Fingerprint Feature Enhancement using Block-Direction on Reconstructed Images

Shohreh Kasaei, Mohamed Deriche, and Boualem Boashash

Signal Processing Research Centre (SPRC), QUT
GPO Box 2434, Brisbane Q 4001, Australia
E-mail: skasaei@sharif.edu

Abstract

Fingerprints have been used as unique identifiers of individuals for a very long time. The identification of fingerprint (FP) images is based on matching the features of a query FP, against those stored in a database. As FP databases are characterized by their large size and may contain noisy and distorted query images, an efficient and robust representation of FP images is essential for a reliable identification.

Assuming FPs to be sample images from non-stationary processes of textured images of flow patterns, we propose here a new technique for preprocessing FP images for the purpose of identification. In the proposed algorithm, enhancement as well as ridge extraction processes are based on local dominant ridge directions. The obtained thinned image is then smoothed using morphological operations to detect FP structural features. The proposed algorithm results in an efficient, robust and fast representation of FPs, which accurately retains the fidelity in minutiae (ridge endings and bifurcations).

1 Introduction

In recent years, many efforts have been made in the research of personal verification methods using human biometric features; such as fingerprint, hand geometry, retinal pattern of eyes, face, and voice. Fingerprint matching is the most popular and reliable biometric technique used in automatic personal identification. Law enforcement agencies use FPs routinely for criminal identification. FPs are also being used in several other applications; such as access control for high security installations, credit card usage verification and employee identification. The main reason for the popularity of FPs, as a method of identification, is that each FP of a person is unique and features used in matching remain invariant with age [1], [2].

In an Automatic Fingerprint Identification System (AFIS), a suitable representation of FPs is essential. This representation should retain the discriminating power (to keep the uniqueness of FPs), be easily computable, amenable to automated matching algorithms, stable and invariant to noise and distortions, and finally efficient and compact [1]. Since the raw digital FP image does not meet these requirements, structural features are extracted from FP image for the purpose of representation and matching.

There are two types of features to be extracted from a FP image: high-level or global features, and low-level or local features. The high-level features characterize the FP "pattern class", and low-level features characterize an individual FP image (by ridge characteristics or minutiae).

The important high-level features are the core and delta points (also referred to as singular points). The core point is the top most point on the inner most ridges and a delta point of the triradial point with three ridges radiating from it. These points are highly stable and also rotation and scale invariant. Further, the distance between these singular points, in terms of number of ridges between them, is used to reduce the search space in the database as well as determining the ridge density [3].

Based on the number and locations of these singular points, FPs are classified into five main patterns: arch, tented arch, left loop, right loop, and whorl. These are listed in Table 1.
FP matching is usually carried out at two different levels. At the coarse level, the pattern of the FP is classified [4]. At the fine level, matching is performed by extracting its minutiae.

Eighteen different types of (low-level) FP features have been introduced in the literature [5]. However, for automatic feature extraction and matching, the set of FP

Pattern Class	Core Points	Delta Points
Arch	0	0
Tented Arch	1	1 (middle)
Left Loop	1	1 (right)
Right Loop	1	1 (left)
Whorl	2	2

Table 1: Fingerprint pattern classes and the number of core and delta points within each class.

features is restricted to two types of minutiae: ridge endings and ridge bifurcations. These are shown in Fig. 1. More complex FP features can be expressed as a combination of these two basic features. Two of them are shown in Fig. 2.

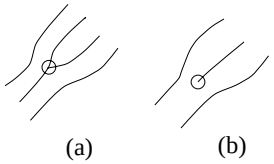

Figure 1: Two commonly used fingerprint features: (a) Ridge bifurcation; (b) Ridge ending.

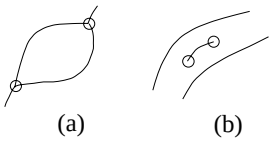

Figure 2: Complex features as combinations of simple features: (a) Enclosure; (b) Short ridge.

Because of the large size of FP databases and noisy FPs encountered in practice, the commercial AFISs usually provide a ranked list of the most possible matches (usually the top 10 matches), which are then verified by human experts. Fig. 3, shows the schematic block diagram of an AFIS. Details of several commercial FP recognition systems (NEC, PRINTRAK, and MORPHO) are described in [6].

