Fingerprint Reconstruction: From Minutiae to Phase

Jianjiang Feng, *Member*, *IEEE*, and Anil K. Jain, *Fellow*, *IEEE*

Abstract—Fingerprint matching systems generally use four types of representation schemes: grayscale image, phase image, skeleton image, and minutiae, among which minutiae-based representation is the most widely adopted one. The compactness of minutiae representation has created an impression that the minutiae template does not contain sufficient information to allow the reconstruction of the original grayscale fingerprint image. This belief has now been shown to be false; several algorithms have been proposed that can reconstruct fingerprint images from minutiae templates. These techniques try to either reconstruct the skeleton image, which is then converted into the grayscale image, or reconstruct the grayscale image directly from the minutiae template. However, they have a common drawback: Many spurious minutiae not included in the original minutiae template are generated in the reconstructed image. Moreover, some of these reconstruction techniques can only generate a partial fingerprint. In this paper, a novel fingerprint reconstruction algorithm is proposed to reconstruct the phase image, which is then converted into the grayscale image. The proposed reconstruction algorithm not only gives the whole fingerprint, but the reconstructed fingerprint contains very few spurious minutiae. Specifically, a fingerprint image is represented as a phase image which consists of the continuous phase and the spiral phase (which corresponds to minutiae). An algorithm is proposed to reconstruct the continuous phase from minutiae. The proposed reconstruction algorithm has been evaluated with respect to the success rates of type-I attack (match the reconstructed fingerprint against the original fingerprint) and type-II attack (match the reconstructed fingerprint against different impressions of the original fingerprint) using a commercial fingerprint recognition system. Given the reconstructed image from our algorithm, we show that both types of attacks can be successfully launched against a fingerprint recognition system.

Index Terms—Fingerprint synthesis, fingerprint reconstruction, interoperability, minutiae, phase image, orientation field, singularity, AM-FM.

◆

1 INTRODUCTION

FINGERPRINT recognition systems play a crucial role in many situations where a person needs to be verified or identified with high confidence. As a result of the interaction of genetic factors and embryonic conditions, the friction ridge pattern on fingertips is unique to each finger. Fingerprint features are generally categorized into three levels (Fig. 1):

1. Level 1 features mainly refer to ridge orientation field and features derived from it, i.e., singular points and pattern type.
2. Level 2 features refer to ridge skeleton and features derived from it, i.e., ridge bifurcations and endings.
3. Level 3 features include ridge contours, position, and shape of sweat pores and incipient ridges.

Most fingerprint matching systems are based on four types of fingerprint representation schemes (Fig. 2): grayscale image [2], phase image [3], skeleton image [4], [5], and minutiae [6], [7]. Due to its distinctiveness,

compactness, and compatibility with features used by human fingerprint experts, minutiae-based representation has become the most widely adopted fingerprint representation scheme. But other representation schemes do show strong performance, i.e., Bioscrypt's algorithm in FVC2002 and FVC2004 (Fingerprint Verification Competition) [8]. Some minutiae-based matching systems [9], [10], [11] also employ additional features, i.e., orientation field, singular points, ridge count, etc., to improve the matching accuracy. In these representation schemes, the grayscale image has the most information and features at all three levels are recorded (depending on the sensor); compared to grayscale image, phase image and skeleton image lose all Level 3 features and compared with phase image and skeleton image, the minutiae template further loses some Level 2 information, such as ridge path between minutiae.

The widespread deployment of fingerprint recognition systems in various applications has caused concerns that compromised fingerprint templates may be used to make fake fingers, which could then be used to deceive all fingerprint systems the same person is enrolled in. Once compromised, the grayscale image is the most at risk. Leakage of a phase image or skeleton image is also dangerous since it is a trivial problem to reconstruct a grayscale fingerprint image from the phase image or the skeleton image. Fig. 3 shows the reconstructed grayscale image from the phase image $\Psi(x, y)$ by $\cos(\Psi(x, y))$ and that from the skeleton image by distance transform. In contrast to the above three representations, leakage of minutiae templates has been considered to be less serious as it is not trivial to reconstruct a grayscale image from the minutiae.

- *J. Feng is with the Department of Automation, Tsinghua University, Institute of Information Processing, Beijing 100084, China. E-mail: jfeng@tsinghua.edu.cn.*
- *A.K. Jain is with the Department of Computer Science and Engineering, Michigan State University, 3115 Engineering Building, East Lansing, MI 48824-1226, and the WCU Project, Department of Brain and Cognitive Engineering, Korea University. E-mail: jain@cse.msu.edu.*
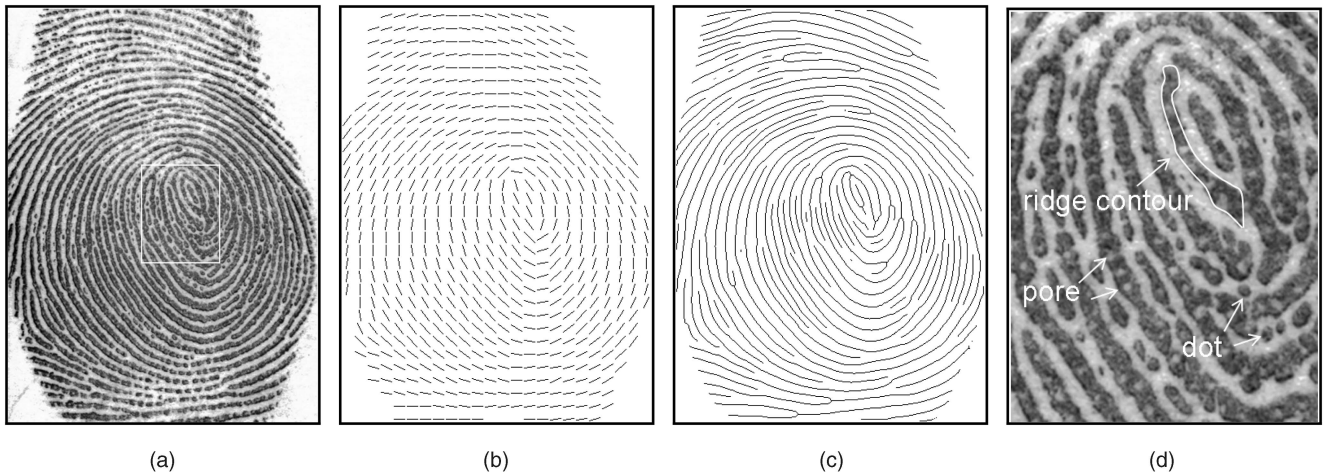
Fig. 1. Features at three levels in a fingerprint. (a) Grayscale image (NIST SD30, A067_11), (b) Level 1 feature (orientation field), (c) Level 2 feature (ridge skeleton), and (d) Level 3 features (ridge contour, pore, and dot).
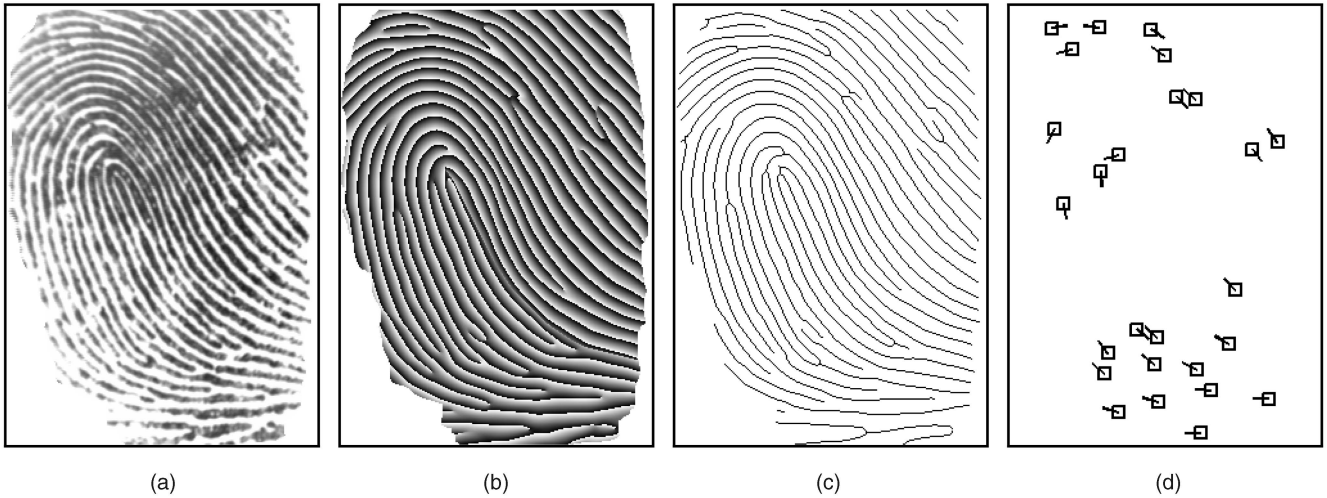

Fig. 2. Fingerprint representation schemes. (a) Grayscale image (FVC2002 DB1, 19_1), (b) phase image, (c) skeleton image, and (d) minutiae.

