Pattern Recognition 42 (2009) 896 – 906

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Crease detection from fingerprint images and its applications in elderly people☆

Jie Zhou*, Fanglin Chen, Nannan Wu, Chenyu Wu

Tsinghua National Laboratory for Information Science and Technology (TNList), State Key Laboratory on Intelligent Technology and Systems, Department of Automation, Tsinghua University, Beijing 100084, PR China

ARTICLE INFO

Article history:
Received 24 March 2008
Received in revised form 17 September 2008
Accepted 22 September 2008

Keywords:
Fingerprint recognition
Crease detection

ABSTRACT

Conventional algorithms for fingerprint recognition are mainly based on minutiae information. But it is difficult to extract minutiae accurately and robustly for elderly people, and one of the main reasons is that there are many creases on the fingertips of elderly people. In this paper, we study on the detection of creases from fingerprint images, in which we treat the creases as a special kind of texture and design an optimal filter to extract them. We also study the applications of crease detection results to improve the performance of fingerprint recognition in elderly people, which include two aspects. First, it is used to remove the falsely detected minutiae. Second, the creases can be treated as a novel feature for elderly people's fingerprints, which is combined with minutiae feature to improve the performance. Experimental results illustrate the effectiveness of proposed methods.

1. Introduction

Fingerprint recognition is one of the most popular and reliable biometric techniques for automatic personal identification. It has received more and more attention and been widely used in civilian applications, such as access control and financial security [1,2]. Since elderly people occupies a rather high proportion (nearly 20 percent of the population) in modern society, it is very important to design a fingerprint recognition system with a satisfying performance for elderly people.

A fingerprint is the pattern of ridges and valleys on the surface of a fingertip. A microscopic feature of the fingerprint is called minutia, which means ridge ending or bifurcation. An ending is a feature where a ridge terminates. A bifurcation is a feature where a ridge splits from a single path to two paths at a Y-junction. In Fig. 1, a fingerprint is depicted, where the ridges are black and the valleys are white. The minutiae, ridge endings and bifurcations, are also shown. Most classical fingerprint verification algorithms [1–6] take the minutiae, including their coordinates and direction, as the distinctive features to represent the fingerprint in the matching process (see Fig. 1 for the illustrations). Minutiae extraction mainly includes the steps as below: orientation field estimation, ridge extraction or enhancement, ridge thinning and minutiae extraction. Then the minutiae feature is compared with the minutiae template; if the matching score exceeds a predefined threshold, these two fingerprints can be regarded as belonging to a same finger. See Fig. 2 for the flowchart of this kind of algorithms.

However, the quality of fingerprint images may be affected by a number of factors, such as creases, skin dryness, shallow/worn-out ridges, injuries and dirt. This is more serious for elderly people. For these fingerprint images, existing minutiae extraction algorithms are likely to detect spurious minutiae or miss some. As a result, the recognition rate of the fingerprint identification would decrease. Specially, many researches reported that the performance of the minutiae-based algorithms degraded heavily in elderly people's applications [1]. Therefore, it is a challenging problem to improve the performance for elderly people. In this paper, we will address the topic of crease detection from fingerprints and its application in elderly people.

There are always many creases existing in the fingerprints of elderly people, which are a kind of stripes irregularly crossing ridges and valleys in the fingerprints (see Fig. 3 for an example). They come into being because of the aging, manual work, accidents, etc. Some of them are permanent, while others are temporary, i.e., existing for a short term. Both the permanent and temporary creases will affect the orientation estimation, introduce the spurious minutiae and thus decrease the performance of fingerprint identification systems, even using popular preprocessing or post-processing steps (e.g. connecting the breaks by analyzing the orientation filed, or removing adjacent minutiae pairs with opposed directions).

☆ This work is supported by Natural Science Foundation of China under Grants 60332010 and 60875017, and Natural Science Foundation of Beijing under Grant 4042020. This research is also supported by National 863 Hi-Tech Development Program of China under Grant 2008AA01Z140. Parts of this work have been published in the Proceedings of CVPR 2003 and ICBA 2004.

* Corresponding author. Tel.: +86 10 62796881; fax: +86 10 62796881.
E-mail address: jzhou@tsinghua.edu.cn (J. Zhou).

0031-3203/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2008.09.011

So it is important to study this kind of pattern for fingerprints recognition. If we can detect the creases in advance, we could locate the spurious minutiae caused by creases and remove them. Consequently, the recognition result will be improved. Furthermore, if we can use the crease as a novel kind of feature for accessorial representation to elderly people's fingerprints, it may also result in an improvement of the performance. As we know, there are no other works aiming on this topic except for our earlier work [7,8].

In this paper, we will study the topic of fingerprint recognition based on crease detection. The contributions of our paper include: first, we use a parameterized rectangle to represent a crease, design an optimal filter as a detector, and employ a multi-channel filtering framework to detect creases in different orientations; second, we propose an algorithm to remove spurious minutiae detected by conventional fingerprint recognition algorithms and thus improve its performance; third, we treat the creases as a novel feature for elderly people's fingerprints, which is combined with minutiae feature to improve the performance. The experimental results on elderly people show that the performance of conventional fingerprint recognition algorithm can be improved by using the proposed algorithms.

Fig. 1. Example of a fingerprint (the ridges are black and the valleys are white), in which the minutiae (ridge endings and bifurcations) with their direction are given.

The rest of the paper is organized as follows: in Section 2, the problem of crease detection is studied; in Sections 3 and 4, two applications of using crease detection results are introduced, respectively, i.e., removing spurious minutiae and using them as novel features. Experimental results are presented in Section 5. Finally, the conclusions are drawn in Section 6.

2. Crease detection

Creases can be regarded as white bars [8] on a textured area, bounded by illusory contours [9,10]. However, because of the similarity between creases and valleys, it is a nontrivial task to extract creases in fingerprint images. Obviously Hough transforms [15] could not work.

Fig. 3. Example of a fingerprint of elderly people, in which creases are obvious.

