

Chapter 14

GRAPHICAL REPRESENTATION OF FINGERPRINT IMAGES

Jie Zhou¹, David Zhang², Jinwei Gu¹, and Nannan Wu¹

1. Department of Automation, Tsinghua University, Beijing 100084, China

2. Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong

jzhou@tsinghua.edu.cn; csdzhang@comp.polyu.edu.hk; gujinwei98@mails.tsinghua.edu.cn; wunannan99@mails.tsinghua.edu.cn

Abstract Fingerprint recognition is very important in automatic personal identification. In conventional fingerprint recognition algorithms, fingerprints are represented graphically with a simple set of minutiae and singular points, but the information in this kind of representation is not adequate for large-scale applications or where fingerprint quality is poor. A complete and compact representation of fingerprints is thus highly desirable. In this chapter, we focus on research issues in the graphical representation of fingerprints. We first introduce minutiae-based representation and provide some models for the graphical representation of orientation fields. Latterly, we deal with the generation of synthetic fingerprint images and conclude with a discussion of how to establish a complete fingerprint representation.

Keywords: Fingerprint recognition, Orientation field, Minutiae, Synthesis, Graphical model, Approximation

14.1. Introduction

In recent years, fingerprint identification has received increasing attention. Among the various biometric techniques used for automatic personal identification, automatic fingerprint identification systems are the most popular and reliable. Nonetheless, while the performance of fingerprint identification systems has reached a high level, they are still not satisfactory when applied to large databases or to poor-quality fingerprints [1,2].

A fingerprint is the pattern of ridges and valleys on the surface of a fingertip. Figure 14.1 (a) depicts a fingerprint in which the ridges are black and the valleys are white. Its *orientation field*, defined as the local orientation of the ridge-valley structures, is shown in Figure 14.1 (b). The *minutiae*, ridge endings and bifurcations, and the *singular points* are also shown in Figure 14.1 (a). Singular points can be viewed as points where the orientation field is discontinuous. They can be classified into two types: a *core*, the point of the innermost curving ridges, and a *delta*, the center of triangular regions where three different directional flows meet. Fingerprints are usually partitioned into six main classes according to their macro-singularities, i.e., arch, tented arch, left loop, right loop, twin loop and whorl (see Figure 14.2). Fingerprints can be represented by a large number of features, including the overall ridge flow pattern (i.e. orientation field), ridge frequency (i.e. density map), location and position of singular points, and the type, direction and location of minutiae points. All of these features contribute to fingerprint individuality.

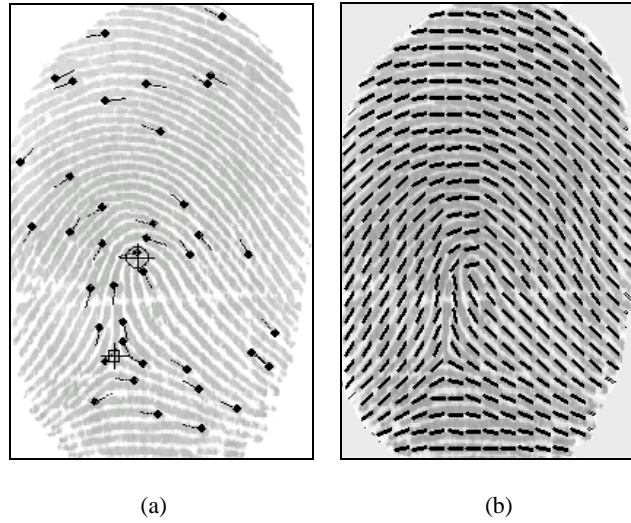


Figure 14.1. Example of a fingerprint: (a) singular points and minutiae with its direction, (b) orientation field shown with unit vector.

The performance of a fingerprint identification system depends mainly on the kind of fingerprint representation it utilizes. Most classical fingerprint recognition algorithms [1,2,3] represent the fingerprint using the minutiae and the singular points, including their coordinates and direction, as the distinctive features. That means fingerprints are first graphically represented with a set of minutiae and singular points are then compared with the template set. If the matching score

exceeds a predefined threshold, two fingerprints can be regarded as belonging to the same finger.

Obviously this kind of representation does not make use of every feature that is available in fingerprints and therefore cannot provide enough information for large-scale fingerprint identification tasks [4]. Certainly, a better representation for fingerprints is needed.

In this chapter, we mainly address the topic of graphical representation of fingerprints. First we introduce the conventional representation of fingerprints, the minutiae-based representation. We then establish a complete representation with a compact form in which much more information other than minutiae features (such as the orientation field) can be taken into account.

The remainder of the chapter is organized as follows. Section 14.2 introduces the minutiae-based representation of fingerprints. Section 14.3 provides four models of fingerprint orientation fields. Section 14.4 describes how to synthesize the fingerprint images utilizing the orientation field models. Section 14.5 discusses ways to establish a complete and compact fingerprint representation. The last section draws some conclusions.