However, several researchers [12], [13], [14] have shown that it is possible to reconstruct a fingerprint image from the given minutiae template.

The methods of Hill [12] and Ross et al. [13] first reconstruct a skeleton image from minutiae, which is then converted into the grayscale image. In [12], the orientation field is generated based on singular points according to the model in [18]. A line drawing algorithm is used to generate a sequence of splines passing through the minutiae. In [13], the orientation field is estimated using selected minutiae triplets in the template. Streamlines are then traced starting from minutiae and border points. Linear Integral Convolution is used to impart texture-like appearance to the ridges. Finally, the image is smoothed to obtain wider ridges. This reconstruction algorithm can only generate a partial fingerprint. In addition, streamlines that terminate due to distance constraint between adjacent streamlines will generate spurious minutiae. The validity of this reconstruction algorithm was tested by matching 2,000 reconstructed fingerprints against the 2,000 original fingerprints in NIST SD4. A rank-1 identification rate of 23 percent was reported.

Cappelli et al. [14] proposed a technique to directly reconstruct the grayscale image from minutiae. The orientation field is estimated by fitting a modified model initially proposed in [20] to the minutiae directions. Gabor filtering is iteratively performed starting from minutiae on
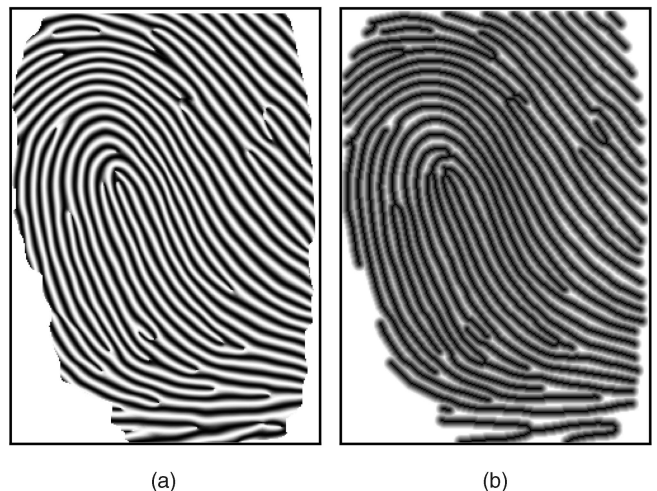

Fig. 3. Reconstruction of grayscale fingerprint image (FVC2002 DB1, 19_1, see Fig. 2a). (a) Reconstructed from phase image and (b) reconstructed from skeleton image.

TABLE 1
Existing Fingerprint Reconstruction (R) and Synthesis (S) Methods

Method	Goal	Input	Model	Performance
Novikov and Glushchenko [15]	S	ridge orientation	iterative filtering	N/A
Araque et al. [16]	S	ridge orientation and frequency	second-order orientation model; filtering using binary mask	N/A
Cappelli et al. [17]	S	singular points, frequency image	orientation model of Sherlock and Monro [18]; Gabor filtering	N/A
Bicz [19]	S	minutiae, ridge orientation and frequency	FM model	N/A
Hill [12]	R	singular points, minutiae	orientation model of Sherlock and Monro [18]; line drawing	N/A
Ross et al. [13]	R	minutiae	minutiae triplets; streamlines; Line Integral Convolution	Type-I attack: 23% rank-1 identification rate on NIST SD4
Cappelli et al. [14]	R	minutiae	orientation model of [18]; Gabor filtering	Type-I attack: 81.49% TAR at 0% FAR on FVC2002 DB1 (average of eight matchers)
Proposed	R	minutiae	FM model	Type-I attack: 94.13% TAR at 0% FAR on FVC2002 DB1, 99.70% rank-1 identification rate on NIST SD4; Type-II attack: 45.89% TAR at 0% FAR on FVC2002 DB1, 65.75% rank-1 identification rate on NIST SD4

an image initialized by the local minutiae pattern. A rendering step is performed to make the reconstructed fingerprint image appear more realistic. The efficacy of this reconstruction algorithm was assessed by attacking nine fingerprint matching algorithms. An average True Accept Rate (TAR) of 81.49 percent at 0 percent False Accept Rate (FAR) was obtained in matching 120 reconstructed fingerprints against the 120 original fingerprints in FVC2002 DB1. However, this algorithm also generates many spurious minutiae in the reconstructed fingerprints.

Fingerprint reconstruction from minutiae (hereinafter simply referred to as fingerprint reconstruction) is very similar to fingerprint synthesis [17] except that the goals and the inputs of the two techniques are different. The goal of fingerprint reconstruction is to obtain an artificial fingerprint that resembles the original fingerprint as much as possible, while the goal of fingerprint synthesis is to generate any artificial fingerprint that is as realistic as possible. For fingerprint reconstruction, the minutiae from a given fingerprint must be provided, while for fingerprint synthesis, no input is needed (except for a statistical model of fingerprint learned from many real fingerprint images).

The well-known SFINGE fingerprint synthesis method of Cappelli et al. [17] performs Gabor filtering on a seed image according to the orientation and frequency images; minutiae automatically emerge during the filtering procedure. Some intraclass variations, such as spatial transformation, touching area, nonlinear distortion, ridge dilation/shrinking, and noise, are simulated to generate realistic impressions of the master fingerprint. One main limitation of SFINGE is that minutiae cannot be controlled. As a result, SFINGE may generate problematic fingerprints that contain too few minutiae or very long ridges. It is well known that the distribution of minutiae in fingerprints is not random and fingerprints of different pattern types have different

minutiae distributions [13]. The minutiae distribution of fingerprints generated by SFINGE may not conform to such distributions since these minutiae are automatically generated during the image filtering process. Similar fingerprint synthesis methods have also been proposed in [15], [16]. The reaction-diffusion technique described in [21] can also be used for synthesizing fingerprints. Bicz [19] described a fingerprint synthesis technique based on the 2D FM model. The phase of the FM model consists of the continuous component and the spiral component, which corresponds to minutiae. A fingerprint is synthesized by first generating each component separately and then combining them. Separation of the continuous phase and the spiral phase makes minutiae controllable. However, the most important step, generating the continuous phase component, was not described in [19]. According to the demo software provided by the author, only a partial fingerprint (around the core) can be generated and the orientation field of each of the four fingerprint pattern types (whorl, left loop, right loop, and arch) is fixed.

Fingerprints can also be synthesized by following the fingerprint formation process. Fingerprints are believed to be formed as a result of the buckling process of the epidermal basal layer, which is caused by regression of volar pads and resistance of nail furrows and flexion creases to the intensive proliferation of the basal layer cells [22]. After simulating the stress, ridge patterns are found as the solution of a von Karman equation. Although the synthesized fingerprints in [22] are basically consistent with real fingerprints, the ridges are not smoothly connected in the area where different ridge systems [23] meet. Table 1 summarizes the eight fingerprint reconstruction and synthesis methods proposed in the literature.