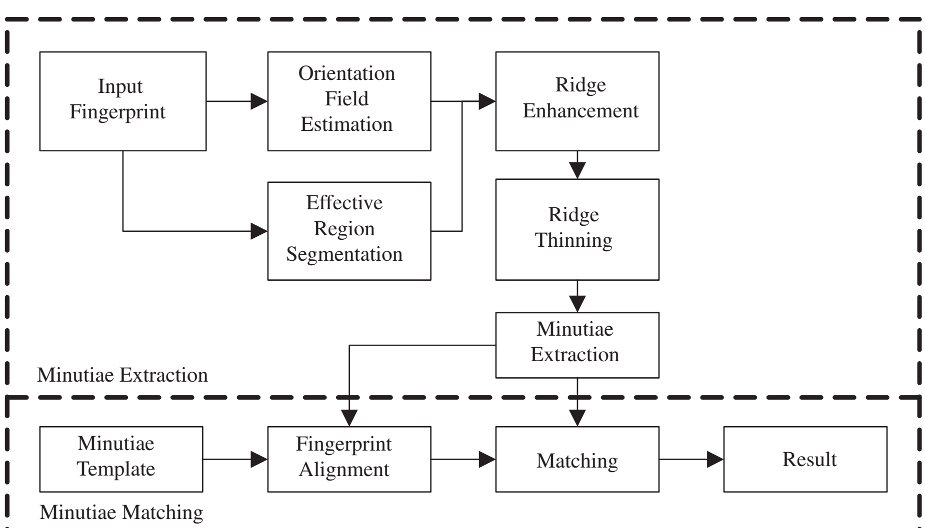

Fig. 2. Flowchart of conventional fingerprint recognition systems.

2.1. Crease modeling

Because of the similarity between creases and valleys, we give a brief description for ridges and valleys before defining the crease. In Ref. [2], Acton et al. have proposed a model to describe the ridges and valleys. Along the direction of a ridge or valley, gray values vary little. The direction of ridges and valleys at each point constitutes the orientation field of the whole fingerprint, which can be computed by using the algorithm proposed in Ref. [3]. In this paper, we call such a direction as *texture's direction*. On the orthogonal direction of the *texture's direction*, there is a prominent periodical variation in gray level through the ridges and valleys. Compared with ridges and valleys, a crease appears as a stripe irregularly crossing the ridges and valleys. The crease contains similar gray level to the neighboring valleys. There is a large difference between the crease and neighboring valleys in direction. Since most creases are like straight stripes, we employ a parameterized rectangle to represent a crease. And most importantly, both the similarity and difference between the crease and neighboring valleys are formulated as constraints to the representation.

In Fig. 4, an ideal model is used to represent a crease. Let $L(C_x, C_y, w, l, \theta)$ denote a crease, where l, w, θ and (C_x, C_y) are the length, width, direction and central point of L, respectively (see Fig. 4). The rectangle should satisfy some constraints as below:

$$w > TH1, \quad l > TH2 \quad \text{and} \quad l/w > TH3, \tag{1}$$

$$m\{I(x,y)\} > TH4 \tag{2}$$

and

$$\varphi > TH5, \tag{3}$$

where $I(x,y)$ is the gray level of the pixel, (x,y), $m\{I(x,y)\}$ is the average gray level in L, φ is the angle between the crease and the texture direction, and $TH1$, $TH2$, $TH3$, $TH4$ and $TH5$ are predefined thresholds. Eq. (2) indicates that the average gray level of the crease should not be too small. Eq. (3) indicates that the crease's direction should have a large enough difference with the *texture's direction*.

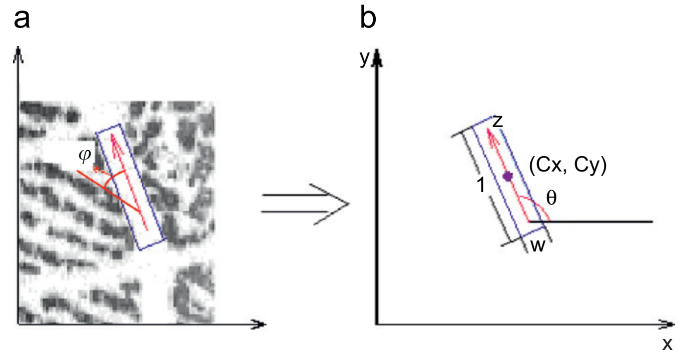

Fig. 4. Illustration of a crease: (a) φ is the angle between the crease and the texture direction, and (b) l, w, θ and (C_x, C_y) are the length, width, direction and central point of the crease, respectively.

2.2. Crease detection using an optimal filter

We regard a filter as a robust crease detector, if it has large responses to the crease region and small responses to the neighboring fingerprint texture. The optimal filter is therefore obtained by maximizing the difference between the responses to the crease and the neighboring fingerprint texture.

In order to find an optimal filter, we firstly study the fingerprint images containing creases. Fig. 5 shows the comparison of fingerprint texture and crease. Suppose their directions are orthogonal to **x** coordinate, thus the optimal filter's direction is along **x** coordinate and it can be regarded as a function $f(x)$ only related to x.

We assume the widths of the ridge and valley are equal, denoted as w_0 (see Fig. 5 (a)). In Fig. 5 (b), an ideal model is used to represent a fingerprint image containing a crease, where w denotes the crease's width and φ denotes the cross angle between the crease and fingerprint texture. Because in applications, the filter's direction is discrete but the crease's is continuous, the difference between the optimal filter and crease in direction may not equal to $\pi/2$. We use $\beta+\pi/2$ to denote the difference (see Fig. 5 (c)), and β is a small angle value.

The main difference between the two textures is along the **x** coordinate. When w, $w0$, φ or β changes a little, the projection of the image Fig. 5 (b)) to **x** coordinate, i.e., the intensity histogram $h(x)$, along **x** coordinate, will also change a little. We count a cumulated histogram with w, $w0$, φ and β changing (see Fig. 5 (d)). Each variable varies in a certain range. The cumulated histogram, denoted as $H(x)$, can be regarded as an expected distribution of the signal (crease) and noise (fingerprint texture), along **x** coordinate. Since $H(x)$ represents the crease with $|x| < \bar{w}/2$, we use a Gaussian function $G_x(0, \sigma_1^2)$ to approximately represent the crease, where $\sigma_1 \approx \bar{w}/6$ (the probability of $|x| < \bar{w}/2$ is 99.73%). Let $n(x) = H(x) - G_x(0, \sigma_1^2)$ denote the fingerprint texture. As shown in Fig. 5 (d), we also use a Gaussian function $G_x(0, \sigma_2^2)$ to approximate $n(x)$ with $\sigma_2 \gg \bar{w}/2$. Thus we have $H(x) \approx G_x(0, \sigma_1^2) + G_x(0, \sigma_2^2)$.

