilled (left) and random forgeries (right).

Fusion results following SVC 2004 protocol are given in Table 2 (5 training signatures). The two systems are shown to provide complementary information for the verification task, which is well exploited with the three reported fusion strategies, especially the sum rule. Results of the SVC2004 winner system [11] following the same experimental protocol are also given for reference. The two studied systems are worse than the SVC2004 winner for skilled forgeries when considered independently, but their combination results in similar performance when considering user-dependent thresholds. In case of random forgeries, the combination of DTW and HMM significantly outperforms the SVC2004 winner. These results motivate the development of a hybrid system combining the DTW and HMM systems followed by user-dependent score normalization [12].

In Fig. 4 we depict detection error trade-off curves of the individual systems and their combination through the sum rule both for skilled and random forgeries.

6   Conclusions

Function-based approaches for on-line signature verification have been studied. These have been classified into local and regional approaches depending on the feature extraction process. Local and regional representative methods from the literature based on Dynamic Time Warping and Hidden Markov Models, respectively, have been selected and compared following the SVC2004 protocol. In case of the DTW system, some improvements aimed at strengthening the system against skilled forgeries have been introduced.

The enhanced DTW system including user-dependent score normalization appears to be quite robust to the common user-dependencies found in signature verification regarding score distribution miss-alignment between different users, especially for random forgeries when the number of training signatures increases. Comparatively, the DTW system outperformed the HMM system in almost all test conditions considering user-independent thresholds, especially for low training set sizes. The opposite occurred when employing user-dependent thresholds, especially for large training set sizes. The two systems are also shown to provide complementary information which is well exploited with the sum fusion rule.

The difficulty of the tasks proposed in SVC2004 comes from several factors, including: i) no visual feedback when writing, ii) subjects using invented signatures different to the ones used in daily life, iii) skilled forgers imitating not only the shape but also the dynamics, and iv) time variability between training and testing (at least one week). These factors affect significantly the performance of signature verification systems and are not well studied in the literature. Future work will be focused on characterizing the behavior of representative methods from the literature to these effects. In order to do so, collection of large biometric databases including these factors under controlled conditions is one of our priorities.

Acknowledgements

References

1. Jain, A.K., Ross, A., Prabhakar, S.: An introduction to biometric recognition. IEEE Trans. on Circuits and Systems for Video Technology **14** (2004) 4–20
2. Plamondon, R., Srihari, S.N.: On-line and off-line handwriting recognition: A comprehensive survey. IEEE Trans. on PAMI **22** (2000) 63–84

3. Plamondon, R., Lorette, G.: Automatic signature verification and writer identification - The state of the art. Pattern Recognition **22** (1989) 107–131
4. Lee, L.L., Berger, T., Aviczer, E.: Reliable on-line human signature verification systems. IEEE Trans. on Pattern Anal. and Machine Intell. **18** (1996) 643–647
5. Kashi, R.S., Hu, J., Nelson, W.L., Turin, W.: On-line handwritten signature verification using Hidden Markov Model features. In: Proc. of ICDAR. (1997) 253–257
6. Zhang, K., Nyssen, E., Sahli, H.: A multi-stage on-line signature verification system. Pattern Analysis and Applications **5** (2002) 288–295
7. Fierrez-Aguilar, J., Nanni, L., Lopez-Penalba, J., Ortega-Garcia, J., Maltoni, D.: An on-line signature verification system based on fusion of local and global information. In: Proc. of AVBPA, Springer LNCS-3546 (2005) 523–532
8. Yang, L., Widjaja, B.K., Prasad, R.: Application of Hidden Markov Models for signature verification. Pattern Recognition **28** (1995) 161–170
9. Nalwa, V.S.: Automatic on-line signature verification. Proceedings of the IEEE **85** (1997) 215–239
10. Jain, A.K., Griess, F., Connell, S.: On-line signature verification. Pattern Recognition **35** (2002) 2963–2972
11. Kholmatov, A., Yanikoglu, B.: Biometric authentication using online signatures. In: Proc. ISCIS, Springer LNCS-3280 (2004) 373–380
12. Fierrez-Aguilar, J., Ortega-Garcia, J., Gonzalez-Rodriguez, J.: Target dependent score normalization techniques and their application to signature verification. IEEE Trans. on Systems, Man and Cybernetics, part C **35** (2005) 418–425.
13. Jain, A.K., Ross, A.: Multibiometric systems. Communications of the ACM **47** (2004) 34–40
14. Bigun, E.S., Bigun, J., Duc, B., Fischer, S.: Expert conciliation for multi modal person authentication systems by Bayesian statistics. In: Proc. of AVBPA, Springer LNCS-1206 (1997) 291–300
15. Kittler, J., Hatef, M., Duin, R., Matas, J.: On combining classifiers. IEEE Trans. on Pattern Anal. and Machine Intell. **20** (1998) 226–239
16. Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: Handbook of Fingerprint Recognition. Springer (2003)
17. Fierrez-Aguilar, J., Garcia-Romero, D., Ortega-Garcia, J., Gonzalez-Rodriguez, J.: Speaker verification using adapted user-dependent multilevel fusion. In: Proc. of MCS, Springer LNCS-3541 (2005) 356–365
18. Yeung, D.Y., Chang, H., Xiong, Y., George, S., Kashi, R., Matsumoto, T., Rigoll, G.: SVC2004: First International Signature Verification Competition. In: Proc. of ICBA, Springer LNCS-3072 (2004) 16–22
19. Krawczyk, S.: User authentication using on-line signature and speech. Master's thesis, Michigan State University, Dep. of Computer Science and Engineering (2005)
20. Jain, A., Nandakumar, K., Ross, A.: Score normalization in multimodal biometric systems. Pattern Recognition (2005) (to appear).
21. Ortega-Garcia, J., et al.: MCYT baseline corpus: A bimodal biometric database. IEE Proc. Vision, Image and Signal Processing **150** (2003) 395–401
22. Martin, A., et al.: The DET curve in assessment of decision task performance. In: Proc. of EuroSpeech. (1997) 1895–1898

Text-Independent Writer Identification
Based on Fusion of Dynamic and Static Features

Wenfeng Jin[1], Yunhong Wang[2], and Tieniu Tan[1]

[1] Center for Biometrics and Security Research (CBSR)
National Lab of Pattern Recognition (NLPR)
Institute of Automation, Chinese Academy of Sciences
P.O. Box 2728, Beijing, 100080, P. R. China
{wfjin,tnt}@nlpr.ia.ac.cn
[2] School of Computer Science and Engineering
Beihang University, Beijing, 100083, P. R. China
yhwang@buaa.edu.cn

Abstract. Handwriting recognition is a traditional and natural approach for personal authentication. Compared to signature verification, text-independent writer identification has gained more attention for its advantage of denying imposters in recent years. Dynamic features and static features of the handwriting are usually adopted for writer identification separately. For text-independent writer identification, by using a single classifier with the dynamic or the static feature, the accuracy is low, and many characters are required (more than 150 characters on average). In this paper, we developed a writer identification method to combine the matching results of two classifiers which employs the static feature (texture) and dynamic features individually. Sum-Rule, Common Weighted Sum-Rule and User-specific Sum-Rule are applied as the fusion strategy. Especially, we gave an improvement for the user-specific Sum-Rule algorithm by using an error-score. Experiments were conducted on the NLPR handwriting database involving 55 persons. The results show that the combination methods can improve the identification accuracy and reduce the number of characters required.

1 Introduction

Handwriting recognition is a traditional and natural way for personal authentication. Extensive research has been done in recent years [1][2][3][4]. Nowadays, pocket PC, cell phone and PDA play important roles in our daily life. Online handwriting is easier to obtain and process in those equipments. So personal authentication based on handwriting have many potential applications.

So far, two different kinds of methods find their ways in handwriting authentication: text-dependent way and text-independent way. Text-independent methods have prominent advantages over text-dependent cases in specific conditions. Text with the same content is not required for the training samples and the testing ones, and the natural handwriting in a wide range can be dealt with [2]. It is difficult for the imposter to imitate the handwriting of others. Therefore, text-independent methods have gained more and more attention in recent years.

Some significant methods for text-independent cases have been proposed, such as using connected-component contours and edge-based features [19], using dynamic

features on strokes [8][13][14] or on hinge angles [12][15], using texture feature [9][10][20], using innovative binarised features [6], using moment feature [11], etc. Furthermore, some classical methods for character recognition are helpful, such as statistical character structure modeling [15], dynamic time warping (DTW) or cluster generative statistical dynamic time warping (CSDTW) [2] and so on. In these methods, dynamic or static features are widely adopted to discriminate the handwriting. They have been proved to be feasible, but some disadvantages still exist.

According to the previous research of other researchers, dynamic features and static features have been found representative for one's individual writing style. Dynamic features are micro-scale features of the handwriting, including the pressure, the azimuth, the velocity, the altitude, the relative time and so on. Static features are mostly macro-scale features, including the texture of the script, the character structure, etc. They are all available in writer identification.

Writer identification using single feature has some distinct limitations. The circumstances, emotions and physical state of writers while writing all affect the performance. The effect is not the same for different writers. For some writers, the variation in macro-scale features is larger than in micro-scale features, but for others, that may be opposite. Combining the classifiers adopting the two different scale features can form a comprehensive sight for the problem, and some of the limitations can be overcome. So using a combination method for writer identification is proposed here.

In this paper, Sum-Rule, Weighted Sum-Rule and User-specific Sum-Rule [21] are applied to combine two classifiers employing the texture feature of the handwriting scripts (macro-scale) and dynamic features (micro-scale). Especially, we proposed an improvement for the User-specific Sum-Rule algorithm by using an error-score.

The remainder of the paper is organized as follows. Section 2 introduces the NLPR handwriting database. In Section 3, the algorithm employing dynamic features for writer identification is introduced. In Section 4, we describe the writer identification algorithm with static features. In Section 5, combination methods are used to combine the two individual classifiers established in Section 3 and Section 4. In Section 6, experiments and results are given. In the end, we draw some conclusions in Section 7.

2   Handwriting Database

In this paper, all experiments are carried out on the NLPR handwriting database. In this database, samples are collected using a Wacom Intuos2 tablet. The tablet has 1024 levels of pressure sensitivity, and its maximum data rate is 200pps. 55 persons were required to write three different pages including approximately 600 Chinese characters altogether. The content of the handwriting is chosen by writers freely, so the database collected by this manner is text-independent handwriting database. The pressure, the azimuth, the velocity, the altitude, the coordinates and the relative time are recorded as data.

3   Writer Identification Using Dynamic Features

In the past few years, the digital handwriting collecting device and interface have been developed rapidly [3]. The dynamic features of handwriting could be collected accurately and conveniently. The pressure, the azimuth, the velocity, the altitude, the

relative time, the coordinates are closely related to the writing habit of the writer. So they are selected to represent the characteristic of writers. Here, the basic steps of writer identification using dynamic features are given. Details may be found in [8].

3.1 Preprocessing

The original data collected from the tablet includes many fake strokes and noises. After the preprocessing of wiping off fake strokes, line separation and stroke separation, the real strokes are classified into 12 primary stroke types [8]. For the statistical results, these 12 primary stroke types are the most common ones in Chinese characters. The primary stroke types are shown in Table 1.

Table 1. The 12 Primary Stroke Types

3.2 Feature Extraction

Features are extracted from the 12 primary types of strokes. The pressure, the azimuth, the velocity and the altitude are employed in this algorithm. We found that the histograms of each stroke type indicate that the attributes of each cluster could be modeled as Gaussian distribution [8]. Therefore, four Gaussian models are constructed from the pressure, the azimuth, the velocity and the altitude. The four models are used on each stroke type and each model could be represented by its mean value and variance. So the feature vectors with 96 dimensions($4{\times}2{\times}12$) for each writer are gained.

3.3 Identification

Feature vectors of the testing data could be extracted as well as the training data. The identification of writers based on given feature vectors is a typical recognition problem. Any classifiers can be used here. For simplicity and efficiency, the Nearest-Neighbor Classifier is selected here.

4 Writer Identification Using Static Features

Texture feature is a very important feature for a handwriting script [10]. It can represent the characteristic of writers. Using texture feature requires no segmentation of connected component analysis, and does not need the fixed content. With the advantages, texture feature is used in text-independent writer identification. Further details about the following steps may be found in [9].

4.1 Handwriting Script Reconstruction

The dynamic information is collected from the tablet. It is stored as time series. The script is concerned in this algorithm, so to reconstruct script from the dynamic infor-

mation is the first step. Conforming to the dynamic information, the real strokes are repainted following the relative time of each point. The points belong to the same stroke should be connected in order. Then the binary images of the handwriting script are available.

4.2 Preprocessing

Big space between two lines or two characters would influence the texture of the script, and it can not represent the characteristic of the script. So the original recon-structed image should be preprocessed and the unwanted space would be deleted.

4.3 Feature Extraction

Either multi-channel Gabor filtering or the gray level co-occurrence technique can be employed here, but the former one has better performance [9][10]. The multi-channel Gabor filter is used to get the texture information of the image here:

$$\begin{aligned}
h_e(x, y) &= g(x, y) \cdot \cos[2\pi f(x\cos\theta + y\sin\theta)] \\
h_o(x, y) &= g(x, y) \cdot \sin[2\pi f(x\cos\theta + y\sin\theta)]
\end{aligned} \tag{1}$$

Where

$$g(x, y) = \frac{1}{2\pi\sigma^2} \cdot \exp\left[-\frac{x^2 + y^2}{2\sigma^2}\right] \tag{2}$$

f, θ and σ are spatial frequency, orientation and space constant of the Gabor envelope. Here, a total of 24 Gabor channels for the Gabor filter are employed, and 48-dimension feature vectors are obtained. The parameters (f, θ, σ) to set will be de-scribed in Section 6.

4.4 Identification

Weighted Euclidean distance (WED) classifier is adopted to identify the writer [9]. Feature vectors of testing data are compared with those of training data.

$$WED(k) = \sum_{i=1}^{N} \frac{(f_i - f_i^{(k)})^2}{(\delta_i^{(k)})^2} \tag{3}$$

Where f_i denotes the i th feature of the testing data, $f_i^{(k)}$ and $\delta_i^{(k)}$ denote the mean value and the Variance of the training data from writer i. N denotes the total number of features extracted from a single writer.

5 Classifier Combination

Many fusion methods are developed in order to produce accurate recognition results. These methods are for different levels such as: abstract level (class label), ranked level (rank order) and measurement level (class scores) [5]. They can give comple-

mentary information on the data rather than only using single classifier for recognition. Three methods in measurement level are adopted here: Sum-Rule, Weighted Sum-Rule with common weights and Sum-Rule using user-specific weights. The matching scores from dynamic features and textures features are combined.

5.1 Normalization

Methods for measurement level are widely applied. In principle, the class scores provide richer information than the class label and the rank order and should give higher combination performance [5]. In fusion system, the measurements given by various classifiers distribute in different range. So the classifier's output should be shifted to a moderate range, so that the outputs of different classifiers are comparable. Here, linear normalization method is adopted. The maximum of output is defined as 1, and minimum is defined as 0. The others are linearly shifted to the range of [0,1].

5.2 Fusion Strategy

T1. Sum-Rule
Sum-Rule is a simple but effective method for classifiers combination. If the output of each classifier is $\{s_{i1}, s_{i2}, s_{i3} \cdots s_{i3}\}$, $(i \in N)$, Sum-Rule is defined as follows:

$$\arg\min{}_{j \in \{1,2,\cdots,c\}} \sum_{i=1}^{N} s_{ij} \tag{4}$$

Here N is the number of classifiers to combine, and C is the number of the classes in training samples.

T2. Weighted Sum-Rule Using Common Weights
This method is also called the Linear Conference Accumulation (LCA) method proposed by Huang [18]. The discriminant function for combination is defined as follows:

$$\arg\min{}_{j \in \{1,2,\cdots,c\}} \sum_{i=1}^{N} W_i * s_{ij} \tag{5}$$

Where s_{ij} is as same as it in T1, N is the number of classifiers combined, W_i is an indicator contribution coefficient defined as the confidence of each classifier. The confidence of each classifier is computed by the training result as $W_i = \dfrac{FIR_i^{-1}}{\sum_{j=1}^{N} FIR_j^{-1}}$ (FIR is the False Identification Rate).

T3. User-Specific Weighted Sum-Rule
This method is proposed by Jain [21]. "User-specific weight" means that the weights for each classifier are selected specifically for each user. Here, we improve the method by using an error-score. With the improvement, the method could be fit for the identification mode and requires a much smaller training set than before. User-

specific weights for each classifier are estimated by 'exhaustive' search from the training data as follows:

1. For the ith user in the database, vary weights $W_{1,i}$, $W_{2,i}$ over the range [0,1], with the constraint $W_{1,i} + W_{2,i} = 1$.
2. We define an error-score for each pair of weights: $m_{i,j}$ is defined as the matching score for the data of the ith user and the template of the jth user. Then we set $m_{i,j}$ as a sequence with ascending order. If the position of $m_{i,i}$ in the sequence is P, the error-score of this pair of weights is (P-1).
3. Choose the set of weights that minimizes the error-score for each user.