In this paper, a novel approach to fingerprint reconstruction from minutiae template is proposed which first reconstructs a phase image from the minutiae template
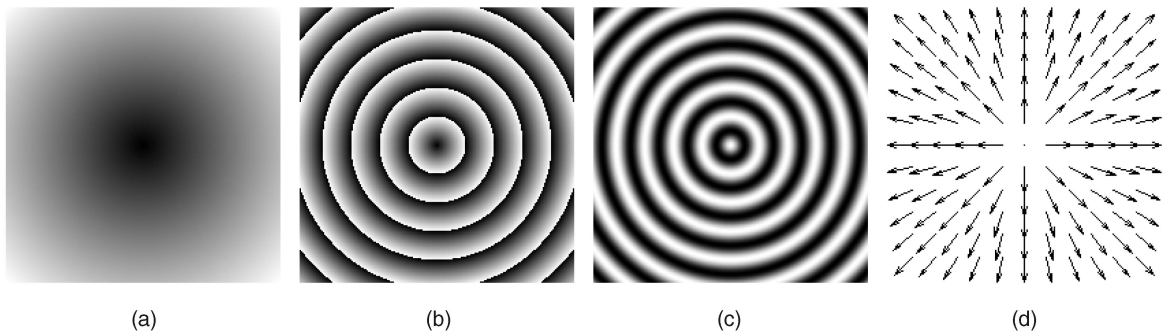
Fig. 4. Continuous phase for a whorl pattern. (a) Continuous phase given by $\sqrt{x^2 + y^2}$, (b) continuous phase modulo 2π, (c) grayscale image given by $cos(\sqrt{x^2 + y^2})$, and (d) gradient of the continuous phase.

and then converts the phase image into the grayscale image. The advantages of our approach over existing approaches to fingerprint reconstruction [12], [13], [14] are: 1) A complete fingerprint can be reconstructed and 2) the reconstructed fingerprint contains very few spurious minutiae. The proposed reconstruction algorithm has been quantitatively assessed by matching reconstructed fingerprints against the corresponding original fingerprints (termed as type-I attack) and against different impressions of the original fingerprints (termed as type-II attack) using a commercial fingerprint SDK, Neurotechnology VeriFinger 4.2 [24]. Type-I attack was found to have a high chance of deceiving the fingerprint recognition system in both the verification and identification experiments. Type-II attack also has a significantly higher accept rate than that of impostor match. A TAR of 94.13 percent at a FAR of 0 percent has been observed in the verification experiment conducted on FVC2002 DB1, and 99.70 percent rank-1 identification rate has been observed in the identification experiment conducted on the NIST SD4 database.

2 FINGERPRINT REPRESENTATION

Larkin and Fletcher [25] proposed representing a fingerprint image as a 2D amplitude and frequency modulated (AM-FM) signal:

$$I(x,y) = a(x,y) + b(x,y)\cos(\Psi(x,y)) + n(x,y), \quad (1)$$

which is composed of four components: the intensity offset $a(x,y)$, the amplitude $b(x,y)$, the phase $\Psi(x,y)$, and the noise $n(x,y)$. Here, we are only interested in the phase $\Psi(x,y)$ since ridges and minutiae are totally determined by the phase; the other three components just make the fingerprint appear realistic. Therefore, an ideal fingerprint can be represented as a 2D FM signal:

$$I(x,y) = \cos(\Psi(x,y)). \quad (2)$$

The gradient of the phase is also termed instantaneous frequency. In a fingerprint image, the direction of instantaneous frequency is normal to the local ridge orientation and the magnitude of instantaneous frequency is equal to the local ridge frequency.

According to the Helmholtz Decomposition Theorem [26], the phase can be uniquely decomposed into two parts: the continuous phase and the spiral phase:

$$\Psi(x,y) = \Psi_C(x,y) + \Psi_S(x,y). \quad (3)$$

Hereinafter, the phase $\Psi(x,y)$ is also termed the composite phase to reflect the fact that it consists of the continuous phase and the spiral phase.

The continuous phase does not contain any rotational component and the integral of its gradient around any simple closed path is zero. For example, the continuous phase given by

$$\Psi_C(x,y) = \sqrt{x^2 + y^2} \quad (4)$$

corresponds to a grayscale image ($cos(\Psi_C(x,y))$) that looks like a whorl pattern (see Fig. 4). Its gradient (instantaneous frequency) is ($cos(\theta), sin(\theta)$), where θ is the angle in the polar coordinate system.

The spiral phase $\Psi_S(x,y)$ consists of a set of N spirals (residues):

$$\Psi_S(x,y) = \sum_{n=1}^{N} p_n \arctan\left(\frac{y - y_n}{x - x_n}\right), \quad (5)$$

where x_n and y_n denote the coordinates of the nth spiral and $p_n \in \{1, -1\}$ denotes its polarity. A spiral with positive polarity is referred to as a positive spiral and with negative polarity is referred to as a negative spiral. The gradient of the spiral phase is not defined in the position of spirals. See Fig. 5 for the phase of a spiral and its gradient.

A minutia emerges after adding a spiral to the continuous phase. Assume a positive spiral is added to the continuous phase shown in Fig. 6a, which is a plane slanted along the y direction. The gradient of the continuous phase is a constant vector field (Fig. 6b). Since

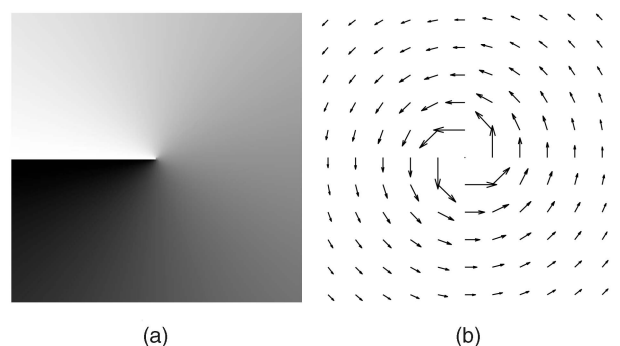(a)   (b)
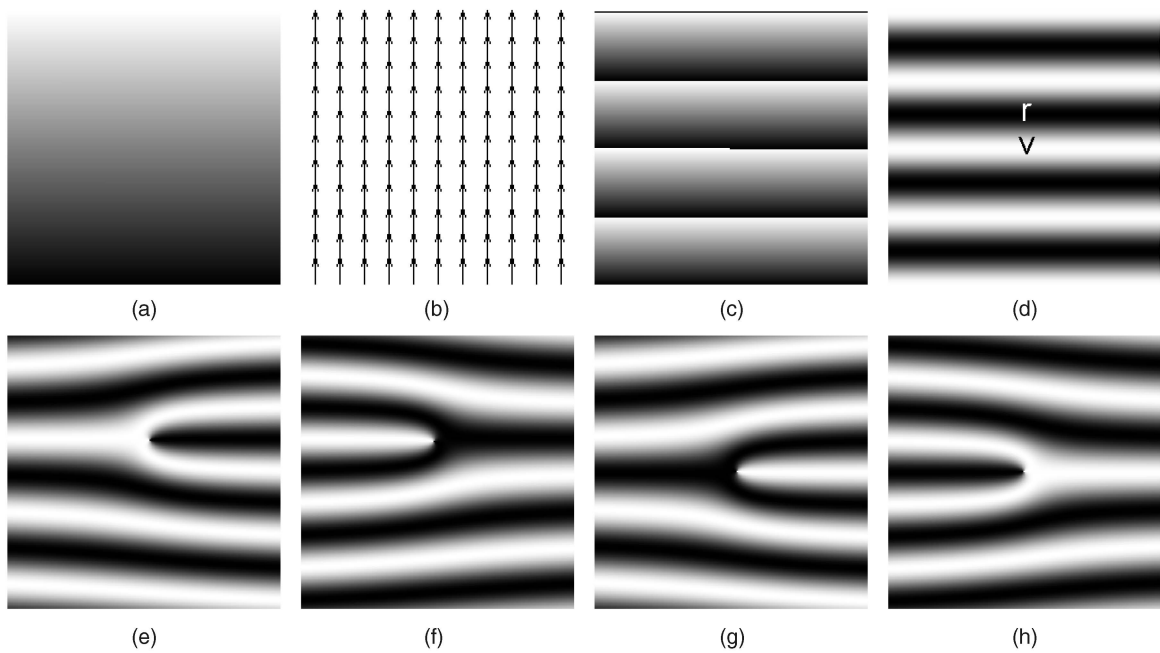
Fig. 5. (a) The spiral phase and (b) its gradient.

Fig. 6. Relationship among minutia, continuous phase, and spiral. (a) Continuous phase given by $2\pi fy$, (b) gradient of the continuous phase, (c) continuous phase modulo 2π, (d) grayscale image given by $cos(2\pi fy)$ (spiral locations are marked), (e) composite image with a positive spiral at "r," (f) composite image with a negative spiral at "r," (g) composite image with a positive spiral at "v," and (h) composite image with a negative spiral at "v."

the gradient of the spiral phase (Fig. 5b) is a rotational vector field, the gradient magnitude of the composite phase on the left side of the spiral is smaller than f and the gradient magnitude of the composite phase on the right side of the spiral is larger than f. In other words, ridge frequency decreases on the left side and increases on the right side. This causes the formation of a minutia with a direction of 0 degree (Fig. 6e). Adding a negative spiral to the same position marked by "r" forms a minutia with a direction of 180 degrees (Fig. 6f). Given the polarity of the spiral, the minutia type (namely, bifurcation or termination) is determined by the position of the spiral. A positive spiral added on the ridge marked by "r" results in a termination (Fig. 6e), while a bifurcation is generated when the positive spiral is added in the valley marked by "v" (Fig. 6g). A negative spiral added on the ridge marked

by "r" results in a bifurcation (Fig. 6f), while a termination is generated when the negative spiral is added in the valley marked by "v" (Fig. 6h).