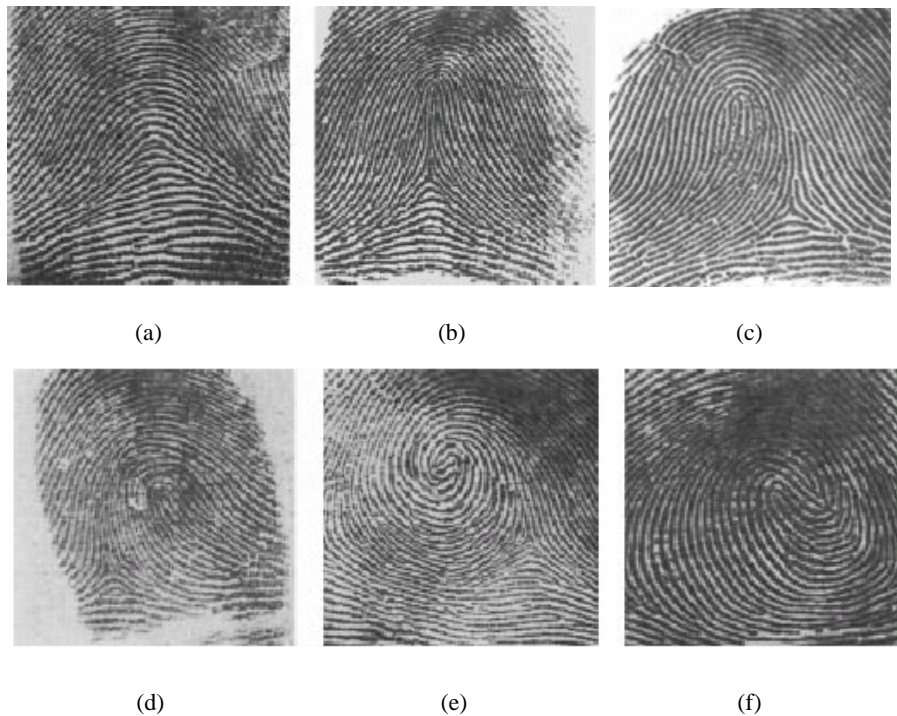


Figure 14.2. One fingerprint for each of six main classes: (a) arch, (b) tented arch, (c) left loop, (d) right loop, (e) whorl, and (f) twin loop.

14.2. Minutiae-based Representation Fingerprint

Most conventional fingerprint recognition algorithms are based on a minutiae-based representation, which has a rather low storage cost. A reliable minutiae extraction method mainly includes the following steps: orientation field estimation, ridge extraction or enhancement, ridge thinning and minutiae extraction. Figure 14.3 provides a flowchart of conventional minutiae extraction and minutiae matching algorithms.

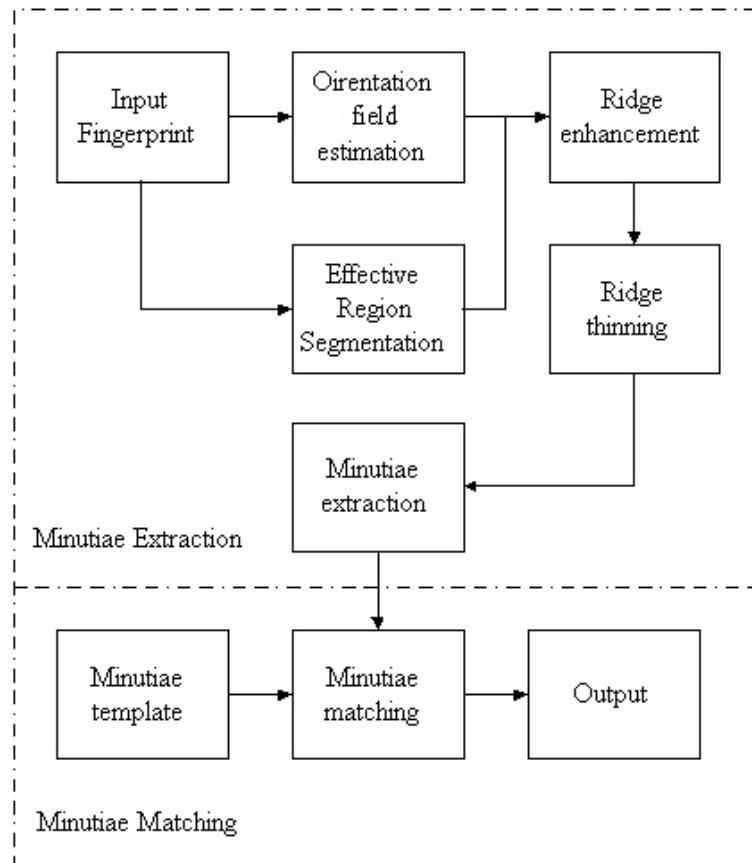


Figure 14.3. Flowchart of minutiae extraction and matching.

By quantifying the amount of information available in minutiae-based representation, a correspondence can be established between two fingerprints, yet between the minutiae-based representations of two arbitrarily chosen fingerprints belonging to different fingers it is also possible to establish a false correspondence.

For example, the probability that a fingerprint with 36 minutiae points will share 12 minutiae points with another arbitrarily chosen fingerprint with 36 minutiae points is 6.10×10^{-8} [3]. Figure 14.4 provides an example of such a false correspondence. Because of noise in the sensed fingerprint images, errors in locating minutiae, and the fragility of the matching algorithms, the observed matching performance of the state-in-the-art fingerprint recognition systems is several orders of magnitude lower than their theoretical performance. Because minutiae-based representations use only a part of the discriminatory information present in fingerprints, it may be desirable for the purposes of automatic matching to explore additional complementary fingerprint representations. Taking Figure 14.4 as an example, the orientation fields of these two fingerprints are clearly different in some places, such as at the bottom of the fingerprints and the left-down part from the core. Including orientation information in the matching step would greatly reduce opportunities for this kind of false match.

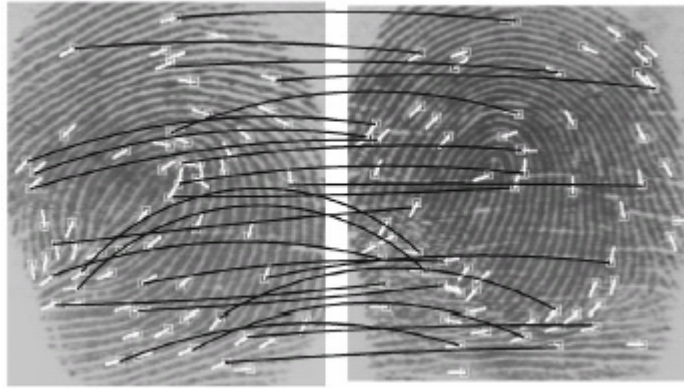


Figure 14.4. A false match between two different fingerprints: 64 minutiae are detected in the left image, 65 in the right image, with 25 “false” correspondences [3].