Therefore, the optimal filter can be obtained by maximizing the response to $G_x(0, \sigma_1^2)$ and minimizing that to $G_x(0, \sigma_2^2)$, or maximizing the response to $s(x) = G_x(0, \sigma_1^2) - G_x(0, \sigma_2^2)$. From signal processing theory [9], a maximal response to $s(x)$ could be reached through a filter with the form of $s(x)$. Furthermore, it has been proven that the difference between two Gaussian functions with different variances could be approximated by a two-order derivative Gaussian function, when the variances are appropriately selected [10,11]. Thus the optimal filter could be approximately formulated as a two-order derivative Gaussian function as

$$f(x) = A \exp\left\{-\frac{x^2}{2\sigma^2}\right\}(\sigma^2 - x^2), \tag{4}$$

where σ is the variance, A is a constant parameter and $A > 0$. Eq. (4) is a 1-D filter, and the 2-D form is as

$$F(x,y) = A \exp\left\{-\frac{x^2 + \eta y^2}{2\sigma^2}\right\}(\sigma^2 - x^2), \tag{5}$$

where η is a parameter. To detect creases in any directions, we rotate the filter $F(x,y)$ by an angle γ. Let $u = x\cos\gamma + y\sin\gamma$, $v = -x\sin\gamma + y\cos\gamma$, the filter with γ-direction is

$$F(x,y)_\gamma = A \exp\left\{-\frac{u^2 + \eta v^2}{2\sigma^2}\right\}(\sigma^2 - u^2). \tag{6}$$

2.3. Detection framework

With the optimal filter, we devise a multi-channel framework to detect creases in any directions. The framework is shown in Fig. 6.

(1) Firstly use a Gaussian filter to compute a mask image for the input fingerprint image. And then valid regions in the fingerprint marked by the mask image will be further processed [12].
(2) Then select 12 channels or directions for $F_\gamma(u,v)$, where γ equals $0, \pi/12, \ldots, 11\pi/12$, respectively. The response images of these filters are denoted as $I'_1, I'_2, \ldots, I'_{12}$, respectively.

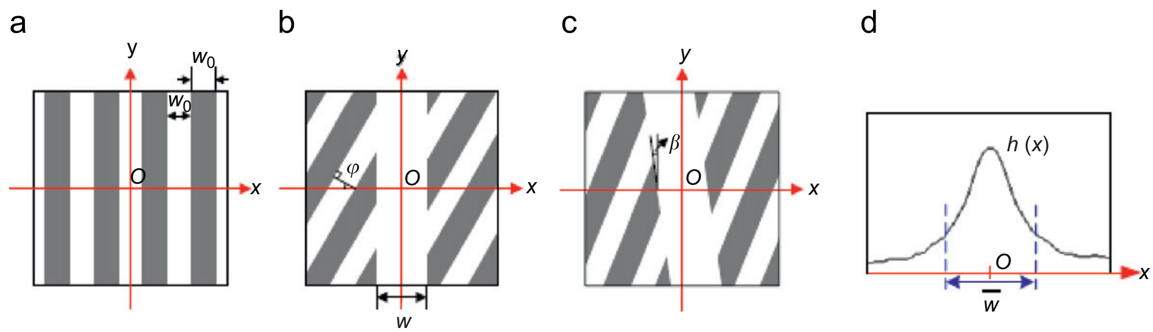

Fig. 5. (a) A rectangular representation for fingerprint texture, (b) an ideal model representing a fingerprint image containing a crease, (c) an example that the direction difference between the crease and optimal filter is equal to $\pi/2+\beta$and (d) an accumulated intensity histogram along the x coordinate, where $\bar{w}$ is average width of creases.

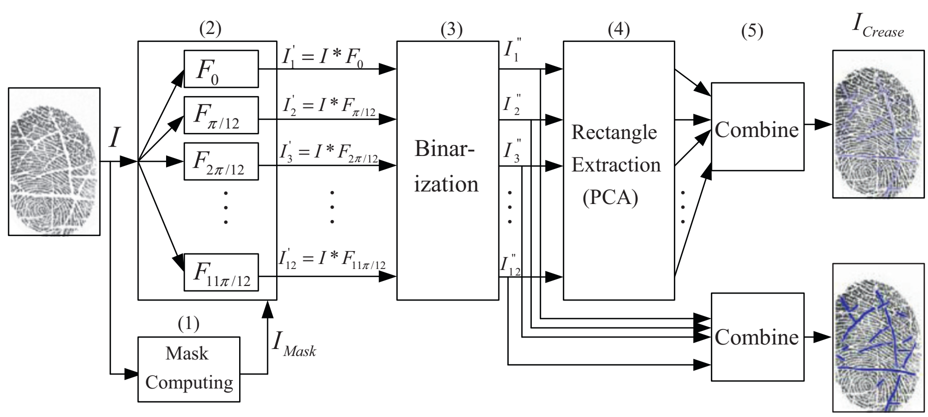

Fig. 6. The flowchart of crease detection.

(3) In each channel, we select a threshold, 200, to binary the response image. It is selected through experiments. We have tested on different thresholds, and found that thresholds in [180, 215] have good performances. We finally selected 200 for the classification. Regions with larger gray level than the threshold are then selected as crease candidates.

(4) For each candidate region, principal components analysis (PCA) is used to estimate the parameters of the rectangle representing a crease (Section 2.1). The point (Cx, Cy) is the center of the region. θ is equal to the direction of the axis with the larger eigenvalue. l and w are computed as the average length and width of the region, using the larger and smaller eigenvalues, respectively. Constrained by Eqs. (1)–(3), we remove some candidates and regard the remainders as valid creases.

(5) Finally combine creases extracted in each channel and get the final result I_{Crease} (see Fig. 6).

As we know, using more channels can provide a better performance, but results in less generalization and much higher cost of computation. We have compared different number of channels, and then chose to use 12 channels as a tradeoff.

3. Improve recognition performance by removing spurious minutiae

Conventional fingerprint recognition methods based on minutiae were reported to have a satisfying performance [5]. However, this satisfying performance is based on the good quality of fingerprint images in the database. In FVC2000 report [13], it was pointed out that the conventional methods have disadvantages if the database has poor-qualitied fingerprints, especially fingerprints of elderly people who have many creases. The main reason that creases affect the recognition rate is that they bring many spurious detected minutiae.

3.1. Two categories of creases

According to the cause, creases can be classified into two categories. The first one is caused by drying crack. In spring and winter when the weather is dry, if the finger is not taken good care of, the fingerprint will become dry and get cracked. As a result the fingerprint image will have many creases. In summer and autumn when the weather is wet, this kind of crease will disappear. An example of this kind of crease is shown in Fig. 7. The second kind of crease normally appears in the fingerprints of elderly people. It is caused by skin aging and abrasion, and cannot be recovered. However, it is rather difficult to classify a crease into these two categories by human.