Then the combination method is similar to T2.

5.3   Combination

Two classifiers are required to combine, so $N = 2$ is set here. As 55 persons are involved in the NLPR database, C is evaluated to 55. Then strategy T1, T2 and T3 are employed. For each testing sample, the minimum of the sum measurement is the goal.

6   Experimental Results

Experiments are carried out on the NLPR handwriting database.

When we carry out the experiments with the method using dynamic features, two pages samples of each person are taken out for training and the other one (approximately 160 characters) is used for testing. The classifier here is called Classifier 1.

Then the static features are used for writer identification in the experiment. We get five sub-images from the reconstruction script image with the same size, two from each training page for training and one from testing page for testing. Since Gabor filters we used are of central symmetry in the frequency domain, only half of the frequency plane is needed. We choose $0°, 45°, 90°, 135°$ as the values of orientation θ. The values of central frequencies f are selected as 8,16,32,64,128,256, which are related with the size of the image[9]. The spatial constants σ are chosen to be inversely proportional to f. The classifier here is called Classifier 2.

In the experiments on fusion, we apply method T1, T2 and T3 separately. For each testing sample, outputs from Classifier 1 and Classifier 2 are combined with T1, T2 and T3. Then the minimum one of the results can indicate the identification result. For T2, The indicator contribution coefficient W_1 and W_2 can be got from the FIR of the two separate classifiers. So we obtained $W_1 = 0.7$ and $W_2 = 0.3$ from our training set. For T3, the specific weights for each user are selected from training set.

In our experiment, we use the samples with different numbers of the total characters for testing. We select samples with 4 lines, 5 lines, 6 lines, 7 lines and 8 lines of characters for each time, and get the identification results by using dynamic features, static features, and fusion. The results are shown in Fig. 2.

From Fig.2, we can see that the results of combination are obviously better than the other two. It proved our idea that combining the dynamic features and the static features could improve the result of identification. And further more, from the results, we can see that when we use samples with 5 lines (80 characters on average) of characters with T1, T2 or T3, it can achieve the accuracy of using samples with 7 or 8 lines

(more than 120 characters on average) in experiments adopting dynamic features only. The performance of T3 is especially satisfied. It shows that the combination method not only can improve the identification accuracy, but also reduces the amount of characters required in testing.
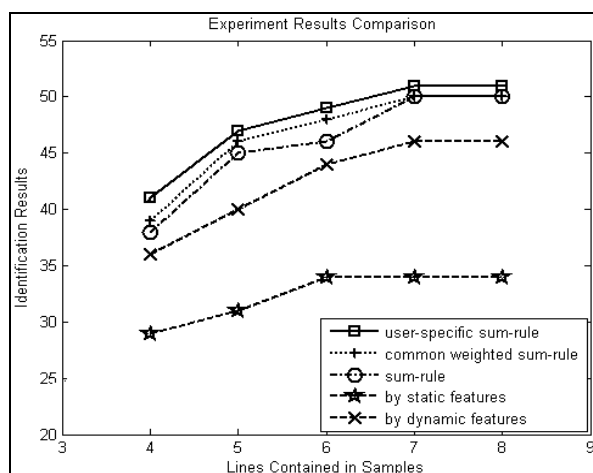

Fig. 1. The identification results for using dynamic feature only, texture feature only, and combining the two classifiers by sum-rule, common weighted sum-rule, and user-specific weighted sum-rule

7   Conclusion

In this paper, we have discussed text-independent writer identification. The previous methods adopting single scale features are unilateral. To form a comprehensive sight for this problem, we have combined the texture feature and the dynamic features in our paper. Compared to methods only employing dynamic features or texture feature, the accuracy has been improved with our method, and especially the number of characters required has been reduced to an acceptable accuracy. The fewer characters the method requires, the more applications it has in practice. So it is significant to bring the fusion method into text-independent writer identification.

We adopted three methods for classifier combination. Especially, we gave an improvement for the User-specific Sum-Rule algorithm by using an error-score. Other fusion methods may be used as well. But a complex one would require much more training data than we used, and reduce the efficiency of identification.

Future work would involve enlarging the NLPR handwriting database, and developing models for dynamic and static features to enhance the performance.

Acknowledgements

References

1. Anoop M. Namboodiri and Anil K. Jain: Online Handwritten Script Recognition. IEEE Trans. on PAMI, vol. 26, no. 3 (2004)
2. Claus Bahlmann and Hans Burkhardt: The Writer Independent Online Handwriting Recognition System frog on hand and Cluster Generative Statistical Dynamic Time Warping. IEEE Trans. on PAMI, vol. 26, no. 3 (2004)
3. Cheng-Lin Liu, Stefan Jaeger and Masaki Nakagawa,: Online Recognition of Chinese Characters: The State-of-the-Art. IEEE Trans. on PAMI, VOL. 26, NO. 2, (2004)
4. Mario E. Munich and Pietro Perona: Visual Identification by Signature Tracking. IEEE Trans. on PAMI, vol. 25 (2003).
5. Chenglin Liu: Classifier combination based on confidence transformation. Pattern Recognition, VOL.38 (2005) 11-28
6. Graham Leedham and Sumit Chachra: Writer Identification Using Innovative Binarised Features of Handwritten Numerals. ICDAR (2003)
7. Alessandro Vinciarelli, Samy Bengio and Horst Bunke: Offline Recognition of Unconstrained Handwritten Texts Using HMMs and Statistical Language Models. IEEE Trans. on PAMI, vol. 26 (2004)
8. Kun Yu, Yunhong Wang and Tieniu Tan: Writer Identification Using Dynamic Features. ICBA (2004)
9. Yong Zhu, Yunhong Wang and Tieniu Tan: Biometric Personal Identification Based on Handwriting. Proc. 15th International Conference on Pattern Recognition, vol.2 (2000) 801-804.
10. H. E. S. Said, T. N. Tan and K. D. Baker: Personal Identification Based on Handwriting. Pattern Recognition, vol.33 (2000) 149-160
11. K. Tsirikolias, B.G. Mertzios: Statistical Pattern Recognition Using Efficient Two-dimensional Moments with Applications to Character Recognition. *Pattern Recognition*, 26 (1993) 877-882
12. 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, (2002)
13. U. V. Marti, R. Messerli and H. Bunke: Writer Identification Using Text Line Based Features. ICDAR (2001)
14. L. R. B. Schomaker and R. Plamondon: The Relation between Pen Force and Pen-Point Kinematics in Handwriting. Biological Cybernetics *63* (1990) 277-289
15. 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, (2003)
16. Ching Y. Suen and Louisa Lam: Multiple Classifier Combination Methodologies for Different Output Levels. MCS 2000, LNCS 1857 (2000) 52-66
17. Wenwei Wang, Anja Brakensiek and Gerhard Rigoll: Combination of Multiple Classifier for Handwritten Word Recognition. IWFHR (2002)
18. Y. Huang and C. Suen: A Method of Combining Multiple Experts for the Recognition of Unconstrained Handwritten Numerals. IEEE Trans. on PAMI, VOL17 (1995) 90–94
19. Lambert Schomaker and Marius Bulacu: Automatic Writer Identification Using Connected-Component Contours and Edge-Based Features of Uppercase Western Script. IEEE Trans. on PAMI, vol. 26 (2004)
20. T.N. Tan: Texture Feature Extraction via Cortical Channel Modeling. Prov. 11th ICPR, vol.III (1992) 607-610
21. Anil K. Jain and Arun Ross: Learning User-Specific Parameters in A Multibiometric System. ICIP (2002)

Combining Wavelet Velocity Moments
and Reflective Symmetry for Gait Recognition

Guoying Zhao[1,2], Li Cui[3], and Hua Li[1]

[1] Key Laboratory of Intelligent Information Processing, Institute of Computing Technology,
Chinese Academy of Sciences, Beijing 100080
`lihua@ict.ac.cn`
[2] Machine Vision Group,
Infotech Oulu and Department of Electrical and Information Engineering,
P.O. Box 4500 FI-90014 University of Oulu, Finland
`gyzhao@ee.oulu.fi`
[3] School of Mathematics Science, Beijing Normal University, Beijing 100875
`licui@bnu.edu.cn`

Abstract. Gait is a biometric feature and gait recognition has become a challenging problem in computer vision. New wavelet velocity moments have been developed to describe and recognize gait. Wavelet moments are translation, scale and rotation invariant. Wavelet analysis has the trait of multi-resolution analysis, which strengthens the analysis ability to image subtle feature. According with the psychological studies, reflective symmetry features are introduced to help recognition. Combination of wavelet velocity moments and reflective symmetry not only has the characteristic of wavelet moments, but also reflects the person's walking habit of symmetry. Experiments on two databases show the proposed combined features of wavelet velocity moments and reflective symmetry are efficient to describe gait.

1 Introduction

Gait recognition is used to signify the identification of individuals in image sequences 'by the way they walk' [7]. From a surveillance perspective, gait recognition is an attractive modality because it may be performed at a distance, surreptitiously. Most other modalities require proximal sensing [8], making it difficult to apply unobserved and to many people. So gait, as a biometric, has some advantages over other biometrics.

Now, there are many methods to contribute to the gait analysis, for example model-based [3,12,13], appearance-based [14,15,16,17]. In the image sequences, translation and scale of walking people often exist, and moments are an efficient tool to deal with them. The application of classical moments to two dimensional images was first shown in the early sixties by Hu[1]. Little used moment based features to characterize optical flow for automatic gait recognition[2], thus linking adjacent images but not the complete sequence. Lee and Grimson computed a set of images features that are based on moments such as centroid, major axis of each part of people [3]. Liu et al. used the first and second moments of two binary silhouettes to determine an affine transformation that coarsely aligns them[4]. Shutler et al. developed new Zernike velocity moments to describe the motion throughout an image sequence to help recognize gait [5].

In this paper, a new method based on wavelet velocity moments and reflective symmetry is proposed, which aims to describe appearance, motion and walking habit through a time varying sequence, and produce multi-level features to recognize gait. This technique combines the feature of motion and appearance together, provide multi-level features to satisfy different requirement, at the same time keep to psychological studies.

2   Orthogonal Complex Wavelet Moments

Moments feature of image, in fact, are integration of space feature in whole image. To get rotation invariant moments, typically the following generalized expression is used[6]:

$$F_{p,q} = \int S_q(r) \cdot g_p(r) r dr .$$ (1)

where $S_q(r) = \int f(r,\theta) e^{-jq\theta} d\theta$. Note that $S_q(r)$ is now a 1D sequence of variable r. It is important to note from Eq. (1) that if $g_p(r)$ is defined on the whole domain of variable r, then $g_p(r)$ is a global feature, on the other hand, if the function of $g_p(r)$ is locally defined, then $F_{p,q}$ may be seen as a local feature.

Based on expression (1), we can easily show that the original definition for Zernike moments is the special case of expression (1) and the extracted features are global features[6]. So they only extract the global information of image, just are appropriate to the discrimination of two signals with apparent difference. And the low order moments describe the important feature and the details are reflected in the high-order moments, but computing high-order Zernike moments is very difficult. Thus it is not easy to correctly classify similar image objects with subtle differences based on such global and low order moment invariants and can not deal with noise.

Wavelet moments are new moment features which combine the wavelet characteristic and moment trait. Wavelet transform is a method for accomplishing localized analysis. But wavelet analysis is not invariant to translation and scaling. Some small translation can drastically change wavelet feature. Thus, simplex wavelet analysis is not widely used in the field of pattern recognition. Considering the respective characteristic of moment features and wavelet analysis, wavelet moments not only have the feature of moments, i.e., translation, scaling and rotation invariance, but also have the trait of wavelet analysis: multi-resolution analysis. It can strengthen the analysis ability to subtle image feature. To get the rotation-invariant moments, we exploit polar coordinates to express wavelet moments.

Complex wavelet moments are expressed as:

$$W_{m,n,q} = \iint f(r,\theta) [\varphi_{m,n}(r) \exp(jq\theta)]^* r dr d\theta .$$ (2)

Where * denotes the complex conjugate, $m = 0, 1, 2, 3 \cdots$; $n = 0, 1, 2, 3, \cdots 2^{m+1}$; $f(r,\theta)$ is the function described in polar coordinates. q is integer parameter, repre-

senting the qth frequency feature of the image object $f(r,\theta)$ in the phase domain. In this paper, $\varphi_{m,n}(r) = 2^{m/2}\varphi(2^m r - n)$, corresponding to the $g_p(r)$ in Eq.(1), is treated as wavelet basis functions, where $\varphi(r)$ is the mother wavelet function.

We consider using the cubic B-spline wavelets which are optimally localized in space-frequency and are close to the forms of Li's(or Zernike's) polynomial moments. Further more, they are orthogonal, so eliminate the high correlation of features. The mother wavelet $\varphi(r)$ of the cubic B-spline in Gaussian approximation form is[18]:

$$\varphi(r) = 4a^{d+1}\big/\sqrt{2\pi(d+1)} \cdot \sigma_w \cdot \cos(2\pi f_0(2r-1)) \cdot \exp(-(2r-1)^2\big/(2\sigma_w^2(d+1))) \qquad (3)$$

Where $a = 0.697066$, $f_0 = 0.409177$ and $\sigma_w^2 = 0.561145$, $d = 3$. Let $\varphi_{m,n}(r)$ sweep across in all angular rotations in the moment computation, it will be able to extract either global or local information depending on the values of m and n.

To calculate the cubic B-spline wavelet moments of an image $f(x,y)$, the region of walking people is mapped to the unit disc using polar coordinates, where the centroid of walking region is the origin of the unit disc. The coordinate are then described by polar coordinate.

Translation invariance is achieved by moving the polar origin to the centroid of walking people by computing the first Hu's moments. And scale invariance is implemented by scaling the walking region to the unit disc.

Because people are walking, velocity can be used as a feature to discriminate them. Velocity can be expressed as: $V(i,\mu,\nu) = (\overline{x_i} - \overline{x_{i-1}})^\mu(\overline{y_i} - \overline{y_{i-1}})^\nu$. Where $\overline{x_i}$, $\overline{y_i}$ are the centroid coordinate of frame i in Cartesian coordinates. The cubic B-spline Wavelet Velocity Moments (BWVMs) are expressed as:

$$W_{mnq\mu\nu} = \sum_{i=2}^{\text{Im}ages} \sum_r \sum_\theta V(i,\mu,\nu)\varphi_{mn}(r)\exp(-jq\theta)f_i(r,\theta)r_\vartriangle r_\vartriangle\theta . \qquad (4)$$

$r = \sqrt{x^2 + y^2}$ is limited in the unit circle $x^2 + y^2 \leq 1$. These results are averaged by normalizing with respect to the number of images and the average area of the object: $\|\overline{W}_{mnq\mu\nu}\| = \|W_{mnq\mu\nu}\|\big/(\overline{Area} \cdot \text{I})$, where $\overline{Area}$ is the average area(in numbers of pixels) of the moving object, I is the number of images and $\|W_{mnq\mu\nu}\|$ is the norm of wavelet moments, $\|\overline{W}_{mnq\mu\nu}\|$ is the norm of cubic B-spline wavelet velocity moments.

Furthermore, using different scale index m and shift index n, we ensure that the wavelets $\{\varphi_{m,n}(r)\}$ cover the whole radial domain $\{0 \leq r \leq 1\}$ and thus the wavelet moment invariants, $\|\overline{W}_{mnq\mu\nu}\|$, can provide features of the object $f(r,\theta)$ at different scale levels. We can implement the from-coarse-to-fine disposal, and gradually extract features until enough resolution satisfying the recognition requirement.

3   Reflective Symmetry Features

On the basis of the psychological studies, which suggest that gait is a symmetrical pattern of motion [9], we in this paper use the Reflective Symmetry (RS) feature of gait to help recognize gait sequence. Reflective symmetry can not only describe the appearance of people, but also implicitly represent the habit with which person's leg and body swings.

Reflective symmetry value of key frames according to the vertical line through centroid is applied as gait features. It's simple to implement, easy to incorporate with other features, and insensitive to the changes of walking speed, background, clothing that person wears.