14.3. Modeling Orientation Fields

As a global feature, an orientation field describes one of the basic structures of a fingerprint and is thus quite important in the modelling and representation of the entire fingerprint. Orientation field variation is low frequency so it is robust with respect to various noises. It has been widely used for minutiae extraction and fingerprint classification. In this section, we focus on the modelling of the orientation field. Our purpose is to represent the orientation field in a complete and compact form so that it can be accurately reconstructed with several coefficients.

This work is significant in three ways. (1) It can be used to improve the estimation of orientation field, especially when fingerprint quality is poor; therefore it will be of benefit in the extraction of minutiae for conventional fingerprint identification algorithms. (2) The coefficients of the orientation field model can be saved for use in the matching step. As a result, information on the orientation field can be utilized for fingerprint identification. By combining it with the minutiae information, we can expect a much better identification performance. (3) We can synthesize the fingerprint by using information on the orientation field, minutiae and the density between the ridges. This makes it possible to establish a complete representation for the fingerprint by combining the orientation model with other information.

1. Zero-Pole Model

Sherlock and Monro [5] proposed a so-called zero-pole model for the orientation field based on singular points, which takes the core as zero and the delta as a pole in the complex plane. The influence of a core, z_c , is $\frac{1}{2}\arg(z-z_c)$ for the point, z , and that of a delta z_d , is $-\frac{1}{2}\arg(z-z_d)$. The orientation at z , is the sum of the influence of all cores and deltas, i.e.

$$\theta(z) = \frac{1}{2} \left[\sum_i \arg(z - z_c^i) - \sum_j \arg(z - z_d^j) \right] \quad (14.1)$$

where z_c^i and z_d^j are the i -th core and j -th delta. Figure 14.5 depicts the influence of a core and delta. They can roughly describe the structure near these singular points.

2. Piecewise Linear Approximation Model

Vizcaya and Gerhardt [6] had made an improvement using a piecewise linear approximation model around singular points to adjust the behaviour of the zero and pole. First, the neighbourhood of each singular point is uniformly divided into eight regions and the influence of the singular point is assumed to change linearly in each region. An optimization implemented by gradient-descent is then performed to get a piecewise linear function.

These two models, the zero-pole model and the piecewise linear model, cannot deal with fingerprints belonging to the plain arch class (i.e. without singular points). Furthermore, they do not take into account the distance from singular points, but as the influence of a singular point is the same as any point on the same central line, whether near or far from the singular point, serious errors arise in the modelling of those regions that are far from singular points. As a result, these two

models cannot be used to accurately approximate a real fingerprint's orientation field.

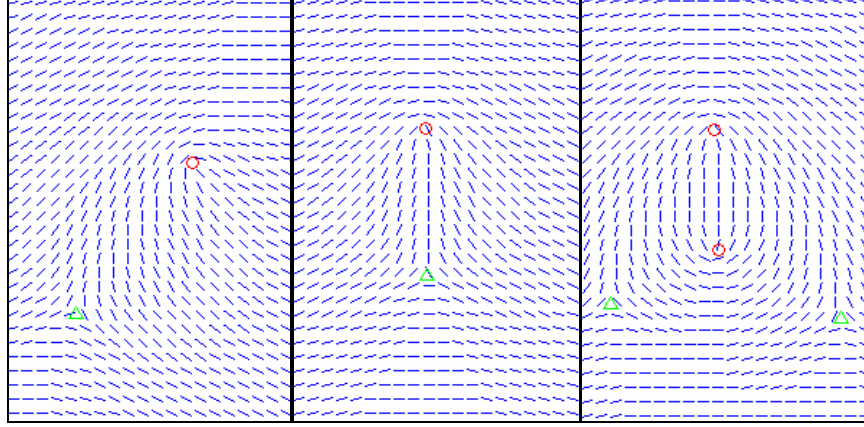


Figure 14.5. Illustration of zero-pole model.

3. Rational Complex Model [7]

Denote the image plane as a complex space, C . For any $z \in C$, the value of a fingerprint orientation, $\theta(z)$, is defined within $[0, \pi)$, so it can be regarded as

half the argument of a complex number, i.e. $\theta(z) = \frac{1}{2} \arg U(z)$. As we know,

the orientation pattern of a fingerprint is quite smooth and continuous except at the singular points (including cores and deltas), so a rational complex function may be utilized here to represent the function, $U(z)$, in which the known cores and deltas act as zeros of the numerator and the denominator, respectively. Thus, the model for the orientation field can be defined as

$$\phi(z) = \frac{1}{2} \arg \left[\frac{f(z)}{g(z)} \cdot \frac{P(z)}{Q(z)} \right], \quad (14.2)$$

where

$$P(z) = \prod_{i=1}^{s_0} (z - z_c^i), \quad Q(z) = \prod_{j=1}^{s_1} (z - z_d^j), \quad \{z_c^i\}_{1 \leq i \leq s_0} \text{ and } \{z_d^j\}_{1 \leq j \leq s_1}$$

are the cores and deltas of the fingerprint in the known region. The zeros of $f(z)$ and $g(z)$ should be outside the known region. Actually, these zeros are always sparsely located on real fingerprints. All the zeros of $f(z)$, $g(z)$, $P(z)$ and $Q(z)$ define the nature of the model.