Since the first kind of crease may appear sometimes, and may disappear in other times, the spurious minutiae caused by them are not steady. Although the second kind of crease will not disappear, when the fingertips' restrain strength changes, the width of the crease will change. When the restrain strength is weak, the width of the crease is wide; otherwise, the width is narrow (see Fig. 8 for an extreme example). The changing width of the crease also causes the spurious minutiae to be unsteady.

From the above analysis, the spurious minutiae caused by crease are not usually steady. Thus, it is very important to develop a method

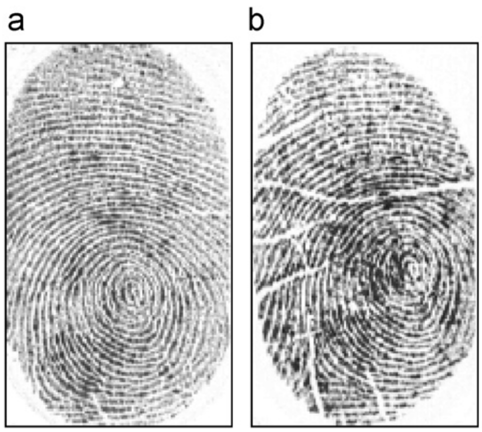

Fig. 7. The crease changes by season: a same fingerprint in (a) autumn, and (b) spring.

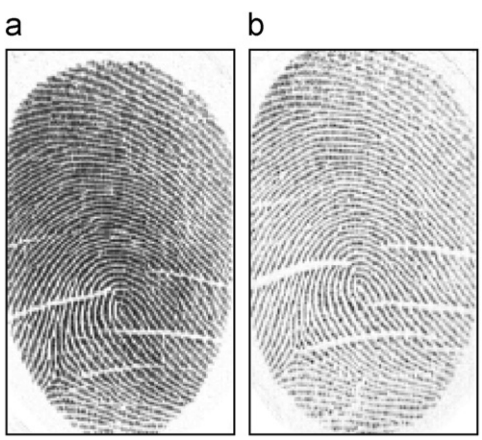

Fig. 8. The width of crease changes: a same fingerprint when (a) the restrain strength is strong, and (b) the restrain strength is weak.

to remove these spurious minutiae to improve the recognition performance.

3.2. Removing spurious minutiae based on crease detection

After the conventional minutiae detection method, we can remove spurious minutiae around the creases (see Fig. 9 for an example). The main point for removing the spurious minutiae is based on the distance between the minutia and the crease. If the distance is smaller than a pre-defined threshold (5 pixels), the minutia is judged to be a spurious one. Denote the minutiae as $p_i(x_i, y_i), i = 1, 2, \ldots, m$, where m is the number of minutiae. Let M be the set of spurious minutiae, $N(p_i)$ be the neighborhood of $p_i(x_i, y_i)$, and C be the area of all creases. We have the following rule to judge a minutia as spurious:

$$p_i(x_i, y_i) \in M \quad \text{if } \exists (x_t, y_t) \in \{N(p_i) \cap C\}. \tag{7}$$

The effect of removing spurious minutiae is shown in Fig. 10. The spurious minutiae is marked in red (in Fig. 10(b)) and removed from the minutiae template (see Fig. 10(c)).

Fig. 9. An example of spurious detected minutiae by crease.

4. Fingerprint recognition using minutiae and crease

In this section, we will analyze the stability of the crease information, and utilize it into the matching stage by combining them with minutiae information.

4.1. Analysis the stability of crease

As mentioned in Section 3.1, there are about two categories of crease. The first kind of crease is not steady in a long time (see Fig. 7). However, in a short time (in our study, after observing the fingerprints in the database, we conclude that this time is longer than a month), the first kind of crease is steady in position and shape. Fig. 11 shows two finger's fingerprints collected at three different times in one month. From the figure, we can see that the crease almost does not change in a short time. The second kind of crease is caused by skin aging and abrasion, and it will not disappear (see Fig. 8). Thus both kinds of crease are rather steady in a short time, especially for elderly people. We can make use of crease as a steady feature in the often used automatic fingerprints recognition systems. There is no contrariness between this usage and the usage of crease detection on removing spurious minutiae. Spurious minutiae are always unsteady due to the changing width of crease, and they are unsuitable to be used as a feature for the fingerprints' description.

4.2. Crease matching

Matching methods can be mainly divided into three classes [14]: (1) algorithms that use the image pixel values directly; (2) algorithms that use low-level features such as edges and corners and (3) algorithms that use high-level features. Methods which use the image pixel values directly, such as correlation methods, are sensitive to any shift and rotation between images, thus they are not widely used. The drawback of high-level matching methods is that high-level features need to be extracted first and identified, which is a rather difficult task. We treat crease-based fingerprint matching as a problem of low-level matching. Compared to the other two methods, low-level matching method is more robust than methods that use the image pixel values directly, and its features are easier to extract than using high-level matching methods.

Among all the low-level matching methods, chamfer matching is a state-of-art algorithm. Chamfer matching was first proposed in 1977 [15], and it is widely used to match edges in two different images. Chamfer matching is robust in handling imperfect (noisy, distorted, etc.) data, so it is very suitable for the comparison between creases.

Fig. 10. An example of removing spurious minutiae based on crease detection: (a) the fingerprint image with the creases, (b) the minutiae detected by the conventional method, in which the spurious minutiae are marked in red and (c) the final result by using the proposed method.

Fig. 11. The steady crease feature in a short time: the three columns were collected at three different times in a month.

To compare two fingerprints' creases, the first step is the alignment of these two fingerprints. It can be done in the same way as in conventional fingerprint algorithms, in which the alignment is mainly based on minutiae information [2–4,12]. In our study, we choose Hough-transform based approach [16,17] to do the alignment due to its low-computational cost. In the matching step, the

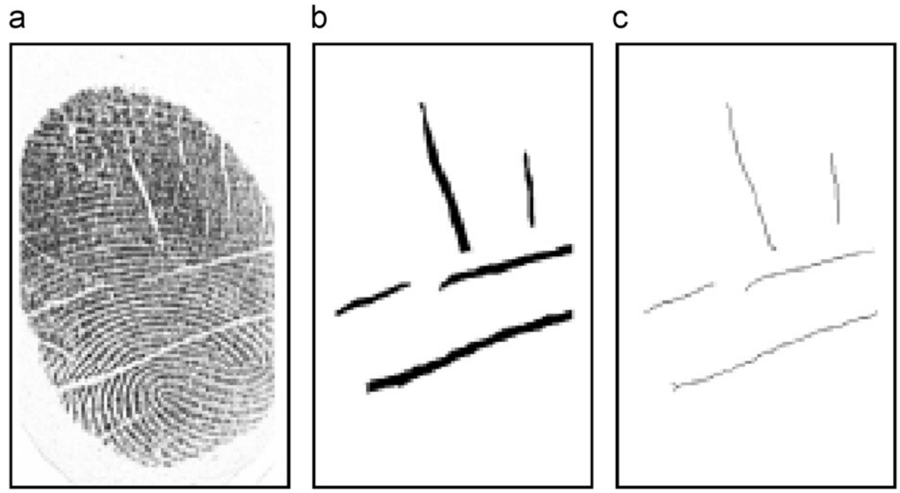

Fig. 12. Refining the crease to get the thinning results: (a) the fingerprint image, (b) the crease and (c) the thinning results.

correlation between two aligned creases, A and B, is computed as below.