Fig. 6 shows that the local ridge orientation in the neighborhood of the spiral is slightly changed after the spiral is added. In fact, the addition of the spiral also affects the local ridge orientation in the entire image. This phenomenon is not obvious in Fig. 6, but it becomes evident in Fig. 7, where local ridge orientations far from spirals are significantly changed when a large number of spirals with the same polarity are added. This phenomenon is very common in the area close to the delta of fingerprints and in the funnel area of palmprints.

Fig. 8a shows a synthesized fingerprint that only contains the continuous phase. Adding seven spirals to the continuous phase leads to the synthesized fingerprint in Fig. 8b.
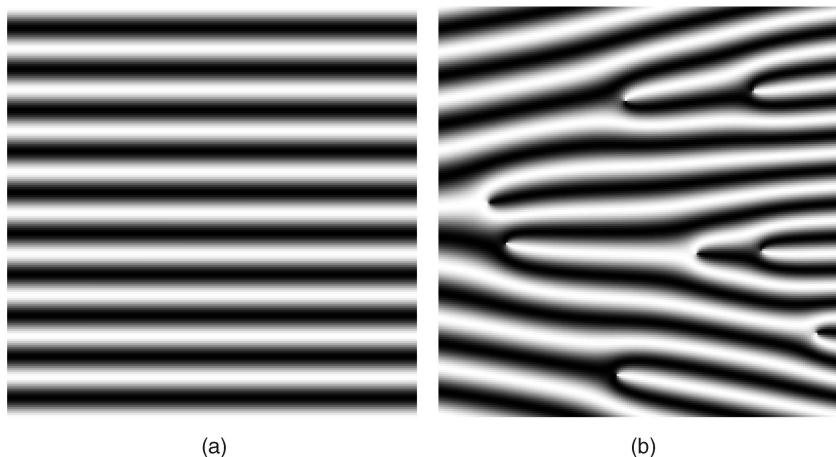

Fig. 7. Effect of spirals on local ridge orientation. (a) Grayscale image of the continuous phase given by $(2\pi fy)$ and (b) grayscale image of the composite phase after adding eight positive spirals.
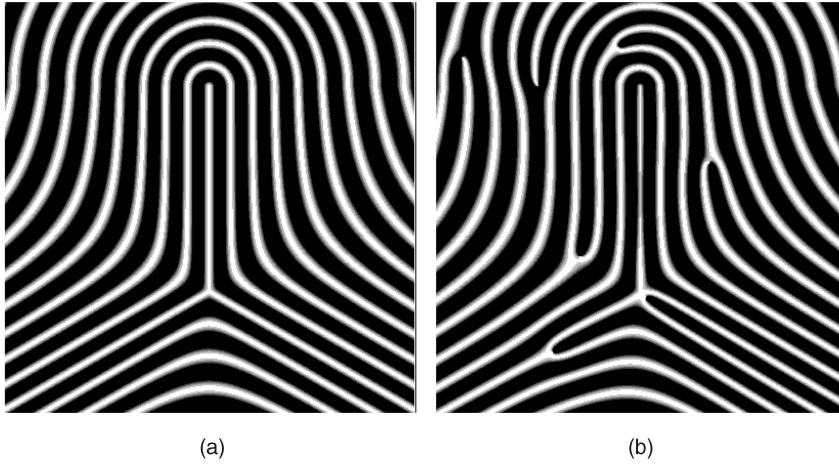
Fig. 8. Synthetic fingerprint (a) without minutiae and (b) with seven minutiae.

3 FINGERPRINT RECONSTRUCTION

3.1 Problem Statement

Given a set of N fingerprint minutiae $\{x_n, y_n, \alpha_n\}, 1 \leq n \leq N$, where (x_n, y_n) and α_n denote the location and direction of the nth minutia, respectively, the goal is to reconstruct the original fingerprint image modeled by (1). In terms of the FM model, this input means that we are given 1) the spiral phase and 2) the direction of instantaneous frequency of the composite phase at the locations of the N minutiae. This is an ill-posed problem since the important information required to reconstruct the continuous phase of fingerprints, namely, the ridge frequency, is unknown. Information needed to reconstruct realistic fingerprints, such as brightness, contrast, the background noise of fingerprint sensor, and detailed ridge features (pores, ridge contours, etc.) is also not available. Thus, a more practical goal is to first estimate the FM representation of the original fingerprint, $cos(\Psi(x, y))$. The 8-bit grayscale fingerprint image is then computed as:

$$I(x, y) = \frac{1 + \cos(\Psi(x, y))}{2} \times 255. \qquad (6)$$

To obtain the phase $\Psi(x, y)$, the following four steps are performed:

1. orientation field reconstruction,
2. estimation of gradient of continuous phase,
3. continuous phase reconstruction, and
4. combination of the spiral phase and the continuous phase.

The flow chart of the proposed fingerprint reconstruction algorithm is depicted in Fig. 9.

3.2 Reconstruction of Orientation Field

Ross et al. [13] used selected minutiae triplets to estimate the orientation field in triangles. Cappelli et al. [14] estimated orientation field by fitting an orientation field model to the orientations at minutiae. We propose a novel orientation field reconstruction algorithm that can work even when only one minutia is available.

The image is divided into nonoverlapping blocks of 8×8 pixels and an orientation value is computed for each foreground block. A foreground mask for the fingerprint image is obtained by dilating the convex hull of minutiae

using a disk-shaped structuring element of 8×8 pixels. The local ridge orientation at block (m, n) is predicted by using the nearest minutia in each of the eight sectors (see Fig. 10). The minutia direction α_k is doubled to make α_k equivalent to $\alpha_k + \pi$. The cosine and sine components of $2\alpha_k$ of all of the K selected minutiae are summed as:

$$u = \sum_{k=1}^{K} \cos(2\alpha_k) w_k, \qquad (7)$$

$$v = \sum_{k=1}^{K} \sin(2\alpha_k) w_k, \qquad (8)$$

where w_k is a weighting function. In our experiment, the reciprocal of the euclidean distance between the block center and the kth minutia is used in order to make minutiae direction dominate the ridge orientation of neighboring blocks. Other weighting functions, such as the Gaussian function [27], can also be used. Then, the orientation at block (m, n) is computed as:

$$O(m, n) = \frac{1}{2}\arctan\left(\frac{v}{u}\right). \qquad (9)$$

In the event that fingerprint singular points (core, delta) are also provided, a different approach is used to reconstruct the orientation field in order to avoid a possible shift of singularity [17].

1. The directions of minutiae are modified by subtracting the direction field of N_s singular points [18], [28]:

    $$D_s(m, n) = \frac{1}{2}\sum_{i}^{N_s} t_{s_i} \arctan\left(\frac{n - n_{s_i}}{m - m_{s_i}}\right), \qquad (10)$$

    where m_{s_i}, n_{s_i}, and t_{s_i} (core: 1, delta: -1) denote the location and type of the ith singular point.
2. The algorithm described in the preceding paragraph is used to reconstruct the continuous direction $D_c(m, n)$ at all foreground blocks.
3. The orientation field $O(m, n)$ is computed as $(D_s(m, n) + D_c(m, n))$.