Obviously, the zero-pole model proposed in [5] can be seen as a special example of the rational complex model, where $f(z)$ and $g(z)$ are both set to a constant, such as 1. It also should be noted that our model is suitable for all types of fingerprints, even for “plain arch” fingerprints, which do not have singular points.

How to compute the coefficients of the model? From the mathematical theorem of complex variables, we know that a rational function can be approximated using a polynomial function in a closed region. So, to simplify the computation of the rational complex model, the model can be written as

$$\phi(z) = \frac{1}{2} \arg[f(z) \cdot \frac{P(z)}{Q(z)}]. \quad (14.3)$$

We want to find a function, $f(z)$, to minimize the difference between $\{\phi(z)\}$ and the original orientation field, $\{\theta(z)\}$. Denote $\psi(z) = \frac{1}{2} \arg[\frac{P(z)}{Q(z)}]$ and $\omega(z) = \frac{1}{2} \arg[f(z)]$, and now what we need to do is to compute $f(z)$ by minimizing the difference between $\{\omega(z)\}$ and $\{\theta(z) - \psi(z)\}$.

It is unsuitable for us to directly compute this minimum. A solution to this problem is to map the orientation field to a continuous complex function by using

$$U(z) = \cos 2[\theta(z) - \psi(z)] + i \sin 2[\theta(z) - \psi(z)], \quad (14.4)$$

Then, instead of computing the minimal difference between $\{\omega(z)\}$ and $\{\theta(z) - \psi(z)\}$, we compute the function, $f(z)$, by minimizing $\sum_z |f(z) - U(z)|^2$. Since $\theta(z)$, $P(z)$ and $Q(z)$ (corresponding to the original orientation field, cores and deltas) are known when we deal with a fingerprint image, it is easy to solve this problem by using Least Square Error principle.

When we choose $f(z)$ from the set of polynomials of an order less than n , only $n+1$ parameters need to be computed and saved (many fewer than the model in [6]). In the experiments, n is usually set as 6. Due to the global approximation, the rational complex model has a robust performance against noise.

An experiment was carried out on more than 100 inked fingerprints and live-scanned fingerprints. These fingerprints were of different types: loop, whorl, twin loop, and plain arch without singular points. They also varied in different qualities. Three orientation models are evaluated on the database, zero-pole model,

piecewise linear model and rational complex model. All of them used the same algorithm for singular points extraction and orientation estimation algorithms. On all these fingerprint images, the performance of the rational complex model was quite satisfying. The average approximation error of the rational complex model was about 6 degrees, which is much better than using the other two models (respectively 14 degrees and 11 degrees). This shows that the rational complex model's more effectively than the previous models.

Figure 14.6 provides an example for comparison, where (a) is the original fingerprint, and (b), (c), and (d) are the reconstructed orientation fields, respectively using the zero-pole, piecewise linear, and rational complex models. The reconstructed orientation field is shown as unit vectors upon the original fingerprint. As shown, the zero-pole model and piecewise linear model perform badly far from the singular points, easily observable in the top-left and the top-right part in (c) and (d). In contrast, the rational complex model describes the orientation of the whole fingerprint image precisely.

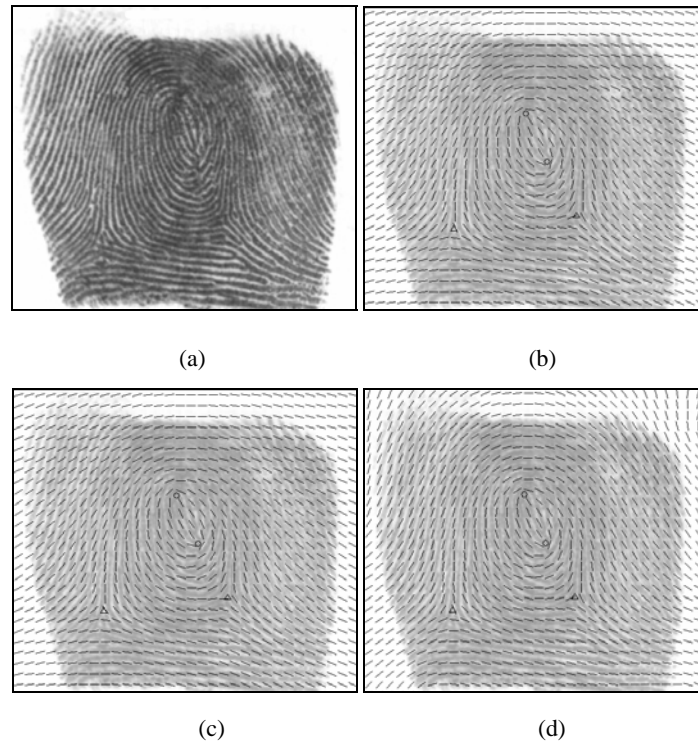


Figure 14.6. Comparative result of the orientation field constructed by using three models: (a) original fingerprint image, (b) zero-pole model, (c) piecewise linear model, and (d) rational complex model.

4. Combination Model [8]

Since the orientation of fingerprints is quite smooth and continuous except at singular points, a polynomial model can be applied to approximate the global orientation field. At each singular point, the local region is described using a point-charge model similar to the zero-pole model. Then, these two models are smoothly combined through a weight function.

Since the value of the orientation of a fingerprint is defined within $[0, \pi]$, it seems that this representation has an intrinsic discontinuity. (In fact, the orientation 0 is the same as the orientation π in a ridge pattern). As a result, we cannot model the orientation field directly. A solution to this problem is to map the orientation field to a continuous complex function. Define $\theta(x,y)$ and $U(x,y)$ as respectively the orientation field and the transformed function. The mapping can be defined as

$$U = R + iI = \cos 2\theta + i \sin 2\theta \quad (14.4)$$

where R and I denote respectively the real and the imaginary parts of the complex function, $U(x,y)$. Obviously, $R(x,y)$ and $I(x,y)$ are continuous with x, y in those regions. The above mapping is a one-to-one transformation and $\theta(x,y)$ can be easily reconstructed from the values of $R(x,y)$ and $I(x,y)$.