(1) Get the binary image A' from A. Suppose C is the set of crease points in A, and a pixel p in A' is assigned a value as:

$$A'(p) = \begin{cases} 1, & p \in C, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

In the same way, B is transformed to B'.

(2) The width of the crease may change when the pressure strength changes (see Fig. 8), thus the width information cannot be used for matching. We refine A' and B' to get the thinning results, A'' and B'' [18,19]. See Fig. 12 for an example.

(3) In Chamfer matching, one of the two images is called the *pre-distance image* and the other the *pre-polygon image* [14]. The *pre-distance image* is converted to another image D by using distance transform (DT). This transformation takes a binary feature image as input, and assigns a value to each pixel in the image according to the distance to its nearest feature. Suppose A'' is the pre-distance image, O is the set of feature points in A'', a pixel p in D is assigned a value as

$$D(p) = \min\{\mathrm{dis}(p, q), q \in O\}, \tag{9}$$

where $\mathrm{dis}(p, q)$ is the distance between p and q. As discussed in Ref. [20], the true Euclidean distance is computational expensive, therefore an approximation is used instead. In Ref. [21], the authors compared 2–3 DT, 3–4 DT and city-block distance. They pointed out that, if 2–3 DT or city-block distance was used, the maximum difference from the Euclidean distance was about 13% and 59%, respectively. If the 3–4 DT was used instead, the maximum difference was reduced to 8%. Thus, we use 3–4 DT to approximate Euclidean distance in this study.

(4) After D is computed, the similarity between two crease images, A'' (pre-distance image) and B'' (pre-polygon image), can be measured by using the root mean square average [14]:

$$d = \frac{1}{3}\sqrt{\frac{1}{N}\sum_{i=1}^{N} d_i^2}, \tag{10}$$

where N is the feature points number in the pre-polygon image B'', and d_i is the distance value in D corresponding to the i th feature point of B''.

4.3. Combine crease matching with minutiae matching

Since the fusion of classifiers may allow alleviation of problems intrinsic to individual classifiers, the matching performance could be improved by combining the output from minutiae-based matching and crease matching.

A variety of combination rules have been proposed, and they can be mainly categorized into two kinds [22–25]: (1) methods based on heuristic rules, such as product rule, sum rule, max rule, min rule, median rule and majority voting rule; (2) methods based on probability, such as Neyman–Pearson rule. But these rules are unsuitable for our research.

The discriminant ability of creases is much smaller than minutiae. Moreover, they are sometimes steady and unsteady in other times. So, if we combine the minutiae matching with creases matching, the fusion scheme should mainly rely on the matching results of minutiae. When the result of minutiae matching is good enough, creases matching can contribute to the final fusion score. Otherwise, the final score cannot be added even though a very good matching between creases is obtained. Based on these points, we induct a specific fusion formula as below:

$$S = f(S_{minutiae} + S_{crease} \times S_{minutiae}), \tag{11}$$

where $S_{minutiae}$ [26] and S_{crease} are the similarity score of minutiae matching and crease matching, respectively. The function $f(x)$ is defined as

$$f(x) = \begin{cases} x, & x < 1, \\ 1 & \text{otherwise.} \end{cases} \tag{12}$$

When there are no creases in any of the two fingerprints to be compared, $S_{crease} = 0$, according to Eq. (11), the fusion score S equals to the similarity score $S_{minutiae}$ which is totally based on minutiae. When there are creases in the two fingerprints , the fusion score S is the sum of two part. The first part is $S_{minutiae}$ based on minutiae, and the second part is the product of $S_{minutiae}$ and S_{crease}. Therefore, the value added to S is determined both by $S_{minutiae}$ and S_{crease}. The added value is high only when these two scores are both high.

5. Experimental results

5.1. Fingerprint database

The experiments are conducted on two databases that consist of elderly people's fingerprints. All the fingerprint images are captured

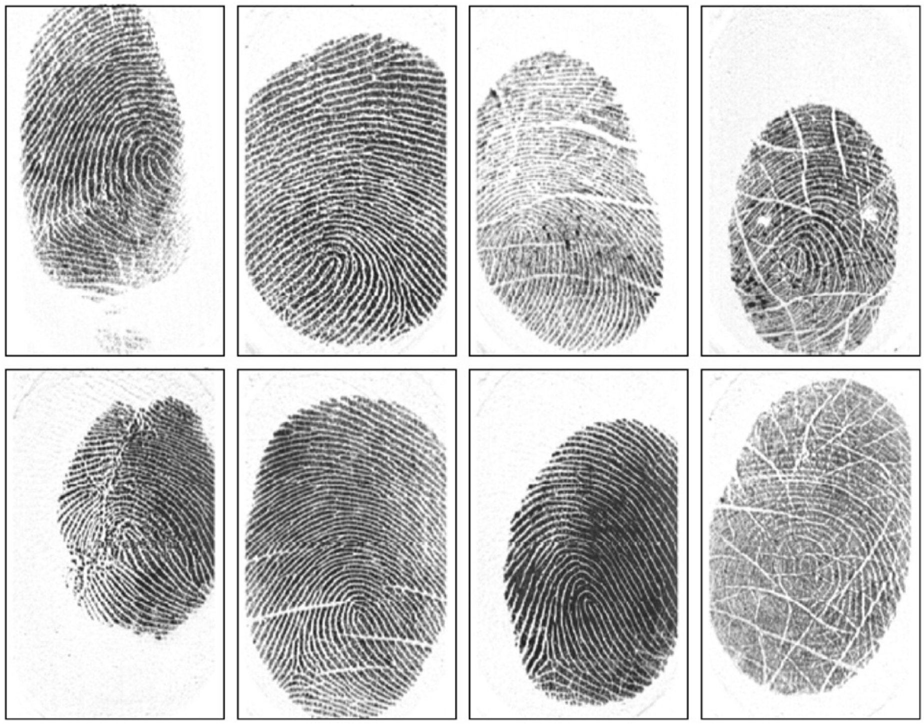

Fig. 13. Some fingerprints in our database.

with live-scanners (image sized 320×512). In our study, the elderly age group is defined as retired people. All the fingerprints were from an association of retired people. The oldest volunteer is 95 years old, and the youngest volunteer is 46. Their average age is 67. In the database, 46.15% volunteers are female, while 53.85% are male.