Fig. 11 shows the reconstructed orientation fields for two fingerprints when singular points are provided and not provided, respectively. The reconstructed orientation fields
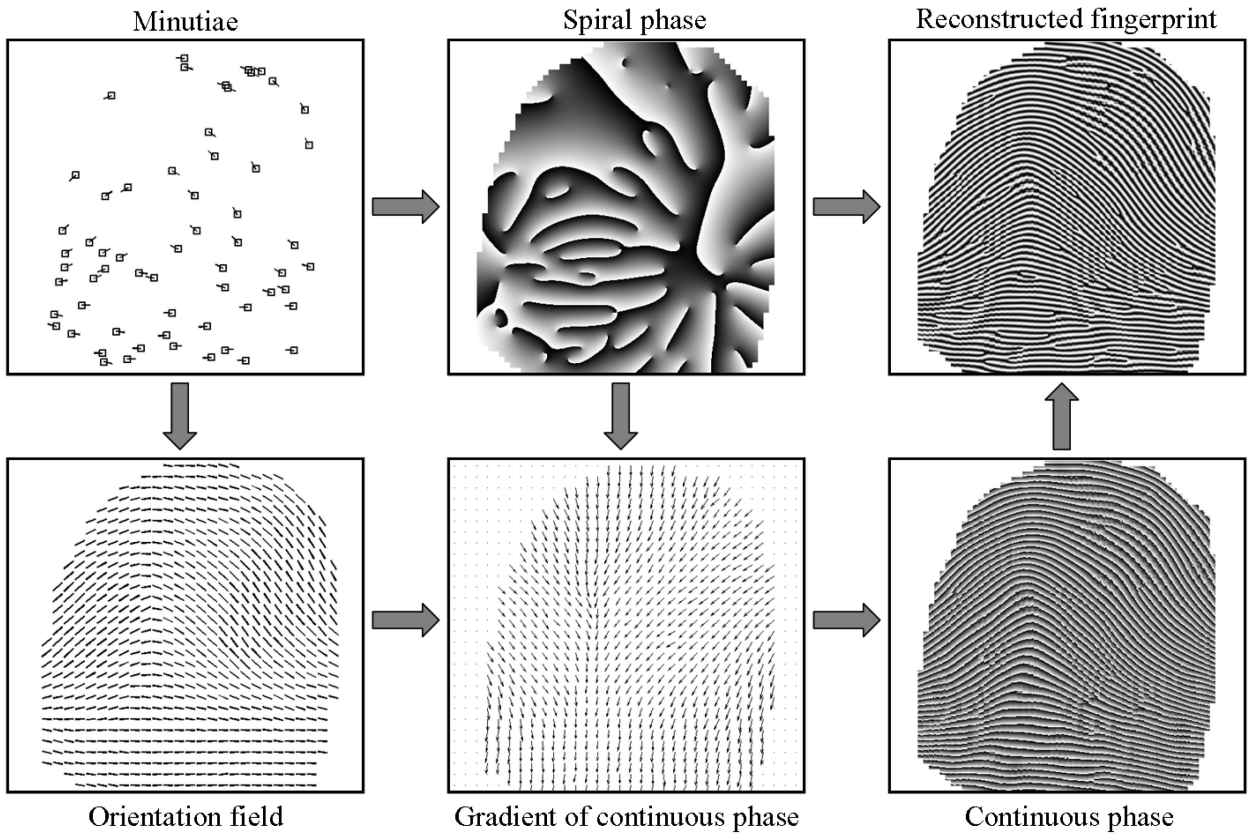
Fig. 9. Flow chart of the proposed fingerprint reconstruction algorithm. The reconstructed fingerprint image shown here is NIST SD4, F0285.

in these two cases are quite similar except for the small difference in the singularity area. The direction of singular points is not used here because it is a redundant feature as long as a few minutiae in the neighborhood of singularity are known, which is almost always the case in our data set.[1] As shown in Fig. 11, the reconstructed orientation field matches the original fingerprint very well, even if the singularity direction is not used.

To evaluate the proposed orientation field reconstruction algorithm (using minutiae-only templates) quantitatively, an experiment was done using all the 2,000 file fingerprints in the NIST SD4 database. The mean of the difference (in degree) between the reconstructed and original orientation fields in the foreground region is computed and the histogram is obtained. The original orientation field is estimated from the skeleton image of the original image output by VeriFinger SDK because it is found to be more robust to noise than the direct estimation from grayscale image. The foreground mask is also obtained from the skeleton image. To evaluate the robustness of the proposed algorithm to the number of available minutiae, this experiment was repeated using a randomly chosen subset (80 percent, 60 percent, 40 percent, and 20 percent) of all minutiae in each fingerprint. The same foreground mask is used for all five experiments to make the comparison fair. The histograms of difference in orientation fields are given in Fig. 12. When all of the minutiae are available, the average orientation difference is around 5 degrees for most of the fingerprints and, for no fingerprint is the difference

greater than 8 degrees. We also observed that the performance does not drop much even when only 60 percent of all the minutiae are available for reconstruction.

3.3 Estimation of Gradient of Continuous Phase

The gradient of the continuous phase $G_c(m, n)$ at block (m, n) can be computed as:

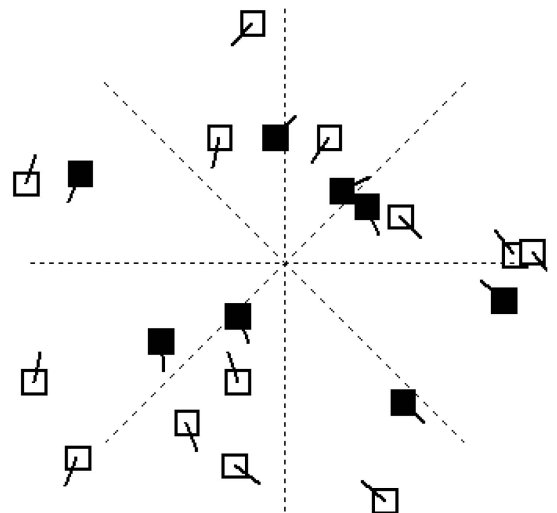$$G_c(m, n) = G(m, n) - G_s(m, n), \qquad (11)$$

Fig. 10. Local ridge orientation is predicted by using the nearest minutiae (denoted by filled squares) in each of the eight sectors.

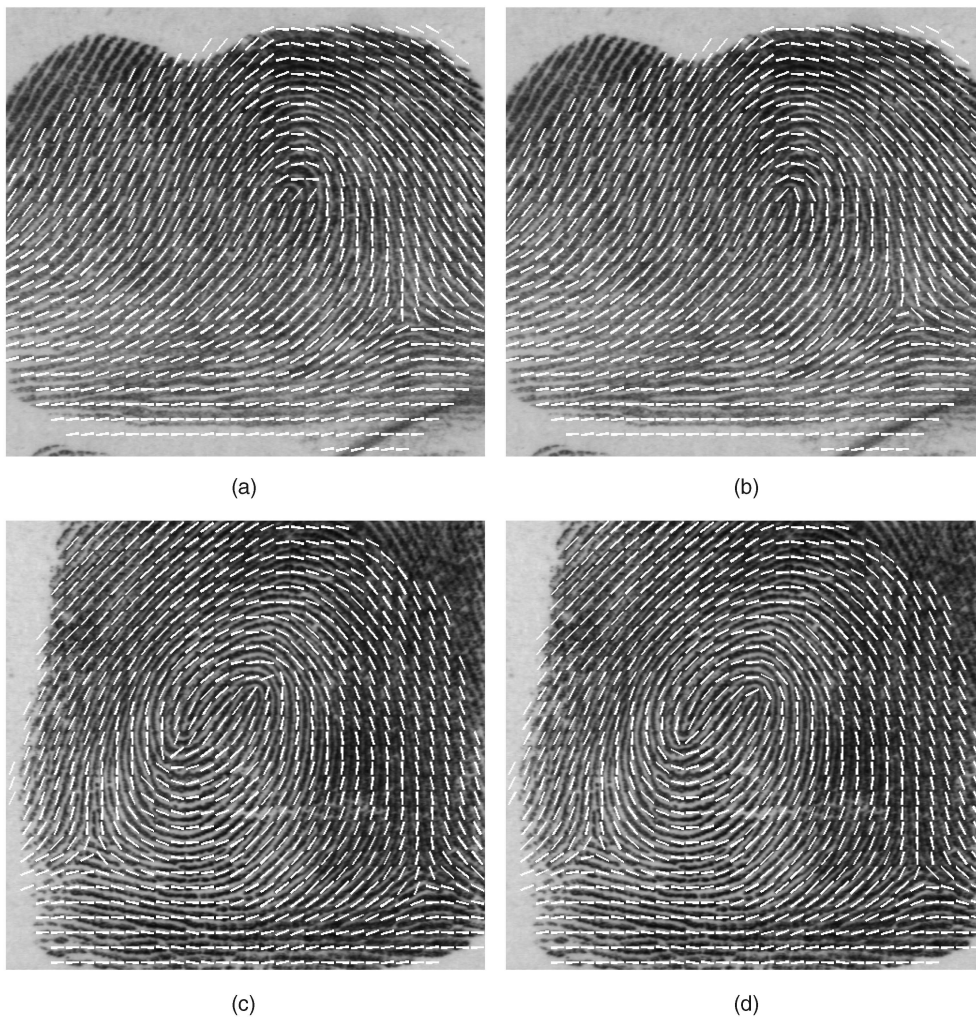1. It is well known that minutiae density is very high in the singularity region of a fingerprint.