We observe that height of centroid change periodically[10]. Therefore we use it to compute the period of gait, then get two kinds of key frames [10]: K_0 and K_1, which is similar to that of [10,11]. K_0 corresponds to the condition that two legs are furthest apart. K_1 represents the condition that two legs are together. We do not discriminate right and left legs because foot dominance has little influence on recognition[10].
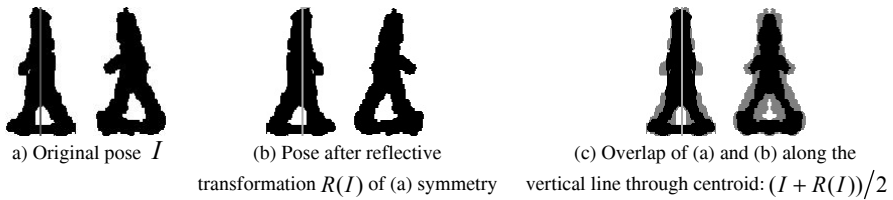

a) Original pose I	(b) Pose after reflective	(c) Overlap of (a) and (b) along the
	transformation $R(I)$ of (a) symmetry	vertical line through centroid: $(I + R(I))/2$

Fig. 1. Reflective symmetry of two persons in key frame K_0

We assume that principal axes of people exist and are vertical in image coordinate systems, such as the vertical line demonstrated in Fig.1(a). We can see that for different persons' gaits, their reflective symmetries of key frames are different too. Reflective symmetry implicitly includes the swing of arms and the inclination of body in walking and standing, so it's a useful feature in gait recognition. For example, the right person in Fig.1 inclines more than left person, the reflective symmetry(Fig.1(c) right) is not as good as that of the left person(Fig.1(c)left), symmetry value is 0.197 which is bigger than left person 0.103.

To get the active area of person and normalize the symmetry values, we compute the minimal distance width distance of centroid to left edge of bounding box and the distance to right edge to compute the active area of key frame: $A = \min(d(CenX, 0), d(CenX, W)) \times H$. Where $CenX$ is the width of centroid, H and W are height and width of bounding box respectively, $d(x, y)$ is the distance between x and y. Assume the original key frame is I, and image following the reflective symmetry is $R(I)$, then the symmetry value of this key frame is

$$SD(I, R(I)) = \|I - (I + R(I))/2\| = \|(I - R(I))/2\| . \tag{5}$$

Where $\|\cdot\|$ represents the sum of the symmetry of all pixel in I and $R(I)$. Because the pixel value in I and $R(I)$ is 0 or 1, if the pixel value in same location in I and $R(I)$ is also 0 or 1, value of $I_{i,j} - R(I)_{i,j}$ is 0, if they are different, $I_{i,j} - R(I)_{i,j}$ is 1.

In fact, it is the XOR(Exclusive OR) operation in logic. We only compute SD in the active area A. Then the reflective symmetry value is normalized: $SymD = SD/A$. Reflective symmetry feature of gait is obtained by averaging all two kinds of key frames to eliminate the influence of noise. The symmetry feature is: $(SymD(K_0), SymD(K_1))'$.

For in usual case, human are standing vertically in image sequences, we get the RS alone the vertical line through the centroid. If there are some rotations for walking people in image, we can compute the principal axis of walking people by Hu's moments, then get the RS along the principal axis. In this way, rotation variance can be eliminated. Computing the reflective symmetry along the principal axis makes us eliminate the variance of translation and rotation, normalizing SD by active area removes the effect of scale. So reflective symmetry features are not susceptive to translation, rotation and scale, too.

4   Experiments and Results

To evaluate our method, we use CMU Motion of Body (MoBo) database[11] and USF database[19] which are widely applied in gait recognition at present.

For each sequence, we use one period of gait to extract cubic B-spline wavelet velocity moments and reflective symmetry features. When computing the cubic B-spline wavelet velocity moments, we get the bounding box of silhouette firstly, then use the maximal length of centroid to four corner points of box as unity, compute the wavelet velocity moments in the unit disc with the centroid as origin. In our experiments, q is assigned to be equal to n in formula (2) to reduce the selection of parameters. To combine the BWVMs and RS features, we use formula (6) to get the similarity.

$$S(i,j) = \sum_{k=1}^{count} (BWVM_i(k) - BWVM_j(k))^2 / count + \sum_{t=0}^{1} (SymD_i(K_t) - SymD_j(K_t))^2 \qquad (6)$$

.

i, j are test sequence and train sequence respectively, $count$ is the number of cubic B-spline wavelet moments, and $BWVM_i(k)$ is the k th feature of cubic BWVMs of sequence i, which is normalized by divided by the maximal value of this feature to make it be in the range of $[0,1]$. $SymD_i(K_t)$ is the reflective symmetry of key frame K_t of sequence i.

4.1   Recognition Results in CMU Mobo Database

CMU MoBo database includes 25 subjects who walk in treadmill positioned in the middle of the room[11]. In computing performance scores, the silhouettes that were provided with the data set are applied. The original image size is 486 by 640. To reduce the computation time, we resize the image to 180 by 240.

Table 1 lists the reported identification rates for six algorithms on commonly reported experiments. The last row lists the performance of our algorithm. Our method

achieved the highest score. In the slow-fast experiment, we train in slow walk and test in fast walk, so robust for walking speed variation is evaluated. Results show our algorithm can deal with the speed change in gait efficiently.

Table 1. Top rank identification rates (percentages) for CMU Mobo data set reported by different algorithms. (The number of subjects in the gallery and probes are in parentheses.)

Train-test	Slow(25)-slow(25)	Slow(25)-fast(25)
CMU[11]	100	76
UMD[23,24]	72	32
UMD[7]	72	12
Georgia Tech.[25]	----------------	45
MIT[22]	100	64
Baseline [26]	92	72
Our algorithm	100	81

4.2   Evaluation in USF Dataset

For the experimental evaluation, we also test the present methodology on USF's Gait Challenge database described by Phillips et al.[19], which are captured under different conditions and is the largest available to date in terms of number of people, number of video sequences, and the variety of conditions under which a person's gait is collected. As shown in Fig.2, subjects walk in elliptical paths on two surface types(**C**oncrete and **G**rass), from two camera viewpoints(**L**eft and **R**ight), and with two shoe types(**A** and **B**). The gallery(reference) set of gait sequences was used as the system database and probe(test) sets A-G were considered to contain sequences of unknown subjects who are to be recognized by comparison of their gait sequences to the sequences in the gallery set[19].

Fig. 2. Images from USF database

Considering the constraints of processing time and storage space(about 300GB of data related to 452 sequences from 74 subjects), we directly use the same silhouette data from the University of South Florida. These data are noisy, e.g., missing of body parts, small holes inside the objects, and severe shadow around feet. We do not make any preprocessing and we use these noisy silhouettes directly to extract features and recognize them, at the same time to evaluate the anti-noise performance.

From Table 2, the same conclusions as [19,20] can be drawn. Among the three variations, the viewpoint seems to have the least impact and the surface type has the most impact[19,20]. Our algorithm(last columns in Table 2) have better performance in view and shoe variations(A,B,C,G), but sensitive to surface change. It outperforms present algorithms in viewpoint and shoe variations. Noisy segmentation results bring

about great impact on feature training and extraction, but it is less clear how much the silhouette type affects the recognition performance. As a whole, our method is comparable to the existing approaches in recognition accuracy, but far superior in observed execution times, the average is 1.6475min/seq, it outperforms [17][19][20][21][22]. Because data of gait database is huge, so in process of gait sequences, this is an advantage. As stated above, these are just basic comparative results. More detailed statistical studies on larger and more realistic data sets are desirable to further quantify the algorithm performance.

Table 2. Results(percentages) on USF database using gallery set(G,A,R). (The number of subjects in each subset is in square bracket and rate in parentheses is the results of Rank=5)

Probe	Difference	Ref[20]	USF[19]	Zernike Moments	Our methods
A(G,A,L)[71]	View	70.42(92.95)	79 (96)	77.94(92.65)	77.94(94.12)
B(G,B,R)[41]	Shoe	58.54 (82.92)	66 (81)	56.1(80.49)	63.41(85.37)
C(G,B,L)[41]	Shoe,View	51.22 (70.73)	56 (76)	34.15(65.85)	56.1(75.61)
D(C,A,R)[70]	Surface	34.33 (64.18)	29 (61)	9.38(32.81)	15.63(42.19)
E(C,B,R)[44]	Surface,Shoe	21.43 (45.24)	24 (55)	10.26(28.21)	15.38(35.9)
F(C,A,L)[70]	Surface,View	27.27 (38.57)	30 (46)	4.69(26.56)	7.81(29.69)
G(C,B,L)[44]	Surface, Shoe,View	14.29 (26.19)	10 (33)	12.82(28.21)	10.26(30.77)

5   Conclusions

In this paper, we combine cubic B-spline wavelet velocity moments and reflective symmetry to describe and recognize gait. Cubic B-spline wavelet velocity moments consider not only the static appearance feature, but also the motion velocity features. They have the trait of multi-resolution analysis, which provides the feature from coarsely to finely. Reflective symmetry can not only describe the appearance of people, but also implicitly represent the habit with which person's leg and body swings. The combination of the wavelet velocity moments and reflective symmetry makes best use of wavelet moment invariance, at the same time keeps to the psychological studies. In our paper, recognition rates are compared with other algorithms. Experiments on two gait databases show that our method is efficient for gait description and recognition.

References

1. Hu M.: Visual Pattern Recognition by Moment Ivariants. IRE Trans. On Information Theory, IT-8(1962) 179-187
2. Little J.J. and Boyd J.E.: Recognising people by their gait: The shape of motion. Videre, vol. 1, no.2(1998) 2-32
3. Lee L. and Grimson W.E.L.: Gait Appearance for Recognition. Biometric Authentication(2002) 143-154
4. Liu Y., Collins R. and Tsin Y.: Gait Sequence Analysis using Frieze Patterns., European Conference on Computer Vision, Copenhagen(2002) 657-671
5. Shutler J.D. and Nixon Mark S.: Zernike Velocity Moments for Description and Recognition of Moving Shapes. BMVC(2001)705-714
6. Shen D., Horace H.S.Ip: Discriminative wavelet shape descriptors for recognition of 2-D patterns. Pattern Recognition 32(1999) 151-165

7. BenAbdelkader C., Cutler R., and Davis L.: Motion-Based Recognition of People in Ei-genGait Space. International Conference on Automatic Face and Gesture Recogni-tion(FGR2002) 254-259

8. Bobick, A.F., Johnson A.Y.: Gait Recognition Using Static, Activity-Specific Parameters. IEEE Conference on Computer Vision and Pattern Recognition(2001)423-430

9. Cutting J.T., Proffitt D.R. and Kozlowski L.T.: A biomechanical invariant for gait percep-tion. J.Exp. Psych.:Human Perception and Performance(1978) 357-372

10. Zhao G., Chen R., Liu G., Li H.: Amplitude Spectrum-based Gait Recognition. 6th Interna-tional Conference on Automatic Face and Gesture Recognition, Seoul, Korea(2004) 23-28

11. Collins R.T., Gross R. and Shi J.: Silhouette-based Human Identification from Body Shape and Gait. FGR(2002)366-371

12. Dockstader S. L., Bergkessel K. A., and Tekalp A. Murat: Feature Extraction for the Analysis of Gait and Human Motion. ICPRQuebec City, QC, Canada(2002)5-8

13. Dockstader S. L. and Tekalp A. M.: A Kinematic Model for Human Motion and Gait Analysis. In Proc. of the Workshop on Statistical Methods in Video Processing (ECCV) Copenhagen, Denmark(2002)49-54

14. Wang L., Ning H., Hu W., Tan T.: Gait Recognition Based on Procrustes Shape Analysis. ICIP, Rochester, New York(2002)433-436

15. Huang P. S.: Automatic Gait Recognition via Statistical Approaches for Extended Tem-plate Features. IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, vol.31, no. 5(2001)818-824.

16. Kale A., Rajagopalan A.N., Cuntoor N., Kruger V., Chellappa R.: Identification of Humans Using Gait. IEEE Transactions on Image Processing, vol. 13, no. 9(2004) 1163- 1173

17. BenAbdelkader C., Cutler R., Nanda H. and Davis L.: EigenGait: Motion-based Recogni-tion of People using Image Self-Similarity. AVBPA(2001)284-294

18. Unser M., Aldroubi A., Eden M.: On the asymptotic convergence of B-spline wavelets to Gabor functions. IEEE Trans. Inform.Theory 38(1992)864-872

19. Phillips P.J., Sarkar S., Robledo I., Grother P., and Bowyer K. W.: The Gait Identification Challenge Problem: Data Sets and Baseline Algorithm. In ICPR(2002)385-388

20. Wang L., Tan T., Ning H. and Hu W.: Silhouette Analysis-based Gait Recognition for Human Identification. IEEE Trans. PAMI, vol. 25, no. 12(2003)

21. Collins R., Gross R., and Shi J.: Silhouette-Based Human Identification from Body Shape and Gait. FGR2002.

22. Lee L.and Grimson W.: Gait Analysis for Recognition and Classification. FGR(2002)155-162

23. Kale A., Cuntoor N., and Chellappa R.: A Framework for Activity Specific Human Identi-fication. Proc. Int'l Conf. Acoustics, Speech, and Signal Processing(2002)

24. Kale A., Rajagopalan A., Cuntoor N., and Kruger V.: Gait-Based Recognition of Humans Using Continuous HMMs. FGR(2002) 336-341

25. http://www.cc.gatech.edu/cpl/projects/hid/CMUexpt.html

26. Sarkar S., Phillips P. J., Liu Z., Vega I. R., Grother P., and Bowyer K. W.: The HumanID Gait Challenge Problem: Data Sets, Performance, and Analysis. PAMI, VOL. 27, NO. 2 (2005)162-177

Model-Based Approaches
for Predicting Gait Changes over Time

Galina V. Veres, Mark S. Nixon, and John N. Carter

Department of Electronics and Computer Science
University of Southampton, Southampton, SO17 1BJ

Abstract. Interest in automated biometrics continues to increase, but
has little consideration of time. This paper deals with a problem of recog-
nition by gait when time-dependent and time-invariant covariates are
added. We have shown previously how recognition rates fall significantly
for data captured over lengthy time intervals. We suggest predictive mod-
els of changes in gait due both to time and now to time-invariant covari-
ates. A considerable improvement in recognition capability is demon-
strated, with potential generic biometric application.

1   Introduction

Recently much attention has been devoted to use of human gait patterns as bio-
metric. Gait recognition aims to discriminate individuals by the way they walk.
Approaches to gait recognition can be broadly classified as being model-based
and model-free. Model-based methods [2, 3] model the human body structure.
Model-free methods [1, 8] generally characterise the whole motion pattern of the
human body. In this paper we employ the model-free (static) method of Veres et
al [6]. However, in these works only databases recorded over a short time interval
were evaluated.

In this paper we consider a gait recognition problem when two databases (the
gallery and probe) were recorded with a time interval of 6 and 12 months between
the finish of recording the first database (gallery) and the start of recording the
second database (probe), i.e. time-dependent covariates are added. Moreover,
some extra covariates were added in the second database such as different shoes,
clothes, carrying different bags. It is shown that correct classification rates fall
significantly and recognition becomes unreliable over time. Similar results have
been obtained for the HumanID Gait Challenge Problem [4], where recognition
collapsed from 82% to 6% for data acquiring at an interval of 6 months.

Under the assumptions that similar people have similar changes in gait,
several predictive models of gait changes are suggested in this paper. Three
databases were analysed in the paper. To estimate the effects of time on gait
recognition a new time-dependent predictive model is suggested. Time-dependent
predictive matrices are estimated for every subject in the training set. Then these
matrices are used to predict the possible changes in the gallery of the test set.
If a subject in the gallery of the test set does not belong to the gallery in the

training set, then the nearest neighbour from the training set is found. The predictive matrix corresponding to the nearest neighbour is used and the difference is included in prediction of the test gallery. Time-invariant covariates are now included in the time-invariant predictive model. Our earlier approach predicted the time-dependent covariate [7]. However we did not take into account time-invariant covariates. In this paper we try to predict changes in gait due both to time-variant covariates and to time-invariant covariates. In this case the time-invariant predictive matrices are calculated which represent dependency between normal walk of a subject and different possible extra covariates of the same subject. The combined predictive model now takes into consideration both time-dependent and time-invariant covariates of gait. The models suggested in the paper are linear, since it is the first approach to handling changes in gait over a lengthy time interval and it is a reasonable initial assumption. We show that CCRs can be increased by several times when using the new predictive models and in same cases more than 90% CCRs were achieved.

Section 2 describes the suggested predictive models for changes in gait over time and the new combined model. The database description is presented in Section 3. Experimental evaluation is presented and described in Section 4. Section 5 concludes this paper.

2   Prediction of Gait Changes

We assume that it is possible to predict the gallery over the given time interval and achieve good recognition results by analysing the probe via the predicted gallery. In this case the training set consists of a set of subjects from the gallery and the same set of subjects from the probe. The test set consists of different set of the same subjects, possible extra subjects and possible extra experiments from the gallery and the probe.

2.1   Prediction of Time-Dependent Covariates

In general case the predicted subject from the gallery can be defined as

$$\hat{\mathbf{g}}_{t+1}(i) = f(\mathbf{g}_t(i)), \tag{1}$$

where $\hat{\mathbf{g}}_{t+1}(i)$ is the ith predicted subject from a gallery, $\mathbf{g}_t(i)$ is the ith subject in the gallery, f is a prediction function for ith subject. We use a prediction function f in the form

$$\hat{\mathbf{g}}_{t+1}(i) = \mathbf{g}_t(i)\mathbf{W}_t(i), \tag{2}$$

where $\mathbf{W}_t(i)$ is $N_f \times N_f$ time-dependent predictive matrix for kth subject, N_f is a number of features for each subject.