To globally represent $R(x,y)$ and $I(x,y)$, two bivariate polynomial models are established, which are denoted by PR, PI respectively. These two polynomials can be formulated as:

$$PR(x, y) = \begin{pmatrix} 1 & x & \cdots & x^n \end{pmatrix} \cdot P_1 \cdot \begin{pmatrix} 1 & y & \cdots & y^n \end{pmatrix}^T, \quad (14.5)$$

and

$$PI(x, y) = \begin{pmatrix} 1 & x & \cdots & x^n \end{pmatrix} \cdot P_2 \cdot \begin{pmatrix} 1 & y & \cdots & y^n \end{pmatrix}^T, \quad (14.6)$$

where n is the polynomials' order and the matrixes, $P_i \in \mathbb{R}^{n \times n}$, $\forall i = 1, 2$.

Near the singular points, the orientation is no longer smooth, so it is difficult to model with a polynomial function. A model named 'Point-Charge' (PC) is added at each singular point. Compared with the zero-pole model, Point-Charge uses different quantities of electricity to describe the neighbourhood of each singular point instead of the same influence at all singular points, while the influence of a certain singular point at the point, (x,y) , varies with the distance between the point and the singular point. The influence of a standard (vertical) core at the point, (x,y) , is defined as

$$PC_{Core} = H_1 + iH_2 = \begin{cases} \frac{y-y_0}{r}Q - i\frac{x-x_0}{r}Q, & r \leq R \\ 0, & r > R \end{cases}, \quad (14.7)$$

where (x_0, y_0) is this core's position, Q is the quantity of electricity, R denotes the radius of its effective region, and $r = \sqrt{(x-x_0)^2 + (y-y_0)^2}$. The radius of a standard delta is:

$$PC_{Delta} = H_1 + iH_2 = \begin{cases} -\frac{y-y_0}{r}Q - i\frac{x-x_0}{r}Q, & r \leq R \\ 0, & r > R \end{cases}. \quad (14.8)$$

In a real fingerprint, the ridge pattern at the singular points may have a rotation angle compared with the standard one. If the rotation angle from standard position is ϕ ($\phi \in (-\pi, \pi]$), a transformation can be made as:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \begin{pmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{pmatrix} \begin{pmatrix} x-x_0 \\ y-y_0 \end{pmatrix}. \quad (14.9)$$

Then, the Point-Charge model can be modified by taking x' and y' instead of x and y , for cores in Eq. (14.7) and deltas in Eq. (14.8), respectively.

To combine the polynomial model (PR , PI) with Point-Charge smoothly, a weight function can be used. For Point-Charge, the weighting factor at the point (x, y) is defined as:

$$\alpha_{PC}^{(k)}(x, y) = 1 - \frac{r^{(k)}(x, y)}{R^{(k)}}, \quad (14.10)$$

where $(x_0^{(k)}, y_0^{(k)})$ is the coordinate of the k -th singular point, $R^{(k)}$ is the radius of the effective region, and $r^{(k)}(x, y)$ is set as $\min(\sqrt{(x-x_0^{(k)})^2 + (y-y_0^{(k)})^2}, R^{(k)})$. For the polynomial model, the weighting factor at the point, (x, y) , is:

$$\alpha_{PM}(x, y) = \max \left\{ 1 - \sum_{k=1}^K \alpha_{PC}^{(k)}, 0 \right\}, \quad (14.11)$$

where K is the number of singular points. The weight function guarantees that for each point, its orientation follows the polynomial model if it is far from the singular points and follows the Point-Charge if it is near one of the singular points.

Then, the combination model for the whole fingerprint's orientation field can be formulated as:

$$\begin{pmatrix} R(x,y) \\ I(x,y) \end{pmatrix} = \alpha_{PM} \cdot \begin{pmatrix} PR \\ PI \end{pmatrix} + \sum_{k=1}^K \alpha_{PC}^{(k)} \cdot \begin{pmatrix} H_1^{(k)} \\ H_2^{(k)} \end{pmatrix}, \quad (14.12)$$

with the constraint of

$$R^2(x,y) + I^2(x,y) = 1, \quad (14.13)$$

where PR and PI are respectively the real and the imaginary parts of the polynomial model, and $H_1^{(k)}$ and $H_2^{(k)}$ are respectively the real and the imaginary parts of the Point-Charge model for the k -th singular point. Obviously, the combination model is continuous with x and y . The coefficient matrices of the two polynomials, PR and PI , and the electrical qualities, $\{Q_1, Q_2, \dots, Q_K\}$, of the singular points will define the combination model.

Obviously, the combination model can be regarded as a generalized method of the other three models introduced above, so it can represent the orientation field more accurately. It does, however, have more parameters than the zero-pole and rational complex models, and this will produce some limitations when it is utilized in synthetic fingerprint generation.

To compute its parameters, the coarse orientation field and singular points need to be estimated from the original fingerprint image. After that, two bivariate polynomials can be computed using the Weighted Least Square (WLS) algorithm. The coefficients of the polynomial are obtained by minimizing the weighted square error between the polynomial and the values of $R(x,y)$ and $I(x,y)$ computed from the real fingerprint. As pointed out above, the reliability, $W(x,y)$, can indicate how well the orientation fits the real ridge. The higher the reliability $W(x,y)$, the more influence the point should have. Then $W(x,y)$ can be used as the weighting factor at the point (x,y) .