Fig. 11. Orientation field reconstructed for two different fingerprint images in NIST SD4. (a) F0086 using minutiae, (b) F0086 using minutiae and singular points, (c) F0025 using minutiae, and (d) F0025 using minutiae and singular points.

where $G(m,n)$ and $G_s(m,n)$ represent the gradients of the composite phase and of the spiral phase, respectively. $G_s(m,n)$ can be easily computed from the spiral phase in (5). Although $G(m,n)$ is normal to local ridge orientation, its direction cannot be simply computed as $O(m,n) + \pi/2$ for two reasons. First, this may produce discontinuity in phase gradient since the orientation field is wrapped in the range $[0, \pi)$. Second, both ridge orientation and frequency are not well defined in the neighborhood of minutiae.

To deal with the first problem, we unwrap the initial orientation field, $O(m,n)$, to obtain an unwrapped orientation field, $O_u(m,n)$. This is basically a phase unwrapping problem [26], except for the trivial difference that phase is wrapped in the range $[0, 2\pi)$, while orientation is wrapped in the range $[0, \pi)$. Starting from the top left-most foreground block, say block (m,n), whose initial orientation is directly set as its unwrapped orientation, the orientation at an adjacent block, say block $(m+1, n)$, is unwrapped by adding $k\pi$ to its initial orientation $O(m+1, n)$. Here, k is an integer number that makes the following inequality hold:

$$|O_u(m+1, n) - O_u(m, n)| \leq \pi/2. \qquad (12)$$

The above operation is performed in a specific order until all of the foreground blocks have been unwrapped. For fingerprints without singularity, depth-first, breadth-first,
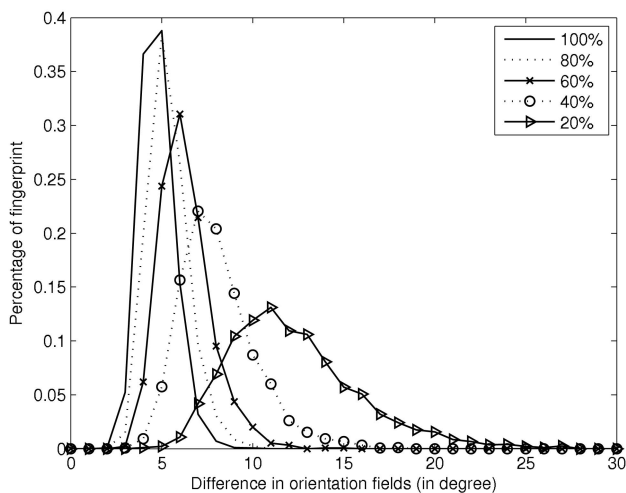

Fig. 12. Histograms of the average difference (in degree) between the reconstructed and original orientation fields of 2,000 file fingerprints in NIST SD4. The orientation field is reconstructed by using 100 percent, 80 percent, 60 percent, 40 percent, and 20 percent of all minutiae in each fingerprint, respectively.
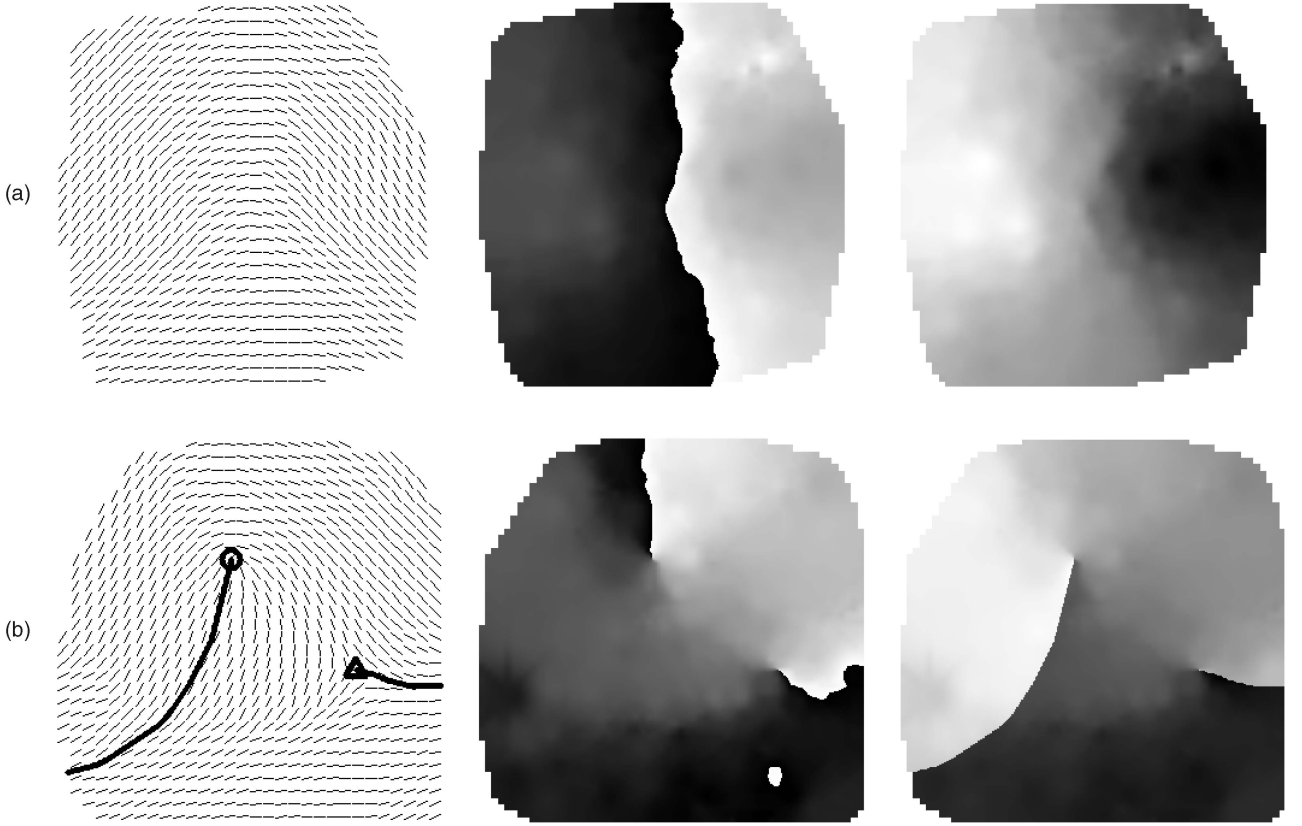
Fig. 13. Orientation field unwrapping for (a) an arch-type fingerprint (NIST SD4, F0006) and (b) a loop-type fingerprint (F0019). From left to right: orientation field with singular points and branch cuts marked, wrapped orientation field shown as grayscale image, and unwrapped orientation field shown as grayscale image.

or other strategies can be adopted to visit and unwrap blocks, since the result, $O_u(m, n)$, is independent of the order [26]. As shown in Fig. 13a, the discontinuity of orientation value[2] in the initial/wrapped orientation field of an arch-type fingerprint disappears in the unwrapped orientation field.

However, when fingerprints contain singular points which are detected from the reconstructed orientation field using the Poincaré index method [17], orientation unwrapping is not path/order independent. In this case, we use the well-known branch cut phase unwrapping algorithm [29].

1. Branch cuts are obtained by tracing the orientation field starting from each singular point until the border of the foreground region or meeting another singular point, as suggested in [25], [30].
2. Starting from the top left-most foreground block, foreground blocks are visited and unwrapped in the same manner as in fingerprints without singularity, except that here branch cuts cannot be crossed.
3. The blocks on the branch cuts are unwrapped.

An example is given in Fig. 13b to show the result of orientation unwrapping for a loop-type fingerprint. Note that while discontinuity in the area without singularity

2. Although the initial orientation field in Fig. 13a is continuous everywhere when it is shown as a line map, the orientation value is not continuous in the middle of the fingerprint when it is shown as a grayscale image.

disappears in the unwrapped orientation field, the discontinuity caused by singularity is unavoidable.

Based on the unwrapped orientation field, $O_u(m, n)$, $G(m, n)$ is computed as:

$$G(m, n) = 2\pi\, f \exp(i(O_u(m, n) + \pi/2)), \qquad (13)$$

where f denotes the ridge frequency. Since it is not possible to estimate the ridge frequency from minutiae (if the ridge count information between minutiae is provided, then it is suitable to estimate the ridge frequency), we have used a constant frequency value 0.12 for the whole image, which corresponds to a ridge wavelength of 8.3 pixels in 500 ppi images.