The first database consists of 3000 fingerprint impressions, which are from 375 different fingers, and 8 fingerprints per finger. All the eight impressions are collected in one month. The second database consists of 3520 fingerprint impressions, which are from 220 different fingers, and 16 fingerprints per finger. The former eight impressions are collected in the spring of 2005; and the latter eight impressions are collected in the autumn of 2005. We call the first database non-time-cross (NTC) database, and the second one time-cross (TC) database. In both database, the fingerprint images vary in different qualities. In them, more than 80% of these images are suffering the affection from creases. In Fig. 13, some fingerprints in our database are listed.

All the threshold values (in Eqs. (1)–(3)) are selected manually. We selected some fingerprints of elderly people, and labeled the creases manually. We got all the values of $w, l, l/w, m\{I(x,y)\}, \varphi$ from each crease, and the thresholds are set as the smallest values ($TH1 = 3$, $TH2 = 12$, $TH3 = 3$, $TH4 = 210$, $TH5 = \pi/18$), respectively. If the thresholds are chosen greater, the false detected creases will be fewer, but some creases will be easily missed.

The conventional minutiae-based fingerprint recognition is the same as that used in Ref. [26].

5.2. Crease detection

Some detected creases are shown in Fig. 14. The original fingerprint images are listed in the first column. In column 2, creases are represented by irregular regions. In column 3, creases are represented by parameterized rectangles. The width of the rectangle is the average width of the crease. Since some creases are not straight, there may be some error for computing the average width by using PCA.

It has difficulties to study the statistical accuracy of the crease detection method. It is similar to the problem of edge detection, which cannot be evaluated by comparisons between the detected results and the ground truth. However, we have tried to manually label 100 fingerprints of elderly people and got the estimated accuracy. The detection rate (defined as the ratio of truly detected creases to all labeled creases) is 86.43%.

5.3. Removing spurious minutiae to improve the performance

Fig. 15 shows the receiver operating curves (ROC) plotting *FAR* versus *FRR* of conventional minutiae matching scheme (solid line) and the proposed scheme of removing spurious minutiae using crease detection (dash line), on NTC and TC database, respectively. False rejection rate (FRR) is defined as the percentage of imposter matches in all genuine pairs, while false acceptance rate (FAR) is defined as the percentage of genuine matches in all imposter pairs. The results show that removing spurious minutiae by crease detecting can largely improve the performance.

5.4. Combine crease matching with conventional minutiae matching

Fig. 16 shows the ROC plotting *F* AR versus *F* RR of conventional minutiae matching scheme (solid line) and the proposed combination scheme (dash line), on NTC and TC database, respectively. A better performance can be obtained by combining the crease information with minutiae matching. *F* RR can be reduced significantly by using the combination scheme against the minutiae-based matching only.

5.5. Experiments on a general population

We have applied the proposed algorithm across a general population (about 20% are elderly people, and 80% are young people).

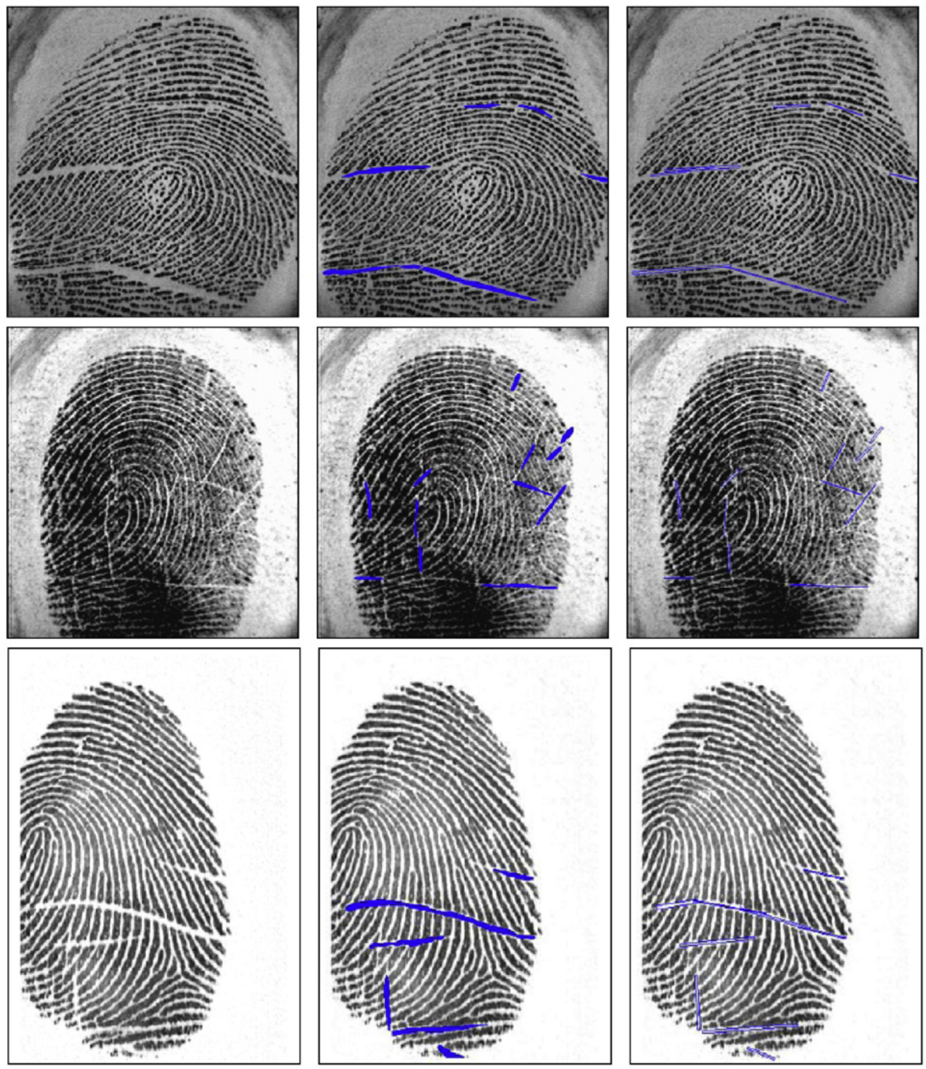

Fig. 14. Some crease detection results. Column 1 contains input fingerprint images. Column 2 lists the extracted creases. Column 3 lists the extracted creases, denoted as rectangles.