Let the gallery and the probe be divided into groups (subjects). Let us consider at first the case when the number of groups in the gallery equals the number of groups in the probe and the groups are the same, i.e. training set. At first for each group in the probe and gallery the mean of the group is calculated

$$\bar{\mathbf{x}}_t^j(p) = \frac{1}{n_p^j} \sum_{i=1}^{n_p^j} \mathbf{x}_t^j(pi) \text{ and } \bar{\mathbf{x}}_t^j(g) = \frac{1}{n_g^j} \sum_{i=1}^{n_g^j} \mathbf{x}_t^j(gi), \tag{3}$$

where $\bar{\mathbf{x}}_t^j(p)$ and $\bar{\mathbf{x}}_t^j(g)$ are the means of group j in the probe and the gallery respectively, $j = 1, \ldots, n_g$, where n_g is a number of groups, n_p^j and n_g^j is a number of records in the jth group of the probe and of the gallery, respectively, $\mathbf{x}_t^j(pi)$ and $\mathbf{x}_t^j(gi)$ are records for ith subject in jth group in the probe and gallery respectively.

Then the time-dependent predictive matrix for each group j is calculated as

$$\mathbf{W}_t(j) = (\bar{\mathbf{x}}_t^j(g))^{-1} \bar{\mathbf{x}}_t^j(p), \tag{4}$$

where the inverse is calculated by pseudoinverse. Formula (4) allows to find changes in the jth subject's gait in two different time instances and can be derived from the equation

$$\bar{\mathbf{x}}_t^j(p) = \bar{\mathbf{x}}_t^j(g) \mathbf{W}_t(j), \tag{5}$$

where the gallery $\bar{\mathbf{x}}_t^j(g)$ was recorded in one time instance; the probe $\bar{\mathbf{x}}_t^j(p)$ in the next (future) time instance; the predictive matrix $\mathbf{W}_t(j)$ represents changes from the subject in the gallery to the subject in the probe due to time.

The ith predicted gallery subject from group j is

$$\hat{\mathbf{g}}_{t+1}(i) = (\mathbf{g}_t(i) + |\mathbf{g}_t(i) - \bar{\mathbf{x}}_t^j(g)|) \mathbf{W}_t(j), \tag{6}$$

where $|\mathbf{g}_t(i) - \bar{\mathbf{x}}_t^j(g)|$ describes deviation of the subject in the gallery from the mean of the group to which this subject belongs; $\mathbf{g}_t(i) + |\mathbf{g}_t(i) - \bar{\mathbf{x}}_t^j(g)|$ is a correction due to deviation from the group mean.

With obtaining $\mathbf{W}_t(j)$ for each group in the training set the training stage is finished and the time-dependent predictive matrices $\mathbf{W}_t(j)$ are saved for future use on the test set.

For the more general case (test set) when a number of groups in the gallery is not the same as number of groups in the probe and/or subjects are not the same both in the gallery and in the probe, the predictive matrix is constructed as follows. Here we present a case when a number of groups in the gallery is more than a number of groups in the probe. Two assumptions are made in this case

1. Every subject in the probe exists in the gallery.
2. The gait of the similar subjects will change in a similar manner with time.

We are looking forward to gathering more data to provide a theoretical analysis or statistical observation to support the second assumption.

If the same subject is presented both in the gallery of the training set and the gallery of the testing set, then group j which the subject from the test set belongs to is recorded and this subject is predicted. If the subject in the test set

$\mathbf{g}_t^{test}(i)$ is not in the training set, then the nearest neighbour from the training set is found by finding r such as

$$r = j : \{\min_j |\mathbf{g}_t(j) - \mathbf{g}_t^{test}(i)|, \; j = 1, \ldots, N_g\}, \tag{7}$$

where N_g is a number of records in the gallery of the training set. Then ith predicted subject is calculated as

$$\hat{\mathbf{g}}_{t+1}^{test}(i) = (\mathbf{g}_t^{test}(i) + |\mathbf{g}_t^{test}(i) - \bar{\mathbf{x}}_t^{j(r)}(g)|)\mathbf{W}_t(j(r)), \tag{8}$$

where $j(r)$ means that group j is chosen according to rth record of the training set.

After the predicted gallery is calculated as (6) or (8), the probe is classified via the predicted gallery. Prediction of time-invariant covariates is presented in the next subsection.

2.2   Prediction of Other Covariates

In this subsection we consider the dataset recorded over a short time interval when time will have little effect on recognition capability, and the effect of other covariates is much more likely. However, the subject cannot only walk normally but have some extra covariates like different clothes, shoes, wearing bags etc. In this case the gallery consists of subjects walking normally and the probe consists of records when some changes to normal walk are added. Then dependency between normal walk and any extra covariate can be expressed through the prediction of the gallery as

$$\hat{\mathbf{g}}_t^e(k) = f^e(\mathbf{g}_t(k)), \;\; \text{or} \;\; \hat{\mathbf{g}}_t^e(k) = \mathbf{g}_t(k)\mathbf{W}^e, \tag{9}$$

where f^e is a function of changes according to eth experiment, $\mathbf{W}^e$ is a square predictive matrix defining differences between normal walk and e th experiment.

The set of time-invariant predictive matrices $\mathbf{W}^e$ can be calculated using training set, and used when there is a possibility of extra covariates. In this case the training set is divided into groups according to experiments and the means of each experiment for each subject is calculated as

$$\bar{\mathbf{x}}_t^e(k) = \frac{1}{n^{ek}} \sum_{i=1}^{n^{ek}} \mathbf{x}_t^{ek}(i), \tag{10}$$

where is $\bar{\mathbf{x}}_t^e(k)$ is mean of eth experiment for kth subject and $\mathbf{x}_t^{ek}(i)$ is ith record in eth group for kth subject. Then the time-invariant predictive matrices $\mathbf{W}^e$ can be estimated as

$$\mathbf{W}^e(k) = (\bar{\mathbf{x}}_t^0(k))^{-1}\bar{\mathbf{x}}_t^e(k), \tag{11}$$

where $\bar{\mathbf{x}}_t^0(k)$ represents mean of normal walk for kth subject and $\bar{\mathbf{x}}_t^e(k)$ are means of all other experiments for kth walk, since the dependency between the normal walk and extra time-invariant covariates can be presented as

$$\bar{\mathbf{x}}_t^e(k) = \bar{\mathbf{x}}_t^0(k)\mathbf{W}^e(k). \tag{12}$$

Then the predicted gallery for k subject in the eth experiment will be

$$\hat{\mathbf{g}}_t^e(k) = \mathbf{g}_t(k)\mathbf{W}^e(k). \tag{13}$$

After the predictive matrices are obtained, the predicted gallery is calculated as (13) and the probe is classified via the predicted gallery. The combination of time-variant and non-time dependent covariates of gait is presented in the next subsection.

2.3   Combination of Different Gait Changes

So far we consider changes of gait either due to time or extra covariates separately. In this subsection we consider the case when both time and extra covariates are presented. Then the predicted gallery can be presented as

$$\hat{\mathbf{g}}_{t+1}^e(k) = f^e(f(\mathbf{g}_t(k))), \tag{14}$$

where f is function defining time-dependent changes and f^e defines changes due to eth experiment for kth subject. Combining formulas (6), (8) and (13) the combined predicted gallery is

$$\hat{\mathbf{g}}_{t+1}^e(k) = (\mathbf{g}_t(k) + |\mathbf{g}_t(k) - \bar{\mathbf{x}}_t^j(g)|)\mathbf{W}_t(j)\mathbf{W}^e(k), \tag{15}$$

if kth subject exists both in the training and test sets or

$$\hat{\mathbf{g}}_{t+1}^{test,e}(k) = (\mathbf{g}_t^{test}(i) + |\mathbf{g}_t^{test}(i) - \bar{\mathbf{x}}_t^{j(r)}(g)|) \times \mathbf{W}_t(j(r))\mathbf{W}^e(k), \tag{16}$$

if kth subject exists only in the test set.

Using formulas (15) and (16) we can predict the gallery taking into consideration both time-dependent and non time-dependent covariates.

3   Databases Description

Three databases were analysed in the paper, comprising indoor (studio) data. The first database, called the large database (LDB), consists of 115 subjects performing a normal walk. The database arrangements are described elsewhere [5]. The small database (SDB) consists of the sample of 10 subjects from the LDB. Each subject was filmed wearing a variety of footwear, clothes and carrying various bags. They were also filmed walking at different speeds. Each subject's data was captured during one continuous filming session. The temporal database (TDB) also consists of 10 subjects from the LDB in normal conditions, but each subject was re-filmed on different dates. One sequence for each subject consists of 50-60 frames describing gait. Examples of features from the LDB and the SDB are presented in Fig. 1. The figure shows subject 46's average silhouette computed for a sequence in the LDB and in the SDB walking normally, carrying a bag and wearing a trench coat. It worth noticing that sequence of filming
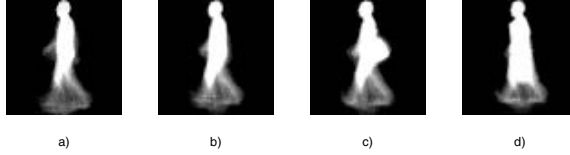
Fig. 1. Average silhouette representing subject 46: a) LDB, b) SDB, walking normally, c) SDB, carrying a bag, d) SDB, wearing trench coat

data in time was LDB, SDB and TDB with approximately 6 months difference between LDB and SDB, and SDB and TDB.

For brevity we shall not fully describe the extraction methods, complete explanations of the techniques used can be found in [6]. This technique yields a 4096 dimensional feature vector derived from the subject's silhouette accumulated over a sequence of images.

4   Experimental Evaluation

In this section we present the performance of the suggested predictive models in the tasks of recognizing people over lengthy time interval by gait, i.e when SDB was analysed via LDB S/L, TDB was analysed via SDB T/S and LDB T/L. All three data sets were divided between training set and testing set. The training set consists of 10 subjects belonging to all three databases walking normally for prediction of time-dependent covariates. However only 50% of records corresponding to a given subject in the database go to the training set, the remaining data belong to the testing set together with extra subjects for LDB and extra experiments (time-invariant covariates) for SDB. The system was tested on four different conditions. For the first we performed leave-one-out experiments using all training and test silhouettes walking normally, i.e. S_{10}/L_{10}, T/S_{10} and T/L_{10}. This experiment tested the accuracy of our approach in prediction of gait changes over time in subjects who had already provided training silhouettes. For the second experiment we added 105 new subjects to the gallery in the test set, i.e. S_{10}/L_{all} and T/L_{all}. In third experiment we investigate how adding extra time-invariant covariates will affect the time-dependent predictive model, i.e. S_{all}/L_{10} and T/S_{all}. The last experiment investigates the performance of the different predictive models when extra subjects and extra experiments are added, i.e. S_{all}/L_{all} and T/S_{all}.

The reduced datasets were used for representing all three databases, since it was shown [6] that recognition rates change little if a subset of the features is used. PCA was run on combination of three databases, and 500 features which represent 98.79% variation explaind were selected for further analysis. To show the influence of time and not a degradation of data quality on recognition, the CCR is calculated for each database separately both for original and reduce datasets. The results are presented in Table 1 and show acceptable CCRs for three datasets. Moreover, CCRs change little when a number of features reduces from 4096 to 500.

Table 1. Analysis of databases without time-dependent covariates

features	L_{all}/L_{all}	S_{all}/S_{all}	T/T
4096	98.47%	99.90%	99.30%
500	97.70%%	99.84%	99.62%

CCRs for analysis of different datasets before training are presented in Table 2. It can be seen that as soon as time-dependent covariates are added to the analysis the fall in CCR is very noticeable, especially when all subjects and all experiments are considered.

Table 2. CCRs for reduced feature sets before training

S_{10}/L_{10}	T/S_{10}	T/L_{10}	S_{10}/L_{all}	T/L_{all}	S_{all}/L_{10}	T/S_{all}	S_{all}/L_{all}
69.54%	73.88%	64.67%	44.20%	36.56%	50.88%	63.30%	24.12%

CCRs after applying the time-dependent predictive model on training and testing sets are presented in Table 3. It can be seen from Table 3, that the time-dependent predictive model copes very well when either only time-dependent covariates present or extra subjects are added from LDB both for training and testing sets. In this case more than 98% CCRs are achieved for training sets and more than 91% CCRs are achieved for testing sets. Moreover, adding extra covariates to SDB when it is considered as a gallery does not change the recognition capability. However when extra covariates are added to SDB treated as a probe, CCRs decrease significantly.

Table 3. CCRs after applying the time-dependent predictive model

set	S_{10}/L_{10}	T/S_{10}	T/L_{10}	S_{10}/L_{all}	T/L_{all}	S_{all}/L_{10}	T/S_{all}	S_{all}/L_{all}
training	99.53%	98.74%	98.59%	99.53%	98.59%	99.53%	98.74%	99.53%
testing	91.16%	91.32%	92.13%	91.16%	92.11%	26.46%	91.32%	26.46%

The results of using time-invariant predictive model for SDB is presented in Table 4. Table 4 shows that it is possible to calculate the predictive matrices at the training stage and then apply then successfully on the testing stage, thus achieving good recognition performance.

Table 4. Prediction of time-invariant covariates

dataset	training	testing
SDB	97.79%	96.57%

Table 5 shows the results of applying the combined predictive model. The results presented in Table 5 are significantly better than results reported in Table 2.

Table 5. CCRs using the combined predictive model

set	S_{all}/L_{10}	S_{all}/L_{all}
training	87.16%	87.16%
testing	85.76%%	72.03%

5   Summary and Conclusions

This paper deals with a problem of increasing correct classification rate when time-dependent covariates together with time-invariant covariates are added to an analysed database for gait recognition. We have shown that CCRs are very low in this case. In this paper we suggest use of the prediction of gait over the given time interval and prediction of possible changes in gait due to time-invariant covariates. The time-variant and time-invariant predictive models were developed. The experimental results showed that good results can be achieved both on the training set and the testing set on four different conditions.

References

1. R. Collins, R. Gross, and J. Shi "Silhouette-based human identification from body shape and gait" *Proceedings of the International conference on Automatic Face and Gesture Recognition*, Washington,DC, 2002
2. D. Cunado, M.S. Nixon, and J. N. Carter "Automatic extraction and description of human gait models for recognition purposes" *Computer Vision and Image Understanding*, **90**(1): 1-41, 2003
3. A. Kale, N. Cuntoor, and R. Chellapa "A framework for activity-specific human recognition" *Proceedings of IEEE International conference on Acoustics, Speech and Signal Processing*, Orlando, Fl,, 2002
4. S. Sarkar, P.J. Phillips, Z. Liu, I.R. Vega, P. Grother, and K.V. Bowyer "The HumanID Gait Challenge Problem: Data sets, Performances, and Analysis " *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**(2): 162-177, 2005.
5. J. D. Shutler, M. G. Grant, M. S. Nixon, and J. N. Carter "On a Large Sequence-Based Human Gait Database" *Proc. 4th International Conf. on Recent Advances in Soft Computing*, Nottingham (UK), 2002
6. G.V. Veres, L. Gordon, J.N. Carter and M.S. Nixon "What image information is important in silhouette-based gait recognition?" *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, D.C., USA, **II**: 776-782 2004.
7. G.V. Veres, M.S. Nixon and J.N. Carter "Modelling the time-variant covariates for gait recognition" *Proceedings of 5th International conference on Audio- and Video-Based Viometric Person Authentication*, New York, USA, 2005.
8. L. Wang , W.M. Hu and T.N. Tan "A new attempt to gait-based human identification" *IEEE Transactions on Circuits and Systems for Video Technology*, **14**(2): 149-158, 2002.

Using Ear Biometrics for Personal Recognition*

Li Yuan, Zhichun Mu, and Zhengguang Xu

School of Information Engineering, Univ. of Science and Technology Beijing. Beijing 100083
`yuanli64@hotmail.com`

Abstract. Application and research of ear recognition technology is a new subject in the field of biometrics recognition. Earlier research showed that human ear is one of the representative human biometrics with uniqueness and stability. Feasibility and characteristics of ear recognition was discussed and recent advances in 2D and 3D domain was presented. Furthermore, a proposal for future research topics was given, such as ear database generation, ear detection, ear occluding problem and multimodal biometrics with face etc.

1 Introduction

In machine vision, ear biometrics has received scant attention compared to the more popular techniques of automatic face, iris, or fingerprint recognition. However, ears have played a significant role in forensic science for many years, especially in the United States, where an ear classification system based on manual measurements has been developed by Iannarelli, and has been in use for more than 40 years [1]. The United States Immigration and Naturalization Service (INS) has a form giving specifications for the photograph that indicate that the right ear should be visible [INS Form M-378 (6-92)]. During crime scene investigation, earmarks are often used for identification in the absence of valid fingerprints [2]. The long history of use of ear shapes/marks suggests its use for automatic human identification.