To deal with the second problem, we compute $G_c(m, n)$ according to (11) in blocks that contain no minutiae. Then, $G_c(m, n)$ in the remaining blocks is interpolated using the neighboring values.

3.4 Reconstruction of Continuous Phase

The continuous phase of a fingerprint is modeled by piecewise planes at each foreground block (m, n) of 8×8 pixels:

$$\Psi_C(x, y) = G_{cx}(m, n)x + G_{cy}(m, n)y + P(m, n),$$
$$8(m - 1) \le x < 8m, 8(n - 1) \le y < 8n, \qquad (14)$$

where $G_{cx}(m, n)$ and $G_{cy}(m, n)$ denote the two components of $G_c(m, n)$, and $P(m, n)$ denotes the phase offset at block (m, n).
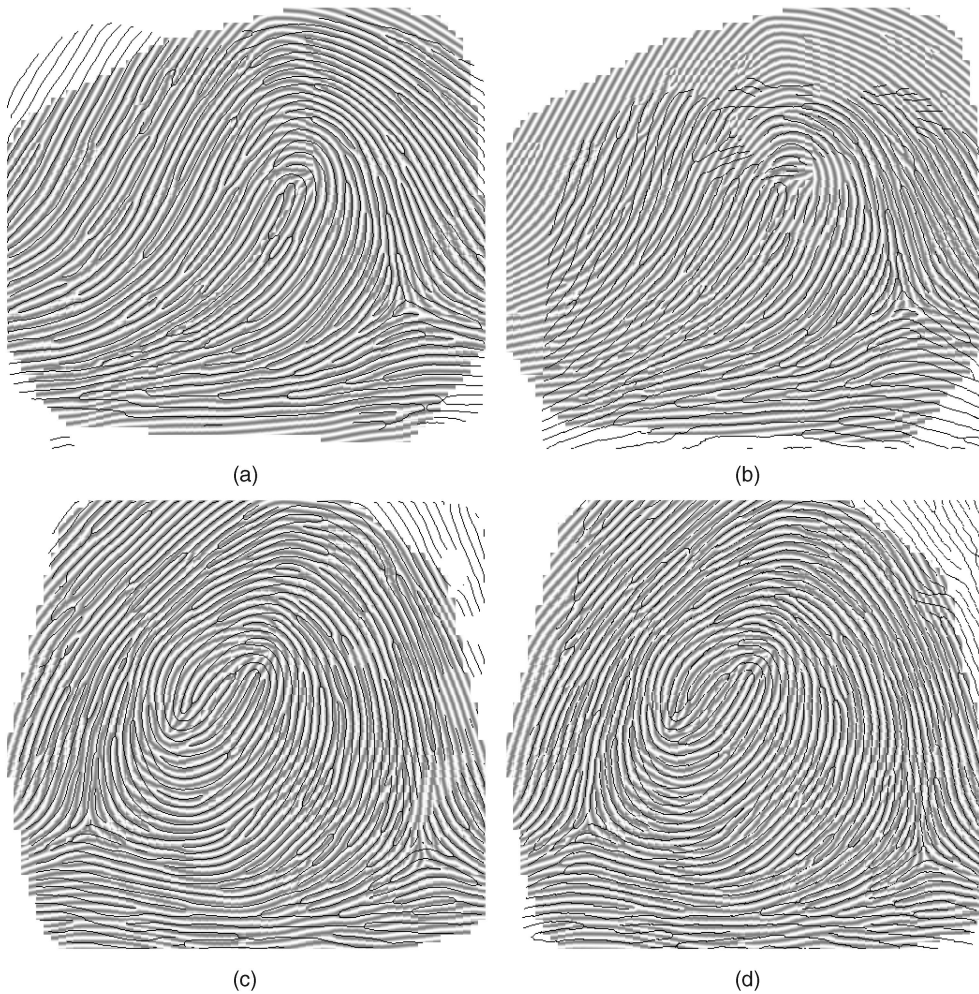
Fig. 14. Reconstructed images for two fingerprints in NIST SD4. (a) The reconstructed image of F0086 and the original skeleton image of F0086, (b) the reconstructed image of F0086 and the original skeleton image of S0086, (c) the reconstructed image of F0025 and the original skeleton image of F0025, (d) the reconstructed image of F0025 and the original skeleton image of S0025.

The only unknown value in (14), the phase offset $P(m,n)$, is estimated by the following algorithm.

Starting with a queue containing the top left-most block (whose phase offset is assumed to be 0), in each iteration a block is obtained from the queue and each of its four-connected neighbors is checked to see if it has been reconstructed (namely, the phase offset has been estimated). If one of the neighboring blocks has not been reconstructed, the phase offset of this block is estimated and it is put into the queue. This procedure is performed until the queue is empty (which means that the continuous phase has been reconstructed at all of the foreground blocks). An ancillary image is used to record the reconstructed blocks.

Here, we describe how to estimate the phase offset of a block using all of the already reconstructed four-connected neighbors. Consider one of the neighbors, say block $(m-1, n)$, of block (m, n). The phase images of these two blocks are required to be continuous at the border pixels $\{(x, y) : x = 8(m-1), 8(n-1) \le y < 8n\}$. For each border pixel (x, y), a phase offset of block (m, n) is estimated:

$$\psi = G_{cx}(m-1, n)x + G_{cy}(m-1, n)y + P(m-1, n) \\ - G_{cx}(m, n)x - G_{cy}(m, n)y. \qquad (15)$$

$P(m, n)$ is then estimated as the mean value of these phase offsets. It should be noted that phase values cannot be averaged directly, but should be first converted to complex numbers which are averaged and reconverted to phase.

After the continuous phase is reconstructed, the reconstructed fingerprint can be obtained by combining the continuous phase and the spiral phase. In Fig. 14, the reconstructed fingerprint is overlaid on the skeleton image of the original fingerprint and of a different impression of the same finger. The reconstructed fingerprints match the original fingerprints well. But, due to distortion, the reconstructed fingerprint does not match the different impression as well as the original fingerprint (Figs. 14a and 14b). The reconstructed fingerprints do contain a few spurious minutiae, especially in the region of singularity. The main cause for the spurious minutiae is the assumption of a fixed ridge frequency for the whole image, which is not true for real fingerprints.

The piecewise planar model for the continuous phase does introduce blocking effects in the reconstructed fingerprints (see Fig. 15b). However, we have not yet been able to find a better model for all types of fingerprints because 1) it is very difficult to find an explicit function to model the continuous phase of a whole fingerprint with singularity and 2) splitting a fingerprint into many large
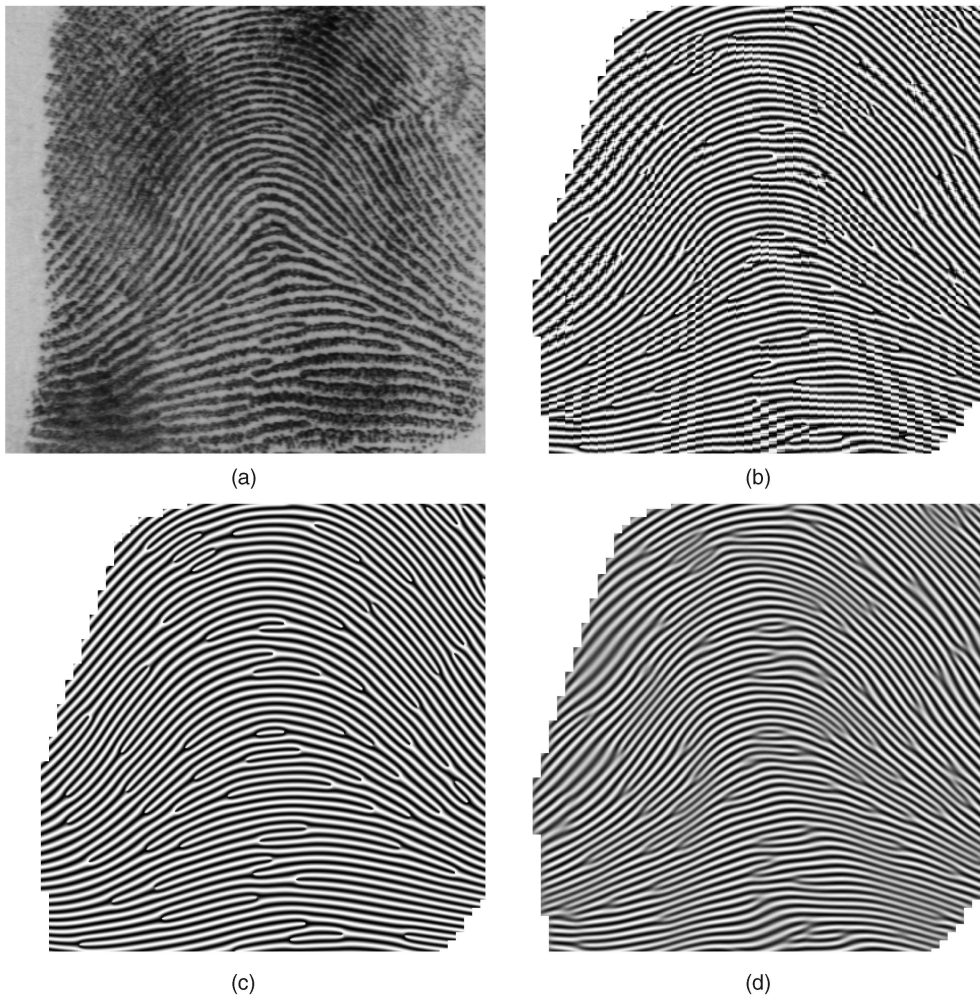
Fig. 15. Blocking effect in reconstructed fingerprints (based on minutiae-only template). (a) Grayscale image (NIST SD4, S0006), (b) reconstructed image using the piecewise planar model, (c) reconstructed image using third order polynomial fitting, and (d) enhanced image of (b) using Gabor filtering.