To make a positive identification we generally require 8 to 17 coincidental characteristics [7]. With the minutiae representation of FPs, matching a query FP against those stored in a database reduces to the problem of point matching.

Given that query FP images are usually of poor quality and the FP database is very large, a powerful enhancement process is essential for identification purposes. We

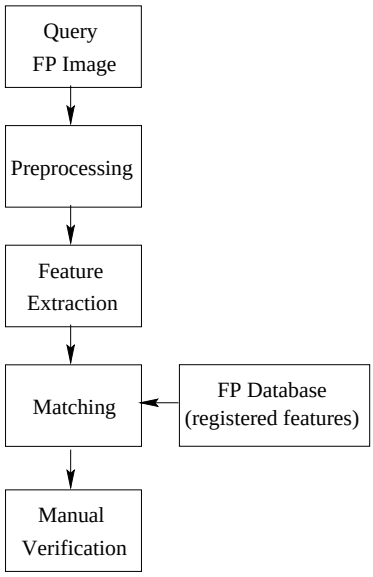

Figure 3: Schematic block diagram of an Automatic Fingerprint Identification System (AFIS).

view FP patterns as narrow ridges separated by narrow valleys. These ridges and valleys alternate and flow in a local dominant direction. This pattern may be corrupted by various kinds of noise causing breaks in ridges, bridges between ridges, and overall gray-scale intensity variation. Despite this noise, humans can often analyze the images easily using such visual clues as local ridge orientation, ridge continuity, and small ridge curvature [8].

The goal of this work is to use some of the assumptions about FP patterns to design effective enhancement and Ridge Extraction (RE) algorithms. As a result, using the local ridge directions the proposed RE algorithm can properly maintain natural shape of gray-level ridges and precise locations of minutiae, as opposed to the RE algorithm proposed in [2]. Furthermore, the proposed RE algorithm is fast and operates only on segmented foreground regions, as opposed to adaptive floating average thresholding process in [9].

2 Proposed FP Structural Feature Extraction Algorithm

The increasing number of FPs, in the form of inked impression on paper cards, collected today by law enforcement agencies has created an enormous problem in storage, transmission, and automated analysis. The authors have introduced an efficient and robust compression technique for FP images which efficiently retains the fine details of ridge information (needed for identifi-

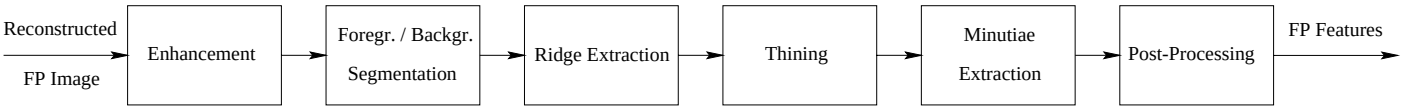

Figure 4: Schematic block diagram of proposed fingerprint structural feature extraction.

cation) in [10] and [11]. In this work, the reconstructed image from this wavelet-based compression algorithm, is used as an input image. Fig. 4 shows a schematic block diagram of the proposed FP structural feature extraction algorithm.

Given that most of the low-energy high-frequency content of the reconstructed images is discarded in compression/decompression process, we can assume that the noise in input images is already reduced considerably. As seen from Fig. 4, the input image is first enhanced and the foreground/background regions are segmented. The ridges are extracted from gray-level foreground region then thinned to have one pixel width. Finally, the minutiae are extracted and false minutiae are eliminated in the post-processing step. These stages are explained in detail in the rest of the paper.

2.1 Enhancement of FP Images

At the first stage, the image is standardized and enhanced using directional filtering. Each 16×16 block is filtered using a library of filters based on its Dominant Ridge Direction (DRD) [8]. The DRD is given as:

$$\theta_b = \frac{90}{\pi} \tan^{-1} \left\{ \frac{\Sigma_{m=1}^{16} \Sigma_{n=1}^{16} 2G_x(m,n)G_y(m,n)}{\Sigma_{m=1}^{16} \Sigma_{n=1}^{16} \left[G_x(m,n)^2 - G_y(m,n)^2\right]} \right\}, \quad (1)$$

where G_x and G_y are the gradient magnitudes obtained using 3×3 Sobel masks. To get the correct θ_b between 0 to 180 degrees, we first check the signs of the numerator and denominator. If denominator is positive we add 90 to θ_b, if denominator is negative and numerator is positive we add 180 to θ_b, and otherwise θ_b is unchanged. The obtained DRDs are then quantized to one of 16 possible directions.