This paper is organized as follows: section 2 presents the structure of the ear and the feasibility of ear biometrics. In section 3 we introduces categories of different methods of ear biometrics recognition. Then conclude the paper in section 4 with possible research interests of ear recognition in future study.

2 Ear Biometrics

The ear is made up of standard features including the helix, the antihelix, the lobe, and the u-shaped intertragic notch between the ear hole and the lobe. Figure 1 shows the locations of the anatomical features in detail.

Ear biometrics has the four properties for a potential biometrics: universality, uniqueness [1], permanence [1] and collectability. Among the popular biometrics, ear biometrics resembles face most. They all make use of facial features and face the same problems, such as illumination, head rotation and occlusion [3]. But it dose have several advantages over face. Ear has a rich and stable structure that is preserved from birth into old age [1], and dose not suffer from changes in facial expression [4,5]. Ear image

---

is smaller under the same resolution, which can be favorable in some situations, such as the audio-visual person authentication using speech and ear images for mobile phone usage [6]. Ear images can be acquired in a similar manner to face images, so it can also be used in surveillance scenarios and negative recognition.
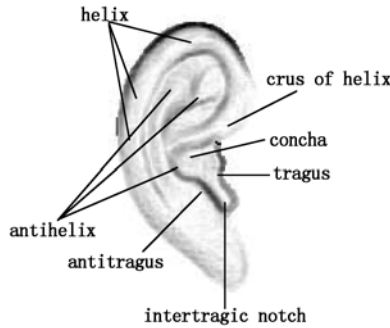

Fig. 1. Anatomy of the human ear

3   Ear Recognition Methods

An ear recognition system involves segmentation of ears (ear detection) from input still or video images of a scene, feature extraction from the ear regions, recognition/verification. Based on the input image, ear recognition can be categorized as follows: ear recognition from intensity images and ear recognition in 3D.

3.1   Ear Recognition from Intensity Images

The ear recognition problem is formulated as recognizing 3D objects from 2D images. The recognition methods can be categorized into feature-based methods (such as geometry feature based methods, neural network and force field transform) and holistic methods (such as PCA and moment invariants methods).

A) Feature Based Ear Recognition
Burge and Burger [4] use the main curve segments to form Voronoi diagram, and build adjacency graph out from Voronoi diagram. They use adjacency graph matching based algorithm for authentication which takes into account the erroneous curves caused by e.g. lightning, shadowing and occlusion. But the curve segments will be affected by even relatively small changes in camera-to-ear orientation or in lighting. So this method is unstable [4].

To extract geometry features that are invariant to translation, scaling and rotation, we proposed Long Axis based Shape and Structural Feature Extraction Method (LABSSFEM) [7]. After extracting the edge image of the ear, we apply least square curve approximation to simulate the outer ear contour. The coefficient vectors of Line1 and Line2 are combined to form the shape feature (figure 2(a)). The structural feature vector is formed as [OC/AB, OD/AB, OE/AB, OG/OF, OH/OF] (figure 2(b), AB is defined as the long axis). This method was tested on USTB ear database (figure 5), and we found that the combination feature of shape and structural feature got a better recognition performance than using the shape or structural feature alone.
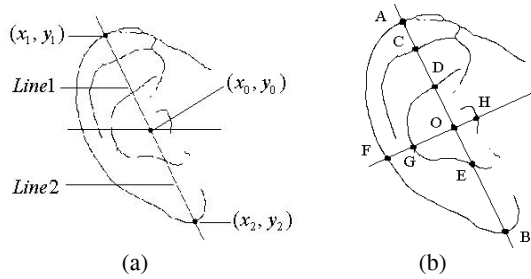
Fig. 2. Ear geometry feature extraction for (a) shape feature and (b) structural feature

Moreno et al. [5] investigated the performance of various neural classifiers and combination technique in ear recognition. The ear image is analyzed by several neural classifiers using outer ear feature points, information obtained from ear shape and wrinkles, and macro features extracted by compression network.

Instead of extracting a feature vector to describe the ear, Hurley et al. [8,9] proposed the force field transformations. The image is treated as an array of mutually attracting particles that act as the source of Gaussian force field. The original image is described by a set of potential channels and potential wells (figure 3). Force field feature extraction method dose not need the ear to be explicitly extracted from the background, and initial results were promising [9].
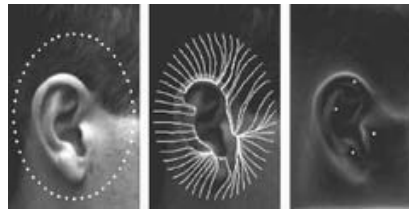

Fig. 3. Extraction of potential wells and channels (a) Exploratory mobile test pixels. (b) Field line and potential energy channels. (c) Potential wells

B) Holistic Approaches

Victor [10] made a comparison between face and ear recognition based on standard PCA. Factors such as time lighting, expression and time lapse between successive image acquisitions were taken into account. Results showed that in all experiments, face-based recognition gives better performance than ear-based recognition.

Later Chang [11] made another comparison with a larger database, and found that there is no significant difference between the face and the ear in terms of recognition performance. The quality of face and ear images in the dataset is more rigorously controlled in the experiments reported in [11]. The different conclusion reached by the two attempts maybe due to quality of their data. The image data sets in [10] had less control over the variations such as earrings, hair over ears, etc. [11].

Wang [12] extracted the high order moment invariants of ear image to form feature vector. The moment invariants are invariant to translation, scaling and rotation of the ear. The recognition performance based on the Neural Network is promising.

3.2   Human Ear Recognition in 3D

In [13], local surface patch comparison with range images is presented. A local surface descriptor is defined by a centroid, its surface type and 2D histogram. The 2D histogram consists of shape indexes, calculated from principal curvatures, and angles between the normal of reference point and that of its neighbors. By comparing local surface patches between a test and a model images, the potential corresponding local surface patches will be found. Geometric constraints are used to filter the corresponding pairs. Verification is performed by estimating transformation and aligning models with the test image.

Another approach for 3D ear recognition is ICP (Iterative Closest Point) algorithm. ICP algorithm is to match a data shape Y to a model shape X by iteratively minimizing the distance between corresponding pairs. Hui [14] proposed a two-step ICP approach. The first-step ICP is to align the model ear helix with the test ear helix which brings the model and test ear into coarse alignment and provide the initial rigid transformation for second-step ICP. The second-step ICP iteratively refines the transformation by minimizing the distance between the control points of the model ear and their closet points of the test ear, and get a root mean square (RMS) registration error. The RMS error is used as the matching criterion. The model ear with the minimum RMS error is declared as the recognized ear. Yan [15] also used ICP approach on a larger scale dataset. In order to speed up ICP algorithm, they control the number of iterations and use Octree and k-d tree data structure to shrink the running time for each iteration.

3.3   Performance Overview and Future Work

Table 1 gives an overview of the recognition performance of the approaches mentioned above.

Table 1. Summary of different ear recognition methods

Approach	Ear Image	Ear Database	Recognition Rate
LABSSFEM [7]	2D	77 (training), 77 (test), USTB ear database [16]	85%
Neural Networks [5]	2D	84(training), 28 (validation), 56 (test)	93%
Force Field Transformation [9]	2D	252 (test) XM2VTS face database [17]	99.2%
PCA [10,11]	2D	197 (training), 88 (registrant) ND Human ID database [18]	71.6%
Moment Invariants [12]	2D	120 (training), 60 (test) USTB ear database	96%
Local Surface Patch [13]	3D	10 (training), 10 (test)	100%
Two-step ICP [14]	3D	30 (training), 30 (test)	93.3%
Improved ICP [15]	3D	302 (training), 302 (test) ND Human ID database [18]	98.8%

From table 1, we can see that research in ear biometrics is just rudimental. The practical application of ear recognition may rely on the following future study.

1. Feature extraction: Most present methods [5,9,10,11,13,14] resemble face recognition methods. The difficulty of ear feature extraction lies in the changes among the same ear caused by head rotation and lighting variation as face recognition [19]. The geometry feature extraction depends heavily on the quality of the image preprocessing. For example, the curve segments extracted in [4] and the structural extracted in [7] will be different even for the same person under different lighting conditions, which makes the methods unreliable. The rotation discrimination is even more challenging because the angle between the ear and the head is not the same among different people. But ear and face do have differences. The gray intensity distribution and edge information should be used.

2. Ear Detection: Present approaches [4,5,9-14] are based on ear images stored in databases. They manually extract the ear from the input images. But in an automatic recognition system, the ear part should be detected from the input image or video, and got normalized in location and size. The location criteria could make use of the anatomy structure of the ear (figure 4). In [21], triangular fossa and antitragus are used for location landmark points (figure 4 (a)). But they are not easy to be marked in the image. So we use the long axis (figure 4 (b)) or the start point and the end point of the outer ear contour (figure 4(c)) as landmarks. The orientation of the line connecting the two landmark points or the long axis of the outer ear contour is used to determine the orientation of the ear, and the distance between the two points or the length of the long axis are used to measure the size of the ear. The orientation and the size should be the same after normalization.

We also applied the improved skin color model to detect ear in color image and template matching to detect ear in gray images [20]. Ellipse simulation was used for real-time ear detection and tracking. Through experiment, we found that recognition performance of some methods (such as PCA, LDA) will increase dramatically when the input image contains much less background information around the ear. So ear detection step is necessary.
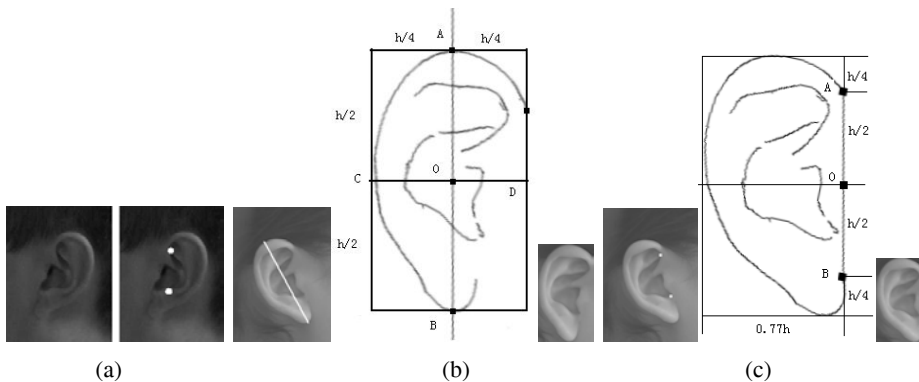

Fig. 4. Landmark Location: (a) Using Triangular Fossa and Antitragus [21]; (b) Using long axis of the outer contour; (c) Using start point and end point of outer ear contour

Table 2. Investigation about ear occluding (818 male subjects, 818 female subjects)

Gender	Occluding Reasons				
	Hat/Scarf/Collar	Hair		Earrings	Ear piece
		Partially	Totally		
Male	73	46	12	2	4
Percentage( )	8.9	5.6	1.5	0.2	0.5
Female	102	205	153	65	3
Percentage( )	12.5	25.1	18.7	7.9	0.4

3. Ear occluding problem: In ear recognition, one obvious problem we must face is the fact that the ears are often occluded by some objects such as hair or hat. Table 2 shows an investigation about this problem.

As we can see from table 2, excluding the seasonal cause (hat/scarf) and occasional cause (ear piece), the fixed occluding rate of female's ears is 56.3%. In the case of active identification systems this will not be a problem, as the subject can pull their hair back and proceed with the authentication process. While the problem arises during passive identification as in this case no assistance on the part of the subject can be assumed.

The possible solutions to the occluding problem are as follows. The thermogram images can be used for total occluding situation. One method is to reconstruct the occluded image with the thermogram image. The other one is to recognize the subject with the thermogram image as refers to [22]. As to the partial occluding images, Minimum Feature Set (MFS) can be constructed and used when limited information is available. The relationship between MFS and the recognition performance is worth of researching.

4. Generating ear database: The following conditions may be considered when taking images of each subject: illumination variation, orientation variation, distance variation, date variation and occluding variation. Now we have established an ear image set of 150 subjects (figure 5) [16].
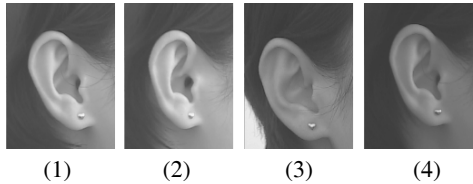

(1)   (2)   (3)   (4)

Fig. 5. USTB Ear Database ((1) is taken under normal lighting condition, (2) and (3) are taken -30°and +30°rotation respectively, (4) is taken under different lighting condition.)

5. Face and ear multimodal biometrics: In a multimodal biometric system, information fusion can occur in three levels: (a) data or feature level, (b) matching score level and (c) decision level [23]. As researchers have studied ear recognition in 2D and 3D, multi-biometrics 2D and 3D ear recognition is explored in [24]. It is found that multi-modal (PCA for 2D ear image and ICP algorithm for 3D ear image), multi-algorithm (PCA, ICP and edge-based algorithm for 3D ear data) or multi-instance (different numbers for gallery and probe images) improves performance over a single biometric. The fusion rules work at the matching score level in this study [24].

Ear is adjacent to face, and they are unlikely to be covered at the same time. So ear can be combined with face to form multimodal biometrics. Chang [10] applied PCA on combined image of ear and face, and found that multi-modal recognition results in statistically significant improvement over either individual biometric. Much work has been done on matching score level fusion and decision level fusion, but ear and face biometrics fusion are supposed to be more effective at the feature level, because ear feature and face feature are possible to be in the same measure scale. So the fusion rules at the feature level is one of our research interests in the future. On the other hand, the association relationship between ear and face will probably generate a new kind of feature with uniqueness, but the challenge lies in the measurement of the "association feature".

4   Conclusions

In this paper we presented an overview of machine recognition of human ears, and considered two types of ear recognition task: one from intensity images and the other from range images. Five issues of ear recognition are discussed for the purpose of practical application as a working system. Ear biometric is expected to be used in access control or video surveillance for "non-intrusive" recognition by itself or com-bined with face as multi-modal biometrics.

References

1. Afred Iannarelli: Ear Identification. Forensic Identification Series. Paramount Publishing Company, Fremont, California (1989)
2. Meijerman Lynn, Sholl Sarah: Exploratory study on classification and individualization of earprints. Forensic Science International. Vol. 140, No. 1 (2004) 91-99
3. K.H. Pun, Y.S. Moon: Recent Advances in Ear Biometrics. Proc. of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition. (2004) 164-169
4. Burge M. and Burger W.: Ear Biometrics in Computer Vision. In the 15th International Conference of Pattern Recognition. (2000) 822-826
5. Belé Moreno, Ángel Aánchez, José F.Vélez: Use Outer Ear Images for Personal Identification in Security Applications. IEEE Proc. of the 33rd Annual 1999 International Carnahan Conference, Madrid (1999) 469-476
6. Koji Iwano, Tomoharu Hirose, Eigo Kamibayashi: Audio-visual person authentication using speech and ear images. Proc. Workshop on Multimodal User Authentication, Santa Barbara, California (2003) 85-90
7. Mu Zhichun, Yuan Li, Xu Zhengguang, Xi Dechun, Qi Shuai: Shape and Structural Feature Based Ear Recognition. Advances in Biometric Person Authentication. The 5th Chinese Conference on Biometric Recognition, Guangzhou, China (2004) 663-670
8. Hurley, D.J., Nixon, M. S., Carter, J.N.: Force field energy functions for image feature extraction. Image and Vision Computing, Vol. 20 (2002) 311-317
9. David J. Hurley, Mark S. Nixon, John, N. Carter: Force field feature extraction for ear biometrics. Computer Vision and Image Understanding, Vol. 98 (2005) 491–512
10. Victor B., Bowyer K., Sarkar S.: An evaluation of face and ear biometrics. Proceedings of International Conference on Pattern Recognition, (2002) 429-432
11. Chang, K., Bowyer. K.W., Sarkar, S., Victor, B.: Comparison and Combination of Ear and Face Images in Appearance-Based Biometrics. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25 No. 9 (2003) 1160-1165

12. Wang Zhongli, Mu Zhichun: Moment Invariant Based Ear Recognition. Pattern Recognition and Artificial Intelligence (Chinese). Vol. 17 No. 4 (2004) 502-505
13. B. Bhanu, H. Chen: Human Ear Recognition in 3D. Workshop on Multimodal User Authentication, CA USA (2003) 91-98
14. H. Chen, B. Bhanu: Contour matching for 3D ear recognition. Proceedings of the Seventh IEEE Workshop on Applications of Computer Vision, (2005) 123-128.
15. Ping Yan, Bowyer K.: ICP-based Approaches for 3D Ear Recognition, SPIE Conference on Biometric Technology for Human Identification Orlando, FL. (2005)
16. http://www.ustb.edu.cn/resb/.
17. K. Messer, J. Matas, J. Kittler, J. Luettin, G. Maitre: XM2VTSDB: the extended M2VTS Database. Proc. Audio-Video Biometric Person Authentication, Washington DC (1999)
18. http://www.nd.edu/~cvrl/HID-data.html
19. Wen-Yi Zhao, Rama Chellappa, PJ Jonathon Phillips, Azriel Rosenfeld: Face Recognition: A Literature Survey. ACM Computing Survey, December Issue (2003) 399-458
20. Wang Min, Mu Zhichun: Skin Color Model Based Ear Detection. Control & Automation (Chinese). Vol.11 (2005)
21. Ping Yan, Bowyer K.: Ear Biometrics Using 2D and 3D Images. TR 2004-31. http://www.cse.nd.edu/research/tech_reports/2004.html.
22. Xin Chen, Patrick J. Flynn, Bowyer K.: PCA-based Face Recognition in Infrared Imagery: Baseline and Comparative Studies. International Workshop on Analysis and Modeling of Faces and Gestures, Nice, France (2003) 127-134
23. Arun Ross, Anil Jain: Multimodal Biometrics: An Overview. Proc. of 12th European Processing Conference. (2004) 1221-1224
24. Ping Yan, Kevin W. Bowyer. Multi-biometrics 2D and 3D ear recognition. Proc. Audio-and Video-Based Biometric Person Authentication, New York (2005)

Biometric Identification System Based on Dental Features

Young-Suk Shin

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

Abstract. In this paper, we present a new biometric identification system based on dental features. First, we collected the plaster figures of teeth which were done at dental hospital, department of oral medicine. Second, we developed a representation of dental images based on principal component analysis(PCA) representation included the 100 principle components as the features for individual identification. The PCA basis vectors had reflected well the features for individual identification from the parts of teeth. Finally, the classification for individual identification based on the nearest neighbor(NN) algorithm from the part of teeth was created. The identification performance is 98% for the part of teeth excluded the right-molar and back teeth, 95% for the part of teeth excluded the front teeth and 80% for the part of teeth excluded the left-molar and back-teeth.