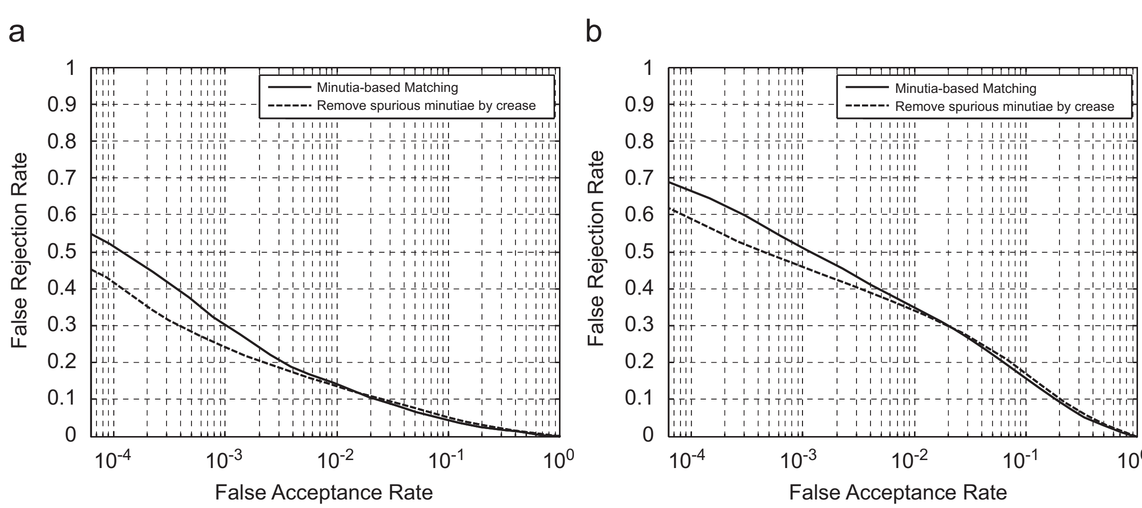

Fig. 15. ROCs of the minutiae matching scheme (solid line) and the proposed method of removing spurious minutiae (dash line), on (a) NTC database, and (b) TC database, respectively.

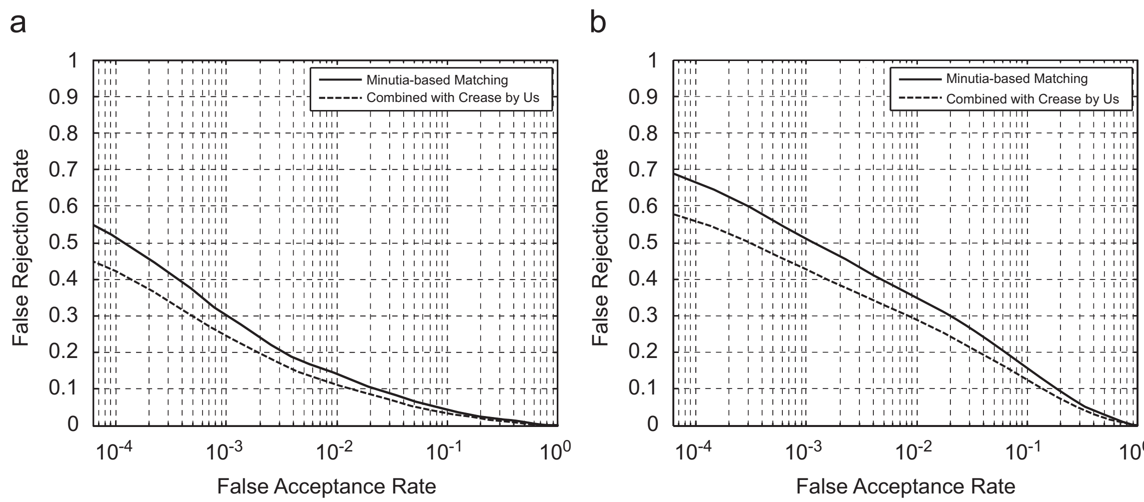

Fig. 16. ROCs of the minutiae matching scheme (solid line) and the combination (combined with crease) scheme (dash line), on (a) NTC database, and (b) TC database, respectively.

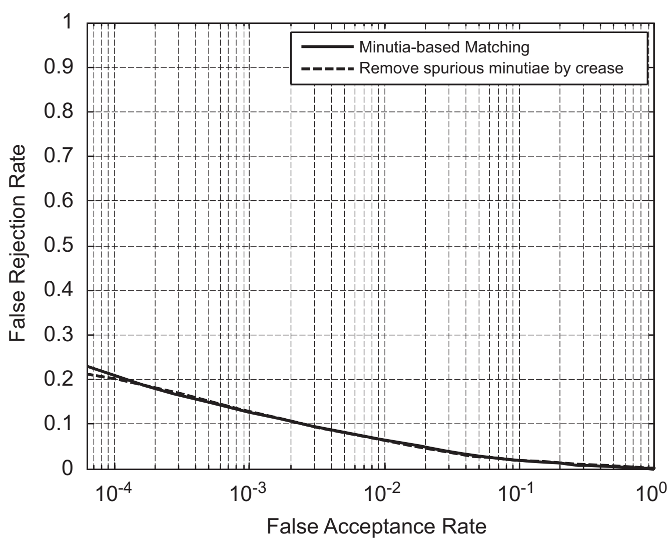

Fig. 17. ROCs of the minutiae matching scheme (solid line) and the proposed method of removing spurious minutiae (dash line), on a general population database.

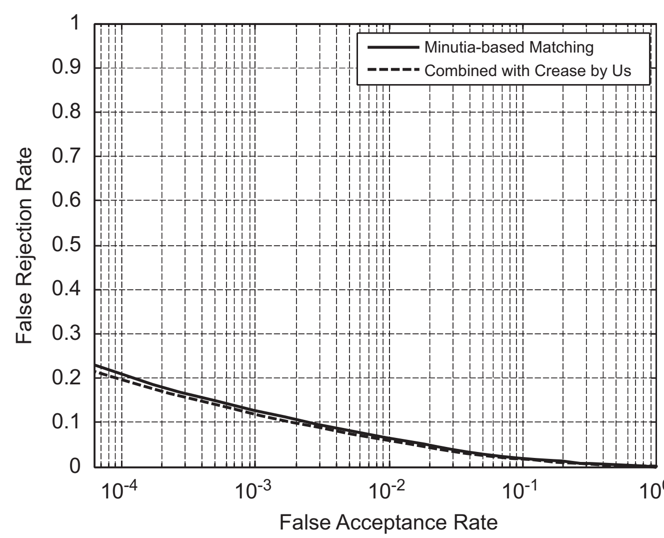

Fig. 18. ROCs of the minutiae matching scheme (solid line) and the combination (combined with crease) scheme (dash line), on a general population database.