The DRDs are used to form the block-direction image; in which each 16×16 block contains a white line representing the corresponding DRD. The core and delta points can now be detected from block-direction image, since they have maximum direction variability in their neighborhood.

Since DRDs are crucial for the following stages, we further enhance them. In this process, DRDs with rapid changes, compared to their near neighboring directions, should be smoothed while the algorithm should maintain the rapid changes around the core and delta points, to keep their natural shape. The proposed DRD enhancement process is as follows:

1. Form a matrix with each of its elements representing the DRD of a 16×16 blocks.
2. Form 3×3 sliding blocks of this matrix, and for each of them:
3. Compute the standard deviation σ_b.
   (a) If σ_b exceeds a threshold σ_t (e.g. $\sigma_t = 25$), discard the middle point and compute the new standard deviation $\tilde{\sigma}_b$.
   (b) If $\sigma_b > \sigma_t$ and $|\sigma_b - \tilde{\sigma}_b| < threshold$ (e.g. 0.9), the high variance is not because of the middle point and the block is near the core or delta point then do not change the center element, otherwise;
      i. Compute the number of repeats within the block.
      ii. If there is no tie, change the center element with the most repeated element (majority), otherwise;
      iii. Find the elements with the minimum absolute distance with the median of the block. If there is no tie, change the center element with this element, otherwise;
      iv. Find the element with the minimum absolute distance with the mean of the block and change the center element with this element.

Fig. 5, shows the obtained block-direction images before and after this smoothing process. As seen from Fig. 5, the image has been smoothed while the natural shape of core and delta areas are kept efficiently.
Having the smoothed block-direction image, the FP image can now be enhanced using directional filtering. In this work, first each 16×16 block is extended (in each side by 3), to avoid artificial grid lines caused by convolution with a rotated filter. Then the horizontally oriented filter is rotated according to the related DRD and convolved with that block of FP image. In this work we have used the 3×3 Gaussian filter and the O'Gorman 7×7 filter of the "3513" case [8], respectively.

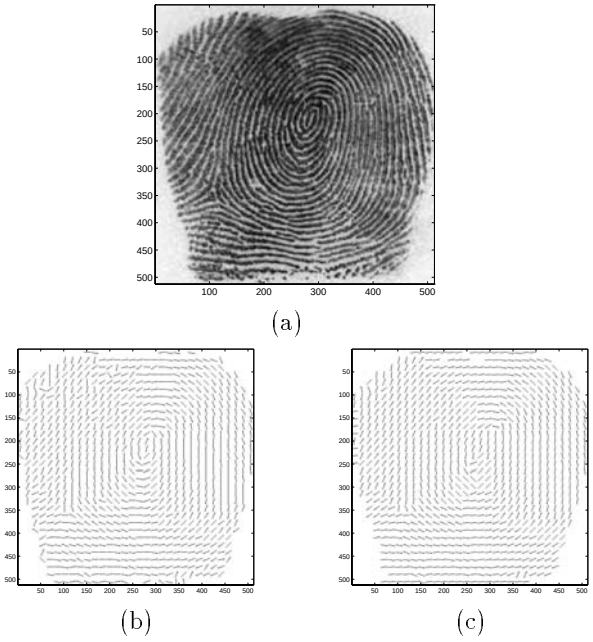

Figure 5: (a) Reconstructed input image; (b) Block-direction image (each line represents dominant ridge direction at each block); (c) Smoothed block-direction image.

2.2 Foreground/Background Segmentation

In FP images, we can view a clear FP ridge area as the foreground and any other area such as smudged regions and noisy regions as the background. The foreground/background segmentation saves processing time and leads to a more precise ridge extraction process. The segmentation is based on the assumption that in a given block, noisy regions have no dominant direction while clear regions flow in a particular direction [9]. Consequently, foreground regions exhibit a very high variance in a direction orthogonal to the orientation of the pattern, θ_o, and a very low variance along its DRD, θ_d.