After the computation of polynomials, the coefficients of the Point-Charge Model at singular points can be obtained in two steps. First, two parameters are estimated for each singular point: the rotation angle, ϕ , and the effective radius,

R (which can also be chosen in advance). Second, charges of singular points are estimated by optimization.

As we know, a higher order polynomial can provide a better approximation, but at the same time it will result in a much higher cost of storage and computation. Moreover, a high order polynomial will be ill-behaved in numerical approximation. As to a lower order polynomial, however, it will yield lower approximation accuracy in those regions with high curvature. As a trade-off, 4-order ($n=4$) polynomials can be chosen for the global approximation. The experimental results showed that they performed well enough for almost all real fingerprints, while the cost for storage and computation remained low. In Figure 14.7, some of the results are presented, in which the reconstructed orientation fields are shown as unit vectors upon the original fingerprint. As shown, the results are rather accurate and robust for these fingerprints, although there is a lot of noise in these images.

The combination model has many more parameters than the zero-pole and rational complex models, making it, in comparison with those two models, much more difficult to use in synthesizing fingerprint images

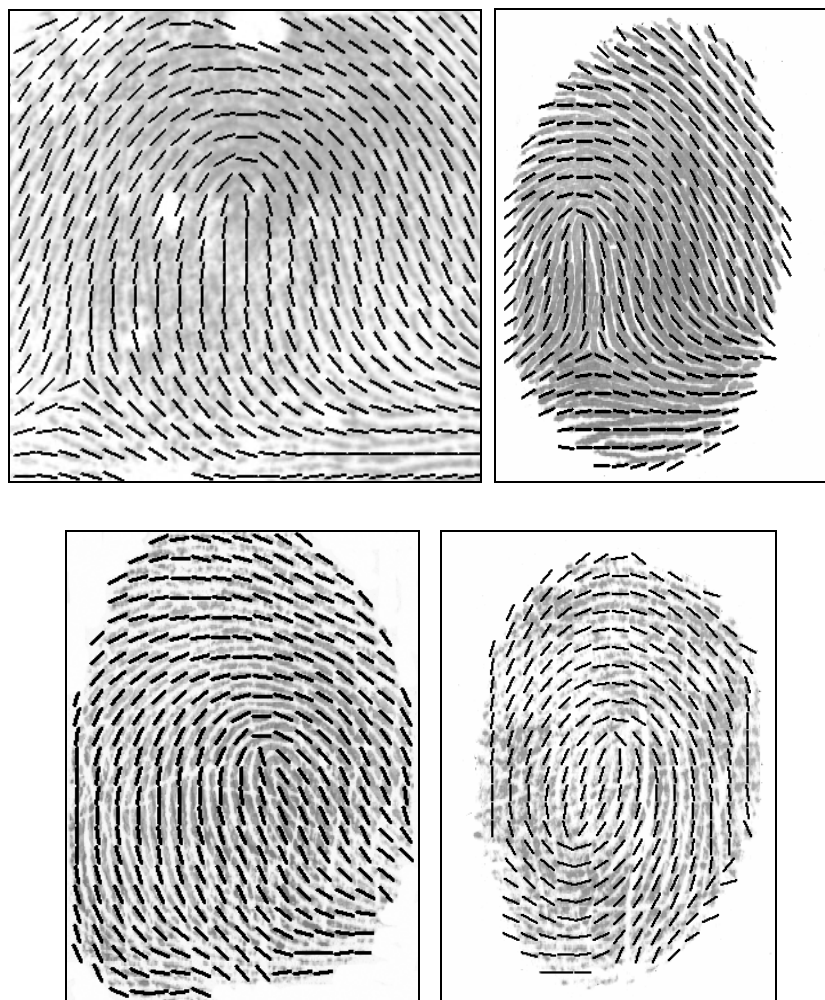


Figure 14.7. Some examples of approximation results for the combination model.

14.4. Generation of Synthetic Fingerprint Images

The main topic of this section is the generation of synthetic fingerprint images. Such images can not only be used to create, at zero cost, large databases of fingerprints, thus allowing recognition algorithms to be simply tested and optimized, they would help us to better understand the rules that underpin the biological process involved in the genesis of fingerprints. The effective modelling of fingerprint patterns could also contribute to the development of very useful tools for the inverse task, i.e. fingerprint feature extraction.

The generation method sequentially performs the following steps [9]: (1) orientation field generation, (2) density map generation, (3) ridge pattern generation and (4) noising and rendering.

Orientation field generation: There are three ways to generate the orientation field for synthetic fingerprints. 1) Using the zero-pole model in [6], a consistent orientation field can be calculated from the predefined position of the cores and deltas alone, (see Figure 14.5) but the generated orientation is less accurate than real fingerprints. 2) The rational complex model in [7] produces a more lifelike result. From Eq. (14.2), we know that the rational complex model is determined by the zeros of $f(z)$, $g(z)$, $P(z)$ and $Q(z)$, in which the zeros of $P(z)$ are cores and those of $Q(z)$ are deltas. The position of cores and deltas can be predefined according to the fingerprint's class, then, several points are randomly and sparsely selected from outside the print region as the zeros of $f(z)$ and $g(z)$. 3) We can first choose several real fingerprints and compute the parameters of the rational complex model by minimizing the approximation error between the model and the fingerprint image' orientation [7]. Then these parameters can be changed a little randomly to produce different orientation fields.

Density map generation [9]: This step creates a density map on the basis of some heuristic criteria inferred from the visual inspection of several real fingerprints. The visual inspection of several fingerprint images, leads us to immediately discard the possibility of generating the density map in a completely random way. In fact, we noted that usually in the region above the northernmost core and in the region below the southernmost delta the ridge-line density is lower than in the rest of the fingerprint. So, the density-map can be generated as follows: 1) randomly select a feasible overall background density; 2) slightly increase the density in the above-described regions according to the singularity locations; 3) randomly perturb the density map and performs a local smoothing.