1   Introduction

Biometric identification systems based on biometric features are expected to provide secure access in the near future to physical and virtual resources or spaces. Any physiological feature could be used for individual identification. The most generalized biometric techniques include the recognition of fingerprints, faces, iris, retina, hand geometry, voice and signature[1,2,3,4,5,6].

Dental features of human was an extremely valuable source of biometric information from a tragic death in a large-scale like a tsunami happened recently or airplane-accident. Therefore, individual identification based on dental features will be support indeed a very fast and reliable identification in a situation excluded from generalized biometric techniques. Jain and Chen[7,8] utilized dental radiographs for human identification. The dental radiographs have a number of challenges to overcome. For poor quality images where tooth contours are indiscernible, the shape extraction is a difficult problem for dental radiographs. Our algorithm utilizes information about differences in the size and shape of the jaws, size of the teeth and teeth structure.

In this paper, we present a new biometric identification system based on dental features like the size and shape of the jaws, size of the teeth and teeth structure. First, we collected the plaster figures of teeth from the department of oral medicine in dental hospital. Second, we developed a representation of dental images based on PCA included the 100 principle components as the features for individual identification. Finally, the nearest neighbor algorithm for the classification of individual identification was applied.

2   Dental Database

Dental data was a database of the plaster figures of teeth which were done at Chosun University dental hospital, department of oral medicine. This database consists of 300 person of males and females. Each person has two images in a upper jaw and lower jaw. The data set used for research contained 100 gray level images in a upper jaw, each image using 800 by 600 pixels. Examples of the original images are shown in figure 1.

Fig. 1. Examples in a upper jaw from the dental database

3   PCA Representations of Dental Features

The dental images were centered with fixed coordinates locations, and then cropped, dug the palatine and rescaled to 30x30 pixels. The luminance was normalized in two steps. First, a "sphering" step prior to principal component analysis is performed. The rows of the images were concatenated to produce 1×900 dimensional vectors. The row means are subtracted from the dataset, X. Then X is passed through the zero-phase whitening filter, V, which is the inverse square root of the covariance matrix:

$$V = E\{XX^T\}^{-\frac{1}{2}}$$
$$W = XV$$

(1)

This indicates that the mean is set to zero and the variances are equalized as unit variances. Secondly, we subtract the local mean gray-scale value from the sphered each patch. From this process, W removes much of the variability due to lightening.

Some of the most successful algorithms applied PCA representation are "eigen faces[9]" and "holons[10]". These methods are based on learning mechanisms that are sensitive to the correlations in the images. PCA provides a dimensionality-reduced code that separates the correlations in the input. Atick and Redlich[11] have argued for such compact, decorrelated representations as a general coding strategy for the visual system. Redundancy reduction has been discussed in relation to the visual system at several levels. A first-order redundancy is mean luminance. The variance, a second order statistic, is the luminance contrast. PCA is a way of encoding second order dependencies in the input by rotating the axes to corresponding to directions of maximum covariance.

For individual identification based on dental features like the size and shape of the jaws, size of the teeth and teeth structure, we employed the 100 PCA coefficients, P_n. The principal component representation of the set of images in W in Equation(1) based on P_n is defined as $Y_n = W * P_n$. The approximation of W is obtained as:

$$\overline{W} = Y_n * P_n^T .\tag{2}$$

The columns of $\overline{W}$ contains the representational codes for the training images. The representational code for the test images was found by $\overline{W}_{test} = Y_{test} * P_n^T$ (see figure 2). Best performance for individual identification based on dental features was obtained using 100 principal components.
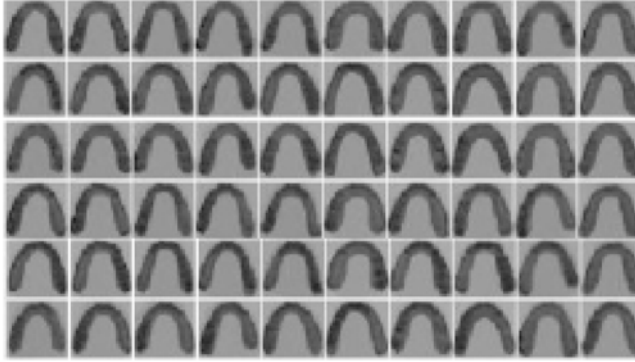

Fig. 2. PCA representation included the 100 principle components

4   Results and Conclusions

100 person images in a upper jaw to train was used. For testing, total 60 person images with missing teeth in the three parts from the training set were used. The three parts consist of right-molar and back-teeth, left-molar and back-teeth and front-teeth. Each part with missing teeth includes 20 person images. Classification for person identification was applied the nearest neighbor algorithm. The principle of the NN algorithm is that of comparing input image patterns against a number of paradigms and then classifying the input pattern according to the class of the paradigm that gives the closest match.

The recognition performance was 98% for the part of teeth excluded the right-molar and back-teeth, 95% for the part of teeth excluded the front-teeth and 80% for the part of teeth excluded the left-molar and back-teeth.

By PCA representation based on dental features like the size and shape of the jaws, size of the teeth and teeth structure, our system developed a new biometric identification system in a situation excluded from generalized biometric techniques like fingerprints, faces, iris, retina, hand geometry, voice and signature. This simulation demonstrates that PCA representation can solve a challenging problem like a correct object recognition from the part shape in object recognition.

Our system extracts PCA representation included only 100 principle components from 30x30 images and person identification was successfully produced. It can reflect the fact that the holistic analysis is important for person identification based on dental features like the size and shape of the jaws, size of the teeth and teeth structure. We suggest that in order to recognize the objects from the parts, the patterns within the

scenes have to be holistic information reduced redundancy. This may be analyzed shape primitives that take part in the composition of the objects more frequently than any other individual structure. In the future study we are planning to analyze individual identification difference from chewing patterns in much larger database than present system.

References

1. Jain, A.K. Pankanti, S. Prabhakar, A., Ross, A.: Recent advances in fingerprint verification, Lecture Notes Comput. Sci. 2091 (2001) 182-190
2. Phillips, P.J.: Support vector machines applied to face recognition, Technical Report, NISTIR 6241, National Institute of Standards and Technology, 1999
3. Daugman, J., Downing, C.: Epigenetic randomness, complexity, and singularity of human iris patterns, Proc. Roy. Soc. 268 (2001) 1737-1740
4. Jain, A. K., Ross, A., Pankanti, S.: A prototype hand geometry-based verification system, Second International Conf. on Audio and Video-based Biometric Person Authentification, Washington DC, USA, March (1999)
5. Furui, S.: Recent advances in speaker recognition, in:J. Bigun, G.Borgeford (Eds.), Audio and Video-based Biometric Person Authentification, Springer, Berlin, (1997)
6. Nalwa, V.S.: Automatic on-line signature verification, Proc. IEEE 85 (2), (1997) 215-239
7. Jain, A. K., Chen, H.: Matching of dental X-ray images for human identification, Pattern Recognition **37**(7), (2004) 1519-1532
8. Chen, H., Jain, A. K.: Tooth contour extraction for matching dental radiographs, proc. 17th ICPR(3) Cambridge, UK, August (2004) 522-525
9. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience **3**(1) (1991) 71-86
10. Cottrell, G., Metcalfe, J.: Face, gender and emotion recognition using holons. In Touretzky, D., editor, Advances in Neural information processing systems (3) San Maleo, CA. Morgan aufmann (1991) 564-571
11. Atic, J., Redlich, A.: What does the retina know about natural scenes?, Neural Computation (4) (1992) 196-210

A Secure Multimodal Biometric Verification Scheme*

Dongmei Sun[1], Qiang Li[1], Tong Liu[2], Bing He[3], and Zhengding Qiu[1]

[1] Institute of Information Science, Beijing Jiaotong University, Beijing 100044, P.R. China
dmsun@center.njtu.edu.cn
[2] Beijing Science and Technology Information Institute, Beijing 100044, P.R. China
[3] ECECS Department, University of Cincinnati, Cincinnati, OH 45221, USA

Abstract. A multimodal biometric scheme using watermarking technique to provide more secure and reliable personal recognition is proposed in this paper. Two distinct biometric traits have been under consideration: palmprint and knuckleprint. The palmprint image is chosen to be the host image. Knuckleprint biometric feature is selected to use as watermark hidden in the host image. Such that knuckleprint watermark not only protects palmprint biometric data, but also can be used as a covert recognition. Meanwhile, the bimodal biometrics recognition provides the improvement in the accuracy performance of the system. The experiment results demonstrate the effectiveness of the proposed method.

1 Introduction

Biometric techniques have inherent advantages over traditional personal identification techniques, but the problem of security and privacy of biometric data is extremely critical. If a person's biometric data is stolen, it is not possible to replace it as in case of a credit card, ID or password is stolen. Furthermore, while biometric data can provide uniqueness, it is not a secret. In order to promote the wide spread utilization of biometric technique, an increase of biometric security level is necessary. Encryption and watermarking are possible techniques to achieve this. There is a possibility that the decrypted data is intercepted, encryption does not provide the overall security of biometric data. Watermarking, which refers to embedding information into the host image, has been a very active research area recently. It can provide security even after decryption. The watermark, which embedded in the biometric data, provides another line of defense against illegal utilization of the biometric data. Meanwhile, it should be as a means to eliminate some of attacks to biometric system [1].

Since embedding watermark may change the inherent characteristics of the host image, verification performance based on watermarked images should not lead to significantly degrade compared to the original non-watermarked images. Ratha et al. [2] proposed a data hiding method applicable to wavelet compressed fingerprint images. Pantanti and Yeung [3] developed a fragile watermarking for fingerprint images verification. Jain et al. [1] describe an effective way of hiding actual fingerprint minutiae data in a synthetic fingerprint image or hiding Eigen face into a fingerprint image. These works are only on biometric data hiding and only involve the use of face and fingerprint images.

In recent years, multimodal biometric systems have received much more attention. Some of limitations of unimodal biometrics can be overcome by combining multiple biometric modalities [4]. Multimodal biometric systems not only result in a better improvement in performance but also ensure sufficient population coverage. Furthermore, these systems provide more security against spoofing. Because it is difficult for an attacker to simultaneous spoof the multiple characteristics of a genuine user.

In this paper, we propose a method that integrating watermarking technique and multimodal biometric system in order to provide more secure and reliable personal recognition. For the proposed scheme, two distinct biometric traits have been under consideration: palmprint and knuckleprint. Palmprint is referred to principal lines, wrinkles and texture in the inner surface of a palm (Fig.1a). Knuckleprint refers to the flexion shrinks in the inner skin of knuckles, especially the second knuckle for it contains more lines and more complex pattern than that of the first knuckle, meanwhile more stable than the third knuckle (Fig.1b)

The rest of the paper is organized as follow. The proposed scheme is described in Section 2. Multimodal biometrics verification is discussed in Section 3. Watermarking algorithm is developed in Section 4. Section 5 gives some experimental results. Section 6 presents conclusion.
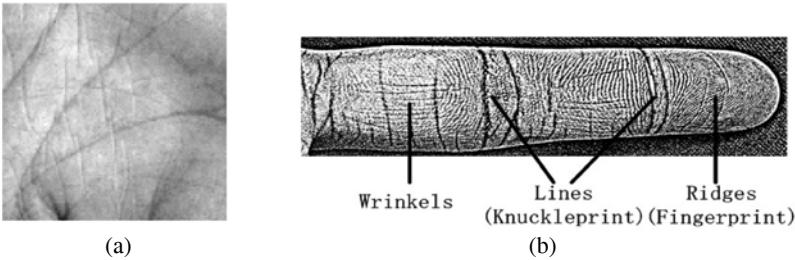

(a)   (b)

Fig. 1. Palmprint and knuckleprint: (a) palmprint (b) knuckleprint

2   The Proposed Scheme

The palmprint image is chosen to be the host image, to be stored or transmitted. The knuckleprint feature data are used as the embedded watermark. In our scheme, knuckleprint biometric features are selected to be used not only protecting palmprint biometric data, but also as a covert recognition. Since the attacker who intercepts the communication channel and obtains the carrier image is likely to think the palmprints as the only recognition means, this scheme provides an increased level of security. At the same time, palmprints and knuckleprint bimodal biometrics recognition also provide the improvement in the accuracy performance of the system.

The proposed framework is shown in Fig.2. For generation the watermark data, knuckleprint feature data are extracted from the knuckleprint image. The data are unique for different person and converted into a binary stream. They are then embedded into the palmprint image of the same person to protect the palmprint image. At the recognition stage, it contains four steps. Firstly, from the decoded watermark bits, the data hidden in the host image are extracted. Secondly, using the extracted data, knuckleprint recognition is performed. Thirdly, a recovered original host palmprint

image is obtained to perform palmprint recognition. Finally, the fusion of two biometric modal is done to give a final decision.
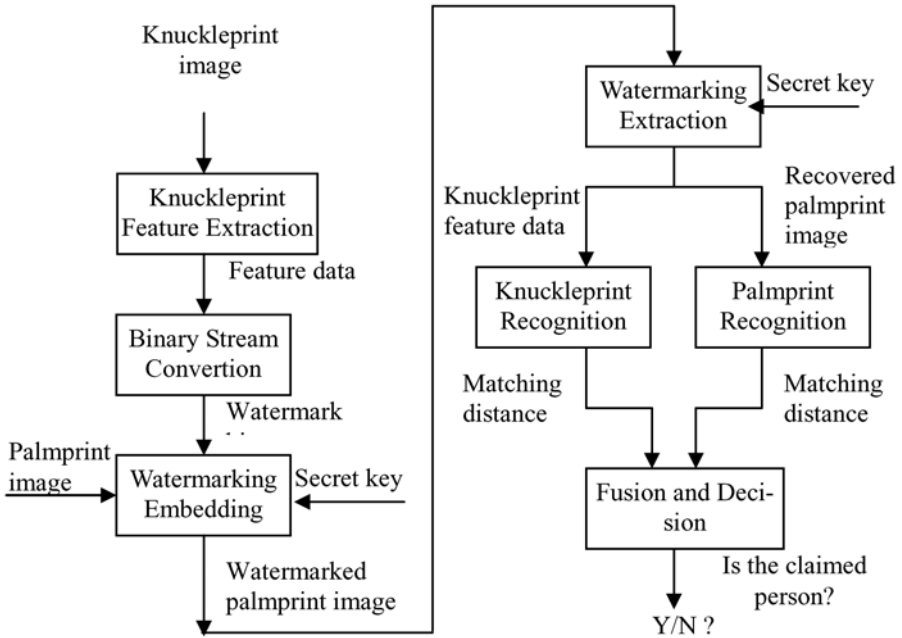

Fig. 2. The flow diagram of secure multimodal biometric verification scheme

3   Multimodal Biometrics Verification

3.1   Knuckleprint Verification

Our approach of knuckleprint verification is based on feature points extracted from knuckleprint images [5]. These feature points data is used as watermark embedded into the host palmprint images. When performing recognition, knuckleprint feature data detected from watermarked palmprint image is matched with another knuckleprint feature template to decide if the two feature sets belong to the same person.

Given a knuckleprint original image, to determine its identity, we propose the following steps and describe in detail below.