To segment the foreground region, we propose to obtain the sum of variances in either of the mentioned orientations (θ_o and θ_d), if the difference between these obtained values exceeds some threshold (e.g., 0.35), the region is assigned to foreground. We also noticed that after standardizing the input image this difference gets larger which leads to a more precise segmentation process. Using morphological operations, the obtained mask is then enhanced where the isolated and spur blocks are ignored.

2.3 Ridge Extraction

Ridge extraction is the process by which the foreground area of the gray-level image is mapped into a black/white image. The noise caused by over-inked/under-inked areas as well as skin elasticity introduce a major problem in this process. In fact this mapping is the major part of preprocessing by which the correctness and efficiency of the extracted minutiae is determined. In this work, we propose an efficient, robust and fast RE algorithm based on the obtained block-direction image. Fig. 6 illustrates the proposed RE process.

Each block is first extended to consider the energy of ridges near block endings to avoid block artifacts. After rotation along the Y-axis (using its DRD), the projected waveform on the X-axis is computed, where the minima of this waveform indicates the centers of dominant ridges. Let one of the minima be at the k-th column. In the proposed RE algorithm, if the energy of pixels in $k-1$, k, and $k+1$ columns is less than the *mean* + *standard deviation* of that block, then these pixels are assigned as ridge points. Furthermore, to better retain the natural shape of the ridges, if the energy of pixels in $k-2$ and $k+2$ columns is less than the *mean* of that block, then these pixels are also assigned as ridge points. The mapped ridges are then rotated back to the original direction and the extended area is discarded.

2.4 Minutiae Extraction

The obtained binary ridge image needs further processing before the minutiae can be extracted. First, extracted ridges are thinned to have one pixel width (using morphological filtering), where the continuity in ridges should be maintained. The thinned image is then enhanced by removing the isolated and spur pixels.

The minutiae are then extracted from this thinned image, using the Crossing Number (CN) at a point P which is expressed as [9]:

$$CN = 0.5 \sum_{i=1}^{8} |P_i - P_{i+1}|, \qquad P_9 = P_1, \qquad (2)$$

where P_i is the pixel value (zero or one) in a 3×3 neighborhood of P, as shown in Fig. 2.
The CN has the following characteristics:
Each minutia has three attributes, the x-coordinate, the y-coordinate, and the local ridge direction θ. For bifurcation points the local ridge direction θ is defined as the direction of an imaginary line located in the middle of that bifurcation, as shown in Fig. 7.

P_4	P_3	P_2
P_5	P	P_1
P_6	P_7	P_8

Table 2: A 3×3 mask.

CN	Characteristic
0	Isolated Point
1	End Point
2	Continuing Point
3	Bifurcation Point
4	Crossing Point

Table 3: Characteristics of Crossing Number (CN).

Bifurcations and end points within a margin of 32 pixels of FP image borders are extracted. The false minutiae are then filtered in the post processing stage based on their structural characteristics and spatial relationships of minutiae.

3 Discussion

Due to the usage of DRD information of each FP image and also considering a wide area around each ridge center, the proposed algorithm results in an efficient representation of FP images. The proposed RE algorithm, properly maintains natural shape of gray-level ridges and precise locations of minutiae, as opposed to RE algorithm in [2]. Furthermore, the proposed RE algorithm is fast and operates only on foreground regions, as opposed to adaptive floating average thresholding process in [9]. The proposed algorithm has been used on a variety of FP images with very satisfactory results.

References

[1] N. K. Ratha et al. A Real–Time Matching System for Large Fingerprint Databases. IEEE Trans. on Pattern Analysis and Machine Intelligence, 18(8):799–813, August 1996.

[2] N. K. Ratha, S. Chen, and A. K. Jain. Adaptive Flow Orientation–Based Feature Extraction in Fingerprint Images. Pattern Recognition, 28(11):1657–1672, Nov. 1995.