Ridge pattern generation [9]: In this step, the ridgeline pattern and the minutiae are created through a space-variant linear filtering; the output is a very clear near-binary fingerprint image. Given an orientation field and a density map as input, a deterministic generation of a ridgeline pattern, including consistent minutiae is not an easy task. One could try a priori to fix the number, type and location of the minutiae, and by means of an explicit model, generate the gray-scale fingerprint image starting from the minutiae neighbourhoods and expanding to connect different regions until the whole image is covered. Such a constructive approach requires several complex rules and tricks to be implemented in order to deal with the complexity of fingerprint ridgeline patterns. A more "elegant" approach could be based on the use of a syntactic approach that generates fingerprints according to some starting symbols and a set of production rules. The method here proposed is very simple, but at the same time surprisingly powerful: by iteratively enhancing an initial image (containing one or more isolated spikes) through Gabor-like filters

adjusted according to the local orientation and density, a consistent and very realistic ridge-line pattern “magically” appears; in particular, fingerprint minutiae of different types (terminations, bifurcations, islands, dots, etc.) are automatically generated at random positions. Formally, the filter is obtained as the product of a Gaussian by a cosine plane wave. A correction term is included to make the filter DC free:

$$f(\mathbf{v}) = \frac{1}{\sigma^2} e^{-\frac{\|\mathbf{v}\|^2}{2\sigma^2}} [\cos(\mathbf{k} \cdot \mathbf{v}) - e^{-\frac{\sigma^2 \|\mathbf{k}\|^2}{2}}], \quad (14.14)$$

where σ is the variance of the Gaussian and \mathbf{k} is the wave vector of the plane wave. The parameters σ and \mathbf{k} are adjusted using local ridge orientation and density. Let \mathbf{z} be a point of the image where the filters have to be applied, then the vector $\mathbf{k} = [k_x, k_y]^T$ is determined by the solution of the two equations:

$$D(\mathbf{z}) = (k_x^2 + k_y^2)^{1/2} \quad \text{and} \quad \tan(O(\mathbf{z})) = -\frac{k_x}{k_y}. \quad (14.15)$$

The parameter σ , which determines the bandwidth of the filter, is adjusted in the time domain according to $D(\mathbf{z})$ so that the filter does not contain more than three effective peaks. The filter is then clipped to get a FIR filter. The filter should be designed with the constraint that the maximum possible response is larger than 1. When such a filter is applied repeatedly, the dynamic range of the output increases and becomes numerically unstable, but the generation algorithm exploits this fact. When the output values are clipped to fit into a constant range, it is possible to obtain a near-binary image. The above filter equation satisfies this requirement without any normalization.

In Figure 14.8, an example of the iterative ridgeline generation process is shown.

Noising and rendering [9]: In this step, some specific noise is added and a realistic gray-scale representation of the fingerprint is produced. During fingerprint acquisition several factors contribute to the deterioration of the original signal, thus producing a gray-scale noisy image: 1) irregularity of the ridge and differences in their contact with the sensor surface; 2) the presence of small pores within the ridges; 3) the presence of very-small-prominence ridges; 4) gaps and cluttering noise due to non-uniform pressure of the finger against the sensor or due to excessively wet or dry fingers. So, the noising and rendering approach sequentially performs the following steps: 1) isolate the valley white pixels into a separate layer by copying the pixels brighter than a fixed threshold to a temporary image; 2) add noise in the form of small white blobs of variable size and shape; 3)

smooth the image; 4) superimpose the valley layer to the image obtained. In the above steps, steps 1 and 4 are necessary to avoid an excessive overall image smoothing.

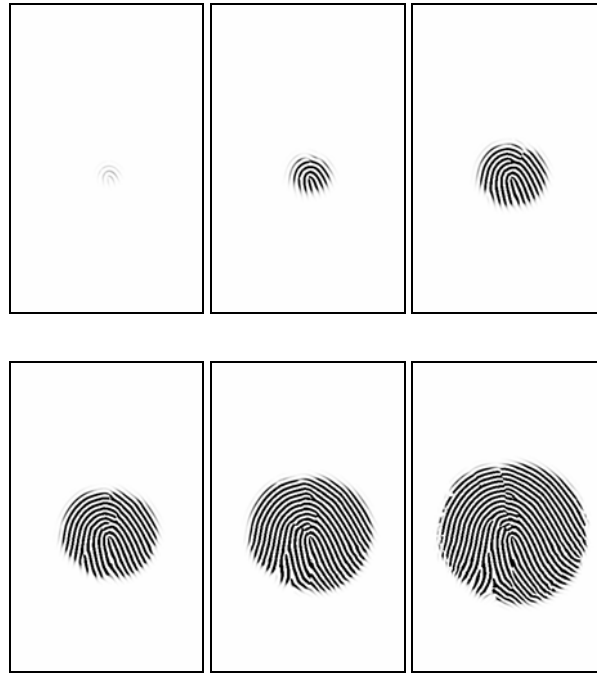


Figure 14.8. An example to illustrate the image-generation process.

14.5. Complete Representation of Fingerprints

Synthesizing a fingerprint using an orientation field and a density map means the orientation field and density map will constitute a complete fingerprint representation. The minutiae primarily originate from the ridgeline disparity produced by local convergence/divergence of the orientation field and by density changes. As stated above, a complete representation of fingerprints will help us to choose an appropriate feature set for the matching. From this it would seem that these two features, an orientation field and a density map, are all enough for the purpose of fingerprint matching. Unfortunately this is not so.

We tested this with two experiments. In one we constructed a fingerprint from the original fingerprint image using the synthesis method. In the other we compared two synthetic fingerprints using different starting points.