The experimental results (Figs. 17 and 18) show that the algorithm can improve the performance a little, because the general population contains a large amount of fingerprints of young people that have few creases.

6. Conclusion

In this paper, we study how to detect creases in the fingerprint images. The detected creases are used to remove spurious minutiae. Experimental results show that a better performance of fingerprint recognition can be achieved by removing spurious minutiae. A fingerprint matching based on creases is also developed in this paper, which can be combined with conventional minutiae matching for real applications. Experimental results show that the performance of the proposed combination algorithm is significantly better than singly using minutiae-based matching.

Acknowledgments

The authors wish to acknowledge support from Natural Science Foundation of China, Natural Science Foundation of Beijing, National 863 Hi-Tech Development Program of China and Basic Research Foundation of Tsinghua University.

References

[1] D. Maltoni, D. Maio, A.K. Jain, S. Probhaker, in: Handbook of Fingerprint Recognition, Springer, New York, 2003.
[2] A.K. Jain, R. Bolle, S. Pankanti (Eds.), BIOMETRICS: Personal Identification in Networked Society, Kluwer, New York, 1999.
[3] D. Zhang, in: Automated Biometrics: Technologies and Systems, Kluwer, New York, 2000.
[4] K. Hrechak, J. McHugh, Automated fingerprint recognition using structural matching, Pattern Recognition 23 (2002) 893–904.
[5] A. Jain, L. Hong, On-line fingerprint verification, IEEE Trans. Pattern Anal. Mach. Intell. 19 (1997) 302–314.

[6] R. Germain, A. Califano, S. Colville, Fingerprint matching using transformation parameter clustering, Comput. Sci. Eng. 4 (4) (1997) 42–49.
[7] C. Wu, J. Zhou, Z. Bian, G. Rong, Robust crease detection in fingerprint images, CVPR (2) (2003) 505–512.
[8] J. Zhou, C. Wu, Z. Bian, D. Zhang, Improving fingerprint recognition based on crease detection, ICBA (2004) 287–293.
[9] M.D. Srinath, P.K. Rajasekaran (Eds.), An Introduction to Statistical Signal Processing with Applications, Wiley, New York, 1979.
[10] S. Sarkar, K. Boyer, Optimal infinite impulse response zero crossing based edge detectors, CVGIP: Image Understanding 54 (2) (1991) 224–243.
[11] S.D. Ma, B. Li, Derivative computation by multiscale filters, Image Vision Comput. 16 (1) (1998) 43–53.
[12] A.K. Jain, L. Hong, S. Pankanti, R. Bolle, An identity authentication system using fingerprints, Proc. IEEE 85 (1997) 1365–1388.
[13] D. Maio, D. Maltoni, R. Cappelli, J. Wayman, A. Jain, FVC2000: fingerprint verification competition, IEEE Trans. Pattern Anal. Mach. Intell. 24 (3) (2002) 402–412.
[14] G. Borgefors, Hierarchical chamfer matching: a parametric edge matching algorithm, IEEE Trans. Pattern Anal. Mach. Intell. 10 (6) (1988) 849–865.
[15] H. Barrow, J. Tenenbaum, R. Bolles, H. Wolf, Parametric correspondence and chamfer matching: two new techniques for image matching, in: Proceedings of the 5th International Joint Conference on Artificial Intelligence, 1977, pp. 659–663.
[16] G. Stockman, S. Kopstein, S. Benett, Matching images to models for registration and object detection via clustering, IEEE Trans. Pattern Anal. Mach. Intell. 4 (1982) 229–241.
[17] N.K. Ratha, K. Karu, S. Chen, A. Jain, A real-time matching system for large fingerprint database, IEEE Trans. Pattern Anal. Mach. Intell. 18 (1996) 799–813.
[18] T.Y. Zhang, C.Y. Suen, A fast parallel algorithm for thinning digital patterns, Commun. ACM 27 (3) (1984) 236–239.
[19] C.J. Ammann, A.G. Sartori-Angus, Fast thinning algorithm for binary images, Image Vision Comput. 3 (2) (1985) 71–79.
[20] G. Borgefors, Improved version of the chamfer matching algorithm, in: NTIS, Springfield, VA, USA, 1983, p. 37.
[21] G. Borgefors, An improved version of the chamfer matching algorithm, in: 7th International Conference on Pattern Recognition, 1984, pp. 1175–1177.
[22] A.K. Jain, S. Prabhakar, L. Hong, Filterbank-based fingerprint matching, IEEE Trans. Image Process. 9 (2000) 846–859.
[23] A.K. Jain, S. Prabhakar, S. Chen, Combining multiple matchers for a high security fingerprint verification system, Pattern Recognition Lett. 20 (1999) 1371–1379.
[24] S. Prabhakar, A.K. Jain, Decision-level fusion in fingerprint verification, Pattern Recognition 35 (2002) 861–874.
[25] C. Yang, J. Zhou, A comparative study of combining multiple enrolled samples for fingerprint verification, Pattern Recognition 39 (11) (2006) 2115–2130.
[26] J. Gu, J. Zhou, C. Yang, Fingerprint recognition by combining global structure and local cues, IEEE Trans. Image Process. 15 (7) (2006) 1952–1964.

About the Author—JIE ZHOU was born in 1968. He received the B.S. and M.S. degrees from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the Ph.D. degree from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology (HUST), Wuhan, China, in 1995.

From 1995 to 1997, he was a Postdoctoral Fellow with the Department of Automation, Tsinghua University, Beijing, China. Currently, he is a Full Professor with the Department of Automation, Tsinghua University. His research area includes pattern recognition, image processing, computer vision, and information fusion. Recently, he has authored more than 10 papers in international journals and more than 40 papers in international conferences. He is an associate editor for the International Journal of Robotics and Automation.

Dr. Zhou received the Best Doctoral Thesis Award from HUST, the First Class Science and Technology Progress Award from the Ministry of Education, China, and the Excellent Young Faculty Award from Tsinghua University in 1995, 1998, and 2003, respectively.

About the Author—FANGLIN CHEN was born in 1983. He received the B.S. degree from the Department of Automation, Xi'an Jiaotong University, Xi'an, China, in 2006. Now he is currently pursuing the Ph.D. degree in the Department of Automation, Tsinghua University, Beijing, China.

His research interests are in pattern recognition, machine learning, image processing, and computer vision.