[3] V. S. Srinivasan and N. N. Murthy. Detection of Singular Points in Fingerprint Images. Pattern Recognition, 25(2):139–153, 1992.

[4] K. Karu and A. K. Jain. Fingerprint Classification. Pattern Recognition, 29(3):389–404, March 1996.

[5] Federal Bureau of Investigation (FBI). The Science of Fingerprints: Classification and Uses. U.S. Government Printing Office, Washington, D.C., 1984.

[6] H. C. Lee and R. E. Gaensslen. Advances in Fingerprint Technology. Elsevier, New York, 1991.

[7] C. A. Gunawardena and V. K. Sagar. Fingerprint Verification using Coincident Sequencing and Thinning. Proceedings IECON '91. 1991 International Conf. on Industrial Electronics, Control and Instrumentation, 3:1917–1922, 1991.

[8] L. O'Gorman and J. V. Nickerson. An Approach to Fingerprint Filter Design. Pattern Recognition, 22(1):29–38, 1989.

[9] B. M. Mehtre. Fingerprint Image Analysis for Automatic Identification. Machine Vision and App., 6:124–139, 1993.

[10] S. Kasaei, M. Deriche, and B. Boashash. Fingerprint Compression Using Wavelet Packet Transform and Pyramid Lattice Vector Quantization. IEICE Special Section on Digital Signal Processing, E80-A(8):1446–1452, August 1997.

[11] S. Kasaei and M. Deriche. Fingerprint Compression Using a Piecewise–Uniform Pyramid Lattice Vector Quantization. In IEEE Conf. on Acoustics, Speech and Signal Processing, ICASSP, pages 3117–3120, Munich, Germany, April 1997.

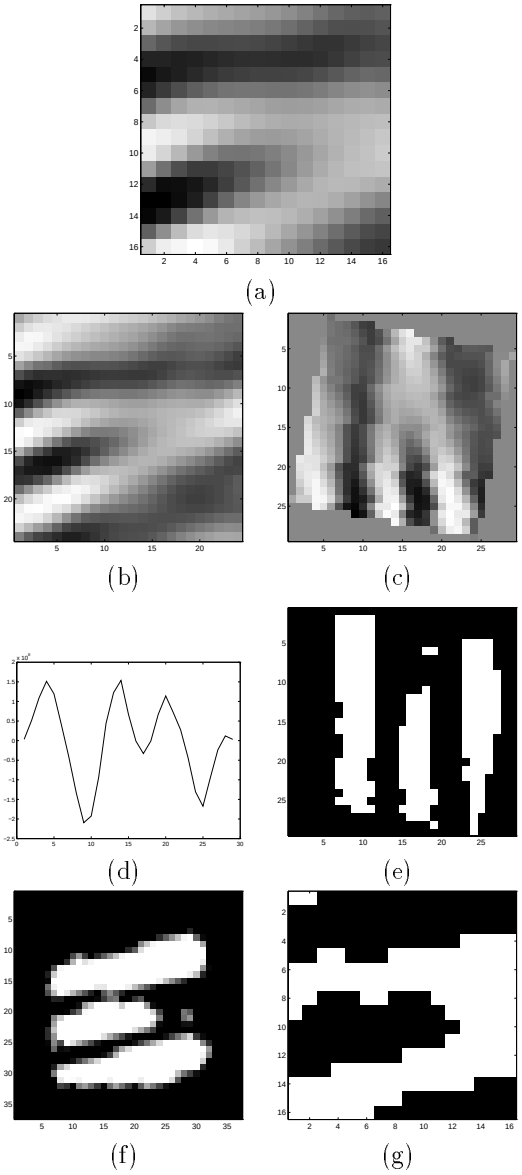

Figure 6: Ridge extraction stage: (a) Input block; (b) Extension of (a) to 24×24; (c) Rotation along Y-axis; (d) Projected waveform on X-axis; (e) Ridge segmentation; (f) Rotation along original orientation; (g) Extracted ridges.

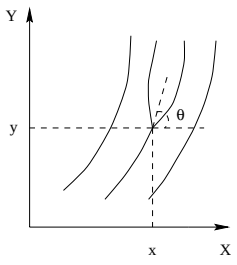

Figure 7: Components of a bifurcation point.