1. *ROI of knuckleprint determination.* Location information of knuckleprint lines can be obtained by projecting image $\mathbf{I_K}$ (Fig.3a) onto its *X*-axis (Fig.3b). Radon transform is employed to achieve this. Point *k* is the location of the second knuckle. A 64x64 knuckleprint ROI (region of interest) is determined based on point *k* (Fig.3c).
2. *Feature points extraction.* We define feature points as those points lying on the most significant knuckleprint lines. The feature points are extracted as follow:
   (i) A mask is designed and applied to the ROI image in order to make the lines clearer. A 1x7 mask **H** is defined as:

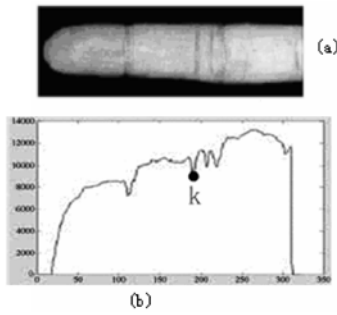$$\mathbf{H} = [-1, -1, -1, 6, -1, -1, -1] \tag{1}$$

Fig. 3. Knuckleprint features extraction (a) knuckleprint image; (b) radon projection of (a) onto X-axis; (c) knuckleprint ROI; (d) knuckleprint lines feature
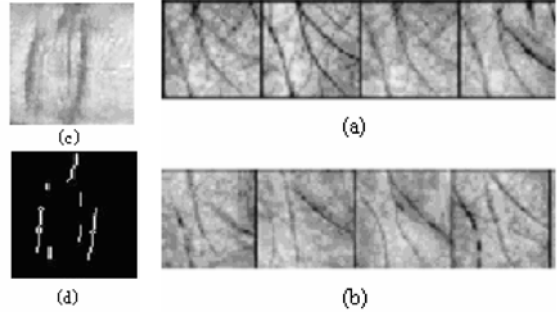
Fig. 4. Samples of palmprint database (a) samples of same class; (b) samples of different classes

(ii) A threshold T is applied to the masked image. The points whose values are greater than T are considered to be lying on lines, these points are still hold (for step (iii)), while the remaining points are set to 0 as background.

(iii) Two-dimensional adaptive wiener filter is used to remove noise. Then the filtered image $\mathbf{I}_{out}$ is binarized.

(iv) A set of morphological operations is performed in order to remove spur pixels, clean isolated points. So the most significant features of knuckleprint are preserved. The remaining line points are subsampled to obtain 50-100 points. These points are considered to be the feature points of knuckleprint (Fig.3d).

3. *Feature points matching*. The two sets of feature points are matched and a matching distance is computed. Here we propose an improved Hausdorff distance d_H, which can be expressed as:

$$d_H = d_{L_1 L_2} = \underset{a_i \in L_1}{mean}(d_{a_i L_2}) = \underset{a_i \in L_1}{mean}(\underset{b_j \in L_2}{\min}(d_{a_i b_j})) \tag{2}$$

Where $L_1 = \{a_1,...,a_m\}$ and $L_2 = \{b_1,...,b_n\}$ are two point sets, and $d_{a_i b_j}$ is Euclidean distance between two points.

If d_H exceeds an optimum threshold, the two point sets are considered to come from different person. Otherwise they are considered to belong to the same person.

3.2   Palmprint Verification

Just like many other biometric recognition, palmprint recognition is faced with the problem of high dimension feature space. Lu [6] extracted palm features by PCA and constructed the subspace by "eigenpalms", and the experimental results showed that PCA is efficient for palmprint feature extraction. 2D-PCA presented by Yang [7] recently, is a new approach of linear subspace based on so defined "image covariance matrix". Yang pointed out 2D-PCA could outperform PCA in covariance matrix estimation, thus achieve higher identification rate and lower computational complexity.

In our study, we found that feature dimension is still too high after using 2D-PCA. This may degrade the result of recognition. Therefore, a more efficient method I2D-PCA (Improved 2D PCA) is developed for palmprint recognition [8].

Let $\{\mathbf{I}_j\}(j=1,2,\dots,M)$ to be $m{\times}n$ (128×128 for our database) palmprint image sequences (Fig.4), the I2D-PCA feature $\mathbf{F}$ is determined by

$$\mathbf{F} = \mathbf{RIC} = (\mathbf{R}(\mathbf{IC})) = ((\mathbf{IC})^{\mathrm{T}}\mathbf{R}^{\mathrm{T}})^{\mathrm{T}}$$

$$\begin{cases} \{\mathbf{c}_1,\mathbf{c}_2,\cdots,\mathbf{c}_p,\mathbf{r}_1,\mathbf{r}_2,\cdots,\mathbf{r}_q\} = \arg\max tr(\mathbf{S}_t) \\ \mathbf{c}_i^T\mathbf{c}_j = 0, i \neq j, i,j=1,\cdots,p \qquad p < n \\ \mathbf{r}_i\,\mathbf{r}_j^T = 0, i \neq j, i,j=1,\cdots,q \qquad q < \min(m,p) \end{cases}, \tag{3}$$

Where, $\{\mathbf{r}_j\}$ are rows of the left transform matrix $\mathbf{R}$ and $\{\mathbf{c}_i\}$ are rows of the right transform matrix $\mathbf{C}$, and $\mathbf{S}_t = \mathcal{E}\,[(\mathbf{F}\text{-}\mathcal{E}\,\mathbf{F})(\mathbf{F}\text{-}\mathcal{E}\,\mathbf{F})^T]$ is the feature scatter matrix.

The main idea of I2D-PCA is: after decorrelating columns of the images by 2D-PCA, rows of the result are decorrelated to find the final feature. So, I2D-PCA can be implemented by using 2D-PCA twice (details can be referred to [8]):

- *Step 1*. Obtain 2D-PCA transform matrix $\mathbf{C}$ and feature $\mathbf{F}_1=\mathbf{IC}$ of each image;
- *Step 2*. Transpose $\{\mathbf{F}_1\}$ into $\{\mathbf{F}_1^{\mathrm{T}}\}$;
- *Step 3*. Using 2D-PCA for the second time and get $\mathbf{F}^{\mathrm{T}}=\mathbf{F}_1^{\mathrm{T}}\mathbf{R}^{\mathrm{T}}$, acquire $\mathbf{F}$ as the I2D-PCA feature.

Without loss of generality, Euclidean distance measure is used to determine the palmprint is belong to the genuine user or not. Suppose $\mathbf{F}_i$ and $\mathbf{F}_j$ are I2D-PCA features of two palmprint images $\mathbf{I}_i$ and $\mathbf{I}_j$. If $d(\mathbf{F}_i,\mathbf{F}_j)$ is smaller than a threshold, the two palmprints are considered to come from the same person.

$$d(\mathbf{F}_i,\mathbf{F}_j) = \sum_{x=1}^{p}\sum_{y=1}^{q}(\mathbf{F}_i(x,y)-\mathbf{F}_j(x,y))^2 \tag{4}$$

3.3   Biometrics Fusion

For the fusion of knuckleprint and palmprint recognition, we use Radial Basis Function (RBF) neural network, presented by Vatsa [9], for decision making. RBF networks are used as fusion since which require less training time and learning positive and negative samples. Vatsa [9] pointed out RBF networks obtain the highest accuracy compared to any other fusion algorithms. Knuckleprint and palmprint recognition give their respective matching distance based on the algorithms discussed above. These matching results are then fused using three layered RBF network. The output of the RBF network is either 0 or 1, where 0 stands for mismatch and 1 stands for match.

4   Watermarking Algorithm

4.1   Watermark Hiding Process

Firstly, the data of knuckleprint feature points to be hidden into the host image are converted to a binary stream. Each coordinate of individual feature points is converted into 6-bit binary representation. Such a representation can code integers between [0, 63] and this range is sufficient for X-coordinate and Y-coordinate of a point, for the knuckleprint ROI generated by our method is 64x64. Then we compute the 4th-level

DWT of the host image using the Mallet algorithm with biorthogonal 9/7 basis. We select {$HL_{4,1}$, $LH_{4,2}$, $HH_{4,3}$, $HL_{3,1}$, $LH_{3,2}$, $HH_{3,3}$, $HL_{2,1}$, $LH_{2,2}$, $HH_{2,3}$} to hide data. These subbands have most energy and information of all high frequency subbands. Furthermore, embedding the secret data in the middle-high frequency makes the algorithm more robust to general image processing [10].

Let $X_{l,f}(u,v)$ be a secret key selected wavelet coefficient, $l=1,2,…,L$ denotes the resolution level, $f=1,2,3$ denotes the frequency orientation of horizontal, vertical and diagonal, respectively. $\Delta_{l,f} = JND_{l,f}$ (JND, Just Noticeable Distortion) be the corresponding quantization interval, which classifies $X_{l,f}(u,v)$ into the corresponding interval. The quantization process is shown as Fig.5. The watermark bit w_i is embedded into the image as the following rule:
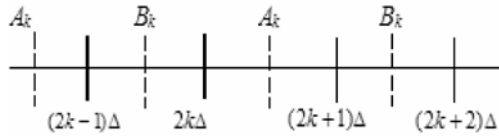
Fig. 5. The quantization process

$$X_{l,f}^*(u,v) = \begin{cases} A_k, & if\ w_i = 0\ and\ \arg\min\left|X_{l,f}(u,v) - A_k\right| \\ B_k, & if\ w_i = 1\ and\ \arg\min\left|X_{l,f}(u,v) - B_k\right| \end{cases} \quad (5)$$

Where, $A_k = (2k + \frac{1}{2})\Delta$, $B_k = (2k - \frac{1}{2})\Delta$, $k = 0, \pm 1, \pm 2, ...$ (6)

So $X_{l,f}(u,v)$ is replaced by the nearest A_k when $w_i = 0$, or by the nearest B_k when $w_i = 1$. The quantization error e is defined as $e = X' - X$. Obviously the quantization error is bounded in Δ, which means the data embedded into the image is perceptually indistinguishable from the original image.

4.2   Watermark Extraction

Let $\tilde{X}_{l,f}(u,v)$ be a modulated wavelet coefficient which has undergone attacks (incidental or malicious). The hidden data can be extracted without the original image by the following quantization process:

$$\tilde{w}(i) = \mod(ceil(\frac{\tilde{X}_{l,f}(u,v)}{\Delta}), 2) \quad (7)$$

Using secret key K, we can recover the original permutation of the watermark bits:

$$w' = impermute(\tilde{w}, K) \quad (8)$$

5   Experimental Results

In this section, experimental results for the method explained in the previous section will be presented. Factors such as decoding (i.e. watermarking extraction) accuracy and

matching performance are emphasized. Our database is set up using the hand image capture device. 1423 sample images from 73 hands are gathered in the database. Fig.6 is one of hand images in the database. Five samples of each person are taken out to form training set, while the remaining 1058 images are taken as testing set.

The size of hidden data is approximate 120 bytes. For all the 1423 sample images, we extracted knuckleprint feature points repectively and embedded into the corresponding palmprint ROI image. As a result, 1423 watermarked images were produced. Each knuckleprint data sets contained about 80 points. The palmprint ROI image (128x128) shown in Fig.7a is watermarked using the feature data extracted the same hand's knuckleprint shown in Fig.7b. The watermark data occupies 120 bytes, corresponding to the 80 feature points data. Fig.7c is the watermarked palmprint image.

Before watermarking, the accuracy of knuckleprint recognition is 96.8%, while palmprint recognition is 99.7%. From all these 1423 watermarked images, we can extract the embedded knuckleprint information with 100% accuracy. After watermarking, the accuracy of knuckleprint recognition is the same as before watermarking, while palmprint recognition based on recovered palmprint image is 98.5%. An accuracy of 99.8% is achieved for watermarking based knuckleprint − palmprint bimodal recognition system with enhanced biometric data security.

The experiment results indicate that our proposed watermarking method does not lead to significant accuracy degradation in both knuckleprint and palmprint recognition, though the palmprint recognition n introduce less degradation.
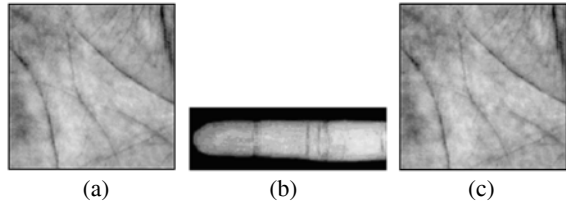

Fig. 6. Hand image in the database

(a)   (b)   (c)

Fig. 7. Watermark embedding: (a) host palmprint image; (b) watermark knuckleprint image; (c) embedded watermark palmprint image

6   Conclusion

Multimodal biometric systems are expected to be more reliable and able to meet the stringent performance requirement imposed by various applications. However, the security and privacy of the biometric data are important issue. In this paper, a secure multimodal biometric scheme using watermarking technique is proposed. Palmprint and knuckleprint are used to be multimodal biometric recogntion. Meanwhile, palmprint image is used as host image. Knuckleprint biometric features are selected to be watermark hidden in the host image. This method has the ability to increase the security of both the host and the hidden biometric data. Multimodal biometrics lead to high recognition accuracy. The experiment results demonstrate the effectiveness of the proposed method. High verification accuracy based on the watermarked biomodal biometrics is obtained.

References

1. A.K. Jain and U.Uludag, "Hiding biometric Data", IEEE Trans. on PAMI, vol.25, no.11, pp.1494-1498, 2003.
2. N.K. Ratha, J.H..Connel, and R.M. Bolle, "Secure Data Hiding in Wavelet Compressed Fingerprint Images", Proc. ACM Multimedia, pp.127-130, Oct. 2000
3. S. Pankanti and M.M.Yeung, "Verification Watermarks on Fingerprint Recognition and Retrieval', Proc. SPIE, vol.3657, pp.66-78,1999
4. A.K. Jain, A. Ross, D. Prabhakar, "An introduction to biometric recognition", IEEE Trans. on Circuits and Systems for Video Technology 14 (1) (2004) 4–20
5. Q. Li, Z.D. Qiu, D.M. Sun, J. Wu, "Personal Identification Using Knuckleprint", 5th Chinese Conference on Biometric Recognition, SINOBIOMETRICS 2004, Guangzhou, China, December 2004, Proceedings.680-689
6. G. Lu, D. Zhang, K. Wang, "Palmprint Recognition using Eigenpalms Features," Pattern Recognition Letters, vol. 24, pp. 1463-1467, 2003.
7. J. Yang, D. Zhang, A. F. Frangi, J. -y. Yang, "Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition," IEEE Trans. PAMI, vol. 26, pp. 131-137, 2004.
8. Q. Li, Z.D. Qiu, D.M. Sun, "I2D-PCA: An Efficient Subspace Method with Its Application to Palmprint Identification", accepted by 6th International Conference on Computer Vision, Pattern Recognition and Image Processing (CVPRIP2005).
9. M. Vatsa, R.S. Singh, P. Mitra and A. Noore, "Digital Watermarking based Secure Multimodal Biometric System", 2004 IEEE International Conference on Systems, Man and Cybernetics, pp.2983-2987
10. T. Liu and Z.D. Qiu, "A DWT-based Color Image Steganography Scheme", International Conference on Signal Processing (ICSP'02)

Automatic Configuration for a Biometrics-Based Physical Access Control System

Michael Beattie, B.V.K. Vijaya Kumar, Simon Lucey, and Ozan K. Tonguz

Carnegie Mellon University; Pittsburgh, PA 15213-3890, USA

Abstract. Selecting appropriate thresholds and fusion rules for a system involving multiple biometric verifiers requires knowledge of the match score statistics for each verifier. While this statistical information can often be measured from training data, that data may not be representative of the environment into which each verifier is deployed. To compensate for missing statistics, we present a technique for estimating the error rates of each verifier using decisions made after a system has been deployed. While this post-deployment data lacks class labels, it is guaranteed to be representative. Extracted error rates can be used to select appropriate fusion rules and search for thresholds that meet operational requirements.

1 Introduction

Physical access control represents an important application for biometric verification that is already available as an optional component in high end building security systems. We consider the next step in biometrics-based access control: multiple cooperating biometric verifiers to protect multiple security zones [1]. Decisions from these verifiers can be combined to balance security and convenience as appropriate for the resource being protected.

Configuration represents a major challenge for this type of system. Each biometric verifier must be configured with an appropriate threshold, and rules for combining decisions must be determined. Good thresholds ensure that each single verifier produces useful decisions. For example, if a threshold is too high then a verifier will generate mostly reject decisions—regardless of whether the claimant is authentic or an imposter. Standard techniques for combining decisions attempt to weight individual decisions by their relative accuracy [2, 3]. In both cases, knowledge of the False Accept Rate (FAR) and False Reject Rate (FRR) can be used to select appropriate configuration parameters.

In many applications, configuration of both thresholds and fusion rules can be completed using training data. Matching algorithms are be applied to a database of biometric samples and the resulting match scores can be used to select appropriate thresholds and measure error rates. In other applications, representative training data may not be readily available. Configuration using mismatched training data may lead to poor performance in the deployed system. We expect this to be the case in physical access control where biometric samples can differ significantly from building to building and even within a single building.