In the first experiment, the orientation field and density map should be computed from the original fingerprint image. Many algorithms have been proposed for the computation of the orientation field of a real fingerprint image. Of these, we prefer the algorithm proposed in [10]. To compute the density map, we first take some steps similar to the ridge detection method used in [4], then use a median filter to remove the noise. After that, it is a simple matter to choose a threshold to segment the ridges adaptively. Along the direction normal to the local ridge orientation, the width of the ridge at this position can be measured. After all pixels are processed, the density map of the fingerprint image has been obtained. After obtaining the orientation field and density map, the method described in last section is taken to produce a new image. Unfortunately, the experimental results are not satisfactory and the reconstructed image is not similar to the original. Another experiment is conducted using different starting points and comparing the synthetic fingerprints. The result shows that starting from different points will result in evident changes in the final synthetic image. One example is provided in Figure 14.9.

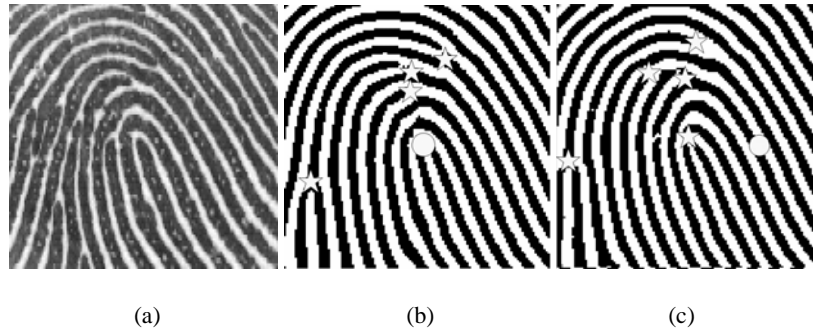


Figure 14.9. An example of reconstruction: (a) a real fingerprint image, (b) and (c) are two reconstructed images, in which the circles denote the starting points and the star symbols denote the minutiae points. The iterative times is 25.

These experiments are not conclusive but they do raise some questions and issues: (1) The proposed synthesis algorithm is an iterative method but is the iterative process convergent? If convergent, is the converged image the same as the original noise-affected fingerprint? (2) From the point of view of numerical analysis, the computational error will greatly influence the final result, so the main cause of the experiment's failure may be inaccurate computation of the orientation field and density map. It also implies that a representation method that uses only an orientation field and a density map is not adequate for recognition tasks. One simple way to reduce the computational errors is to use more constraints, i.e., more features in our task, such as minutiae points.

14.6. Summary

Fingerprint recognition applications seek a complete but compact representation of fingerprints. In this chapter, we introduced issues related to the graphical representation of fingerprints and some approaches to these issues. Since orientation field is an important feature in the description of the global appearance of fingerprints, we have focused on orientation field models and have described methods based on them for generating synthetic fingerprint images. Finally, we discussed the reconstruction of a noise-less fingerprint image from an original fingerprint image. We summarise our conclusions as follows:

- Conventional fingerprint recognition algorithms rely on minutiae-based representations of fingerprints. As a minutiae-based representation uses only a part of the discriminatory information present in fingerprints, further exploration of additional complementary representations of fingerprints for automatic matching is needed.
- An orientation field can be well represented by either a rational complex model or a combination model. In the rational complex model, the orientation field of fingerprints is expressed as the argument of a rational complex function. In the combination model, it is represented by bivariate polynomials globally and is rectified locally by several point-charge models. As a comparison, the rational complex model is more compact while the combination model approximates more effectively. Both of these models make it feasible to utilize the orientation information into the matching stage.
- Using a feature set consisting of an orientation field and a density map, we can synthesize a new fingerprint, which shows that this feature set constitutes a complete fingerprint representation. However, in order to obtain a robust representation for real applications, minutiae information is still very helpful, so for effective fingerprint matching it may be desirable to use features of minutiae, orientation fields, and density maps

Acknowledgments

This work is supported by National Science Foundation of China (Grant No. 60205002) and National Hi-Tech Development Program of China.

Also, the authors would like to express their gratitude to Prof. Zhaoqi Bian and Prof. Gang Rong for their contributions on technical discussions.

References

- [1] A.K. Jain, R. Bolle, S. Pankanti (Eds.), *BIOMETRICS: Personal Identification in Networked Society*, Kluwer, New York, 1999.

- [2] D. Zhang, Automated biometrics: Technologies and systems, Kluwer Academic Publisher, USA, 2000.
- [3] S. Pankanti, S. Prabhakar, and A. K. Jain, "On the individuality of fingerprints", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.24 (8), pp.1010-1025, 2002.
- [4] A. Jain and L. Hong, "On-line fingerprint verification", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.19 (4), pp. 302-314, 1997.
- [5] B. Sherlock and D. Monro, "A Model for Interpreting Fingerprint Topology", Pattern Recognition, vol. 26 (7), pp.1047-1055, 1993.
- [6] P. Vizcaya and L. Gerhardt, "A Nonlinear Orientation Model for Global Description of Fingerprints", Pattern Recognition, vol. 29 (7), pp. 1221-1231, 1996.
- [7] J. Zhou and J. Gu, "Modeling Orientation Field of Fingerprints with Rational Complex Functions", Pattern Recognition, 2003, in press.
- [8] J. Gu, J. Zhou, and D. Zhang, "A combination Model for orientation Field of Fingerprints", Pattern Recognition, 2003, in press.
- [9] R. Cappelli, A. Erol, D. Maio and D. Maltoni, "Synthetic Fingerprint-image Generation". Proc. of ICPR, pp. 471-474, 2000.
- [10] J. Zhou and J. Gu, "A Model-based Method for the Computation of Fingerprints' Orientation Field", in revised form, submitted to IEEE Trans. on Image Processing, June, 2003.