Further intensifying this problem, physical access control is often a single component of a larger building security system that can involve a variety of components from several different vendors. The parties building and installing these systems may not have the expertise to adequately configure biometric devices. In these cases, we expect systems to be left using default thresholds—a practice that we will show to be dangerous.

To facilitate a solution to this problem, we propose a strategy for estimating the FAR and FRR using verification decisions created during normal system operation. These decisions have the potential to provide accurate error rate information because they are formed using samples captured in the exact environment of interest. On the other hand, a lack of class labels (i.e. imposter or authentic) makes direct calculation of error rates impossible.

To overcome this challenge, we have derived an estimation technique based on the Expectation-Maximization (EM) Algorithm [4]. Using this this technique, it is possible to estimate error rates for each individual verifier based on sequences of decisions constructed as a claimant moves through a building. Our evaluation against synthetic verifier sequences with a wide variety of underlying error rates implies that this estimation technique can produce highly accurate estimates of individual verifier error rates.

Knowledge of verifier error rates provide an essential step toward enabling automatic system configuration. Comparing FAR and FRR to security and convenience requirements can indicate whether a local threshold should be adjusted. Additionally, error rates interpreted as estimators for class-conditional error probabilities can be used to determine how the sequence of decisions should be combined.

We proceed in Sect. 2 by further motivating the need for automatic configuration in biometrics-based physical access control systems. Sect. 3 presents the details of our error rate estimation, and Sect. 4 continues with the results of an empirical evaluation of that technique using a large number of synthetic data sets. Finally, we conclude our report in Sect. 5.

2   Motivation

Variation among biometric samples collected in different environments is the primary motivation for employing error rate estimation to configure a deployed set of biometric verifiers. Such variation can result from differing environmental conditions such as lighting and humidity. Variation due to more subtle conditions such as sensor positioning or individual user behavior is also possible. All of these variations will lead to different match score behavior in different buildings or even different sections of the same building. As a result, we cannot expect to find a single verification threshold to work in all environments. Furthermore, if we use the same threshold in two different environments, we may observe dramatically different error rates in each.

To demonstrate an extreme example of this phenomenon we used frontal face images from the Version 1 data set of the Face Recognition Grand Chal-
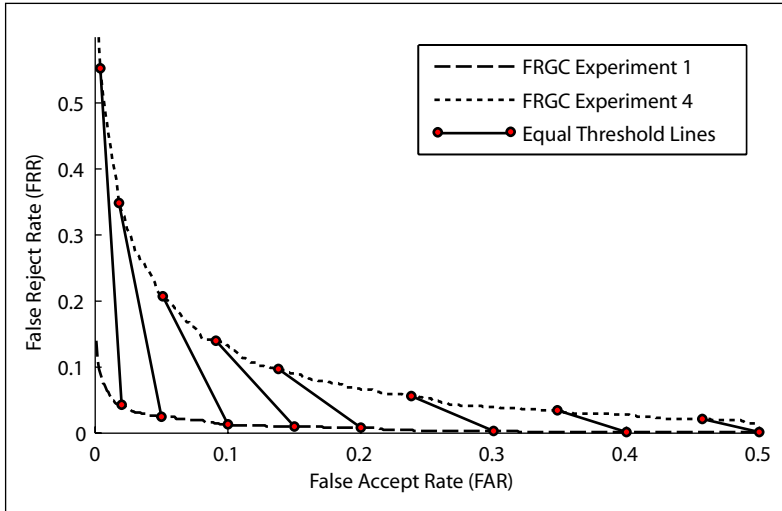
Fig. 1. Difference in performance between FRGC Experiments 1 and 4.

lenge (FRGC) [5]. This particular data set includes samples from two different collection scenarios: controlled and uncontrolled. Samples from the controlled scenario are captured in a studio setting with frontal lighting and a solid background. Samples from the uncontrolled setting are captured in a hallway and are plagued by shadows and irregular backgrounds. The FRGC protocol defines two experiments of interest for our evaluation. Experiment 1 compares controlled templates with controlled samples while Experiment 4 compares controlled templates with uncontrolled samples. To evaluate the difference in verification performance between these two experiments, we constructed match scores for each using a variant of Linear Discriminant Analysis (LDA) described in [6].

Experiments 1 and 4 can be interpreted as a single biometric verification product deployed to two different environments within the same building. In Fig. 2, we present two Receiver Operating Characteristics (ROCs): one for Experiment 1 and another for Experiment 4. Also visible in this figure are a series of equal-threshold lines that connect points on the two curves that correspond to the same threshold. It is clear from this figure that a particular threshold can lead to vastly different error rates in different environments. This figure also demonstrates that the change in performance from one experiment to the other is due largely to changes in FRR. This implies that the distribution of authentic scores is changing while the distribution of imposter scores remains relatively stable. We are unsure if this phenomenon is peculiar to this particular experiment or if it would be true for a wide range of algorithms and data sets.

Changes in accuracy for a fixed threshold suggest that the same biometric verifier deployed in two different locations may require different thresholds to meet the same operating requirements. It is also possible that in some environ-

ments, the verifier may not be able to meet the requirements at all. To motivate the need for automating the process of threshold selection, consider the installation of a biometrics-based physical access control system. Building owners purchase biometric verifiers from a vendor specializing in such devices (often via an intermediate system integrator). Because different thresholds are required for different environments, the problem of choosing a threshold is pushed from the biometric vendor down to the building administrators or system integrators. While biometric vendors are likely to have access to standard test databases, this is not the case for administrators or integrators. Their core competence is likely to fall outside the realm of biometrics, and they should not be required to maintain training data to assist in the configuration of biometric verifiers.

The solution we propose is analogous to a thermostat in a heating system. An identical furnace can be placed in two different buildings, and that furnace may need to be running more often in one building than the other. Building administrators are not required to calculate precisely how often the furnace must be running in each building. Instead, they determine a target temperature and the thermostat turns the furnace on and off as often as necessary to maintain that temperature. This control loop is made possible by the thermostat's ability to measure the current temperature. In much the same way, estimated verifier error rates enable an access controller to adjust verification thresholds and combination rules according to each verifier's operating environment.

3   Error Rate Estimation

The main challenge in estimating error rates from post-deployment verification decisions is a lack of class labels. We cannot be certain whether a particular decision was created from an authentic claimant or an imposter. As a result, we are unable to declare whether or not the decision is in error. There are a number of possible strategies to overcome this problem based on assumptions about the building environment. Most of these potential strategies are ad hoc in nature and rely on assumptions that may not be accurate. For example, we could estimate FRR by asking authentic claimants to report when they are rejected. This assumes that each time an authentic claimant is rejected, he or she will report that event exactly once. This is unlikely to be the case. Another simple strategy for estimating FRR is to assume that the vast majority of claimants will be authentic and simply count every reject decision as a false reject for the purpose of computing error rates.[1] In this case, an unexpectedly large number of impersonation attempts can heavily bias the estimate.

We propose a more structured approach to estimating error rates based on the Expectation-Maximization (EM) Algorithm [4]. The data available for this procedure are sequences of decisions from a single claimant with the true class (authentic or imposter) of that claimant interpreted as a hidden variable. Each sequence is created as a single claimant moves past multiple verifiers (e.g.,

---

[1] Of course, this does not imply that all claimants will be accepted as authentic; this assumption would be used strictly for error rate estimation.

through multiple doors). A per-subject access token connects a sequence of access attempts at different verifiers. We collect these sequences into vectors and label the jth decision sequence $\boldsymbol{u}_j$ and a the ith decision from that sequence as u_j^i.

Each decision u_j^i is an observation of the random variable u^i representing a decision from verifier i. We assume that u^i takes on a value of 1 to denote an accept decision (i.e. accept the claimant as authentic) and 0 to denote a reject decision. Similarly, the authentic and imposter classes are denoted by ω_1 and ω_0 respectively. Within this framework, we use the class-conditional decision probabilities in (1) and (2) as proxies for FRR and FAR. The EM Algorithm aims to find values for these probabilities that maximize the likelihood of the provided decision sequences.

$$\text{False Reject Probability: } p(\, u^i = 0 \mid \omega_1 \,) \tag{1}$$

$$\text{False Accept Probability: } p(\, u^i = 1 \mid \omega_0 \,) \tag{2}$$

To complete the estimation procedure, EM is initialized with a pair of initial probabilities for each verifier. These initial values may be provided by the biometric vendor or they may be nothing more than guesses. For example, the results in this paper all used initial error probabilities of 0.05. Given these initial values, the EM algorithm iteratively refines the estimated probabilities using (4) where $1(\,\cdot\,)$ is the indicator function. As usual, the posterior probability can be calculated with Bayes formula and an assumption of conditional independence as in (4).

$$p(\, \omega_k \mid \boldsymbol{u}_j \,) = \frac{p(\, \omega_k \,) \prod_{l=1}^{N} p(\, u_j^l \mid \omega_k \,)}{p(\, \boldsymbol{u}_j \,)} \tag{3}$$

$$p(\, u_i = k \mid \omega_k \,) \leftarrow \frac{\sum_{j=1}^{T} 1\left(\, u_j^i = k \,\right) p(\, \omega_k \mid \boldsymbol{u}_j \,)}{\sum_{j=1}^{T} p(\, \omega_k \mid \boldsymbol{u}_j \,)} \tag{4}$$

After constructing updated error rate estimates for each verifier, the process can be repeated. In each iteration, the class-conditional decision probabilities in (4) are replaced by the estimates from the previous iteration and their complements. After a small number of iterations (5, in our evaluation), we can usually converge to an accurate pair of error rate estimates for each verifier.

In (4) we have assumed that each available decision sequence was generated by the same series of verifiers. This need not be the case. It is possible to extend this methodology to allow for decisions from different sequences of verifiers. The only requirement is that each $\boldsymbol{u}_j$ contains at least one decision from verifier i. Using this strategy, we can envision a process running on the access controller that collects all decision sequences presented over a specific interval and use them to estimate error probabilities for each verifier in the building. By repeating this procedure at scheduled intervals, it is possible to monitor the accuracy of all verifiers in the building and adjust the thresholds and fusion rule accordingly.

One of the difficulties with this procedure is determining the *a priori* class probabilities in (4). We expect that for most buildings, the number of authentic claimants will be much larger than the number of imposters; however, the exact probabilities are unknown. A small number of imposters also presents a potential difficulty in using estimated FAR as a proxy for the probability of false accept. If only a few imposters attempt to gain access, then even perfect error rate measurements would be poor indicators of false accept probabilities. For the stated purpose of configuration, error probabilities are our primary interest.

Our solution to this problem is to generate additional decision sequences using pseudo-imposters. The notion of pseudo-imposters is common in the speech community and compares each sample to more than one template [8, 9]. In addition to comparing each collected sample to the template of the claimed identity, the system compares the same sample to an additional template for a different identity. If the number of actual imposters is small relative to the number of authentic claimants, this results in a nearly an equal number of authentic and imposter decisions. We have found that this preprocessing step can improve estimation accuracy significantly. As we will show in Sect. 4, submitting a balanced set of decision sequences to the estimation algorithm leads to accurate error rate estimates for a wide range of verifier configurations.

4   Evaluation

In a sense, the estimation technique that we have presented relies on the combined decisions of an ensemble of verifiers to determine whether a local decision is in error. The technique will work well when the ensemble is accurate, with estimation accuracy degrading as the ensemble error rate increases. For this reason, the evaluation of our error rate estimation technique should be parameterized by the accuracy of the ensemble of verifiers. In the following, we present an empirical evaluation of estimation accuracy for a wide range of verifier ensembles and present the results with respect to ensemble error probabilities.

Our evaluation applies the estimation technique from Sect. 3 to a number of synthetic data sets and measures the resulting estimation error. To evaluate estimation error for a single ensemble of verifiers, we first assign a pair of underlying error probabilities to each verifier in the sequence. Then we generate T binary decisions for each verifier according to the chosen error probabilities. Counting the number of actual errors produces a pair of error rates for each verifier: the measured FAR and FRR. We can then use raw decision data to construct decision sequences and submit these sequences to our estimation algorithm. Estimation error is calculated as the difference between estimated error rates and measured error rates. Generating a new set of decision data and repeating this procedure for a single set of underlying error probabilities allows us to measure the statistics of the estimator.

Fig. 4 presents the average magnitude error in estimating verifier error rates for a wide range of ensembles. For brevity, we limit our presentation to error rate estimation for the third verifier in each sequence. Plots for other positions would
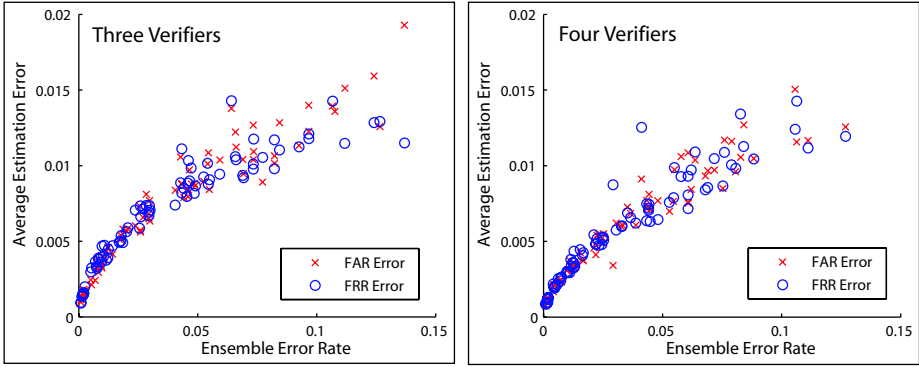
Fig. 2. Average magnitude of estimation error at the third verifier.

be nearly identical. Each point in the figure corresponds to a single set of error rates, and the error magnitude is averaged over 1000 data sets each containing $T = 1000$ decision sequences. The data contains an equal number of imposter and authentic decision sequences under the assumption that this balancing has been completed using pseudo-imposter decisions as described in Sect. 3. Ensemble error rates are predicted using underlying verifier error probabilities according to the minimum probability of error criterion.

For both the three and four verifier plots in Fig. 4, estimation errors less than 0.01 are possible for ensemble error rates less than 0.05. For ensemble error rates as high as 0.10, estimation error is still small. Although not presented in the figures, we have found that for ensemble error rates between 0.10 and 0.40 estimation error follows a linear trend with a moderate slope of 0.10. We expect that most reasonable ensembles will have combined error rates less than 0.15, but it is reassuring to see that even for large ensemble error rates the strategy maintains only linear growth in estimation error.

5   Conclusion

In the preceding, we have argued that representative training data is typically not available for evaluating the accuracy of a biometric verifier in a particular environment. This is particularly relevant in the context of physical access control, where verifiers may be deployed to many different environments. Typically accuracy information would be used to configure thresholds and decision fusion rules. In environments where a single configuration is insufficient, we believe that the selection of these configuration parameters should be completed automatically. To support this automatic configuration, we have presented a strategy for estimating the FAR and FRR of each verifier in an ensemble when class labels are unavailable.

These error rates can be directly applied to the selection of a fusion rule for combining decisions from multiple verifiers. We also expect that knowledge of these error rates can be used to search for appropriate thresholds for each local verifier. By comparing estimated error rates to specified requirements, the system can infer when thresholds require adjustment and the necessary direction of change. We do not provide further detail for adjusting thresholds in this manner, but we expect future work in this area to investigate this topic more thoroughly.

We expect that this work is applicable in a variety of applications beyond physical access control. An ability to estimate verifier error rates is an important tool for such automated configuration. Without knowledge of how well the system works under current configuration parameters, there is no way of knowing when parameters should be changed. Estimating verifier error rates provides feedback to remedy this situation.

Acknowledgments

References

1. Beattie, M., Kumar, B.V.K., Lucey, S., Tonguz, O.: Combining verification decisions in a multi-vendor environment. In: Audio- and Video-based Biometric Person Authentication (AVBPA). (2005)
2. Kittler, J., Hater, M., Duin, R.: On combining classifiers. IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI) **20** (1998) 226–239
3. Ross, A., Jain, A.: Information fusion in biometrics. Pattern Recognition Letters **24** (2003) 2115–2125
4. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society **39** (1977) 1–38
5. Phillips, P., Flynn, P.J., Scruggs, T., K.W. Bowyer, J.C., Hoffman, K., Marques, J., Min, J., Worek, W.: Overview of the face recognition grand challenge. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2005)
6. Chen, L., Liao, H., Lin, J., Ko, M., Yu, G.: A new LDA-based face recognition system which can solve the small sample size problem. Pattern Recognition **33** (2000) 1713–1726
7. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. Fisherfaces : Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **19** (1997) 711–720
8. Mak, M., Zhang, W., He, M.: Determination of a priori decision threshold for phrase-prompted speaker verification. In: International Workshop on Multimedia Data Storage, Retrieval, Integration and Applications. (2000) 96–103
9. Pierot, J.B., Lindberg, J., Koowaaij, J., Hutter, H.P., Genoud, D., Blomberg, M., Bimbot, F.: A comparison of a priori threshold setting procedures for speaker verification in the cave project. In: International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Volume 1. (1998) 125–128

Author Index