regions and using higher order polynomials to model the continuous phase in each region can only move the discontinuity to the border between the adjacent regions. However, for fingerprints without singularity (such as arch-type fingerprints or partial fingerprints), a high order polynomial (third order polynomial was used in our experiments) can be used to approximate $G_c(m,n)$, the gradient of the continuous phase of the whole fingerprint; model parameters can be estimated using the least squares method. Then, the explicit function of the continuous phase can be simply obtained from the function of its gradient. As shown in Fig. 15c, the reconstructed fingerprint using the high order polynomial model is very smooth and does not contain any spurious minutiae. Another method to reduce blocking effects is Gabor filtering (see Fig. 15d).

4 EXPERIMENTS

The proposed fingerprint reconstruction algorithm was used to reconstruct plain and rolled fingerprints,[3] respectively. A

3. Rolled fingerprints are obtained by rolling a finger from one side to the other side in order to capture all of the ridge details of a finger. Plain fingerprints are obtained by directly pressing the finger on a flat surface [17].

reconstructed fingerprint may be used to attack the system that stores the original fingerprint template (termed type-I attack) or other systems where the same finger has also been enrolled with a different impression (termed type-II attack). Such fingerprint recognition systems may work either in the verification mode or in the identification mode. To evaluate the performance of the proposed reconstruction algorithm in these situations, VeriFinger 4.2 by Neurotechnology [24] was assumed to be the fingerprint recognition system. To understand the effect of additional features (besides minutiae) on the reconstruction performance, we reconstruct fingerprints based on three types of templates: 1) minutiae, 2) minutiae and singular points, and 3) minutiae and orientation field. As a comparison, we also report the matching accuracy when the original grayscale images are directly used to attack the system, which may be thought of as the fourth type of template.

4.1 Plain Fingerprints

The verification experiment was performed on FVC2002 DB1_A [31], which consists of 100 fingers and eight plain impressions per finger. Fingerprints were reconstructed from three types of templates of all the 800 images. For each
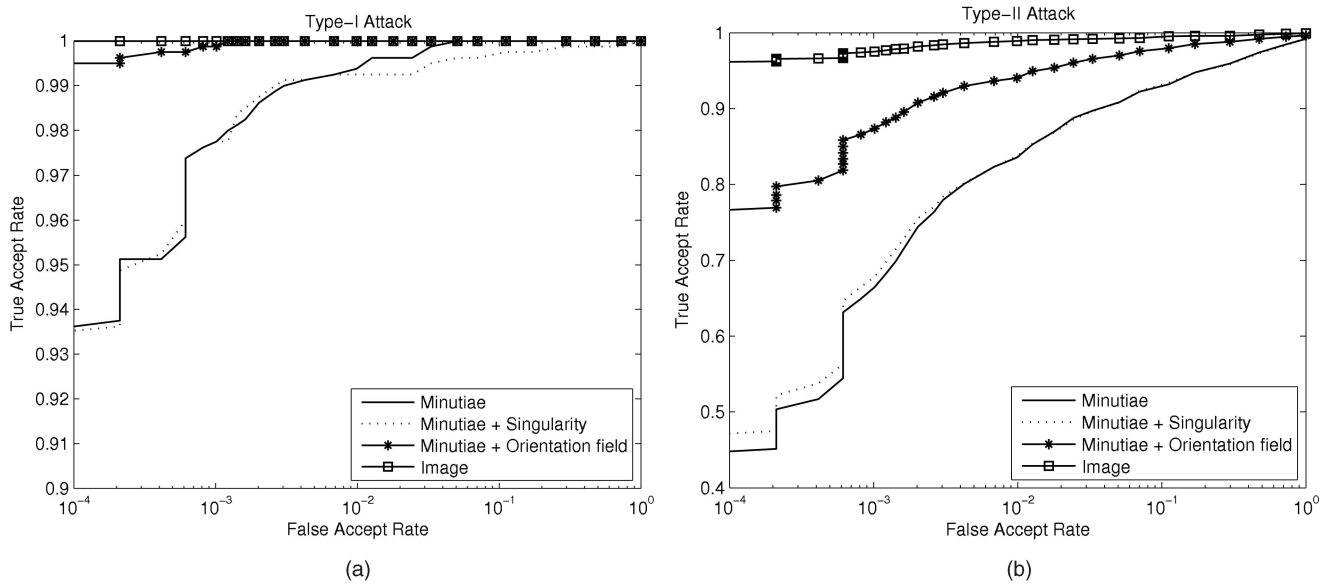
Fig. 16. ROC curves of the proposed reconstruction algorithm on FVC2002 DB1_A. (a) Type-I attack and (b) type-II attack.

type of template, each reconstructed fingerprint was matched against all eight impressions of the same fingerprint in the database, which produced 800 type-I attacks and 5,600 type-II attacks. Similarly, original fingerprints were matched against all eight impressions of the same fingerprint to produce 800 type-I attacks and 5,600 type-II attacks. A total of 4,950 impostor matches were obtained by cross matching the first impressions of different fingerprints. By changing the decision threshold, the accept rates of type-I attack, type-II attack, genuine match, and impostor match were obtained. The four Receiver Operating Characteristic (ROC) curves in Fig. 16a are plots of the true accept rates (termed as TAR) of type-I attack using four types of templates against the false accept rates (termed as FAR) of impostor match.

It can be observed that even under a secure setting (0.1 percent FAR) and using minutiae-only template, both type-I attack and type-II attack have a reasonably high accept rate (94.13 percent and 45.89 percent, respectively). As expected, type-I attack has a higher chance than type-II attack in successfully deceiving a fingerprint verification system. But the impact of a type-II attack should not be underestimated since it can be launched against all fingerprint systems where the same finger has been enrolled. While the availability of the orientation field can significantly improve the reconstruction performance, singular points have negligible impact on the reconstruction performance, which was also observed in [14].

It should be noted that this performance cannot be directly compared to the performance reported in [14] since

1. Only type-I attack was considered in [14].
2. Only the first impression (with large image size and good image quality) of 120 fingers in FVC2002 DB1 was used for reconstruction in [14]. In contrast, we have used all eight impressions of 100 fingers in FVC2002 DB1_A, which contains many fingerprints with a very small area.

3. Four fingerprints which are reconstructed using four different ridge frequencies are matched against the template and then the maximum score is adopted in [14]; we generate only one reconstructed image by using a fixed ridge frequency.
4. Different fingerprint matchers have very different ability in dealing with artificial fingerprints, as shown in [14, Table 4]. We have used a well-known state-of-the-art commercial matcher [24].

4.2 Rolled Fingerprints

The identification experiment was performed on NIST SD4 [32], which consists of 2,000 pairs of ink-on-paper rolled fingerprints (called file and search fingerprints, respectively). Fingerprints were reconstructed from the templates of 2,000 file fingerprints. For each of the four types of templates, the reconstructed fingerprints were matched against 2,000 file fingerprints and 2,000 search fingerprints to obtain 2,000 type-I attacks and 2,000 type-II attacks, respectively. The Cumulative Match Characteristic (CMC) curves of type-I attack and type-II attack are given in Fig. 17.

The rank-1 identification rate of 99.70 percent indicates that type-I attack with minutiae-only template has a very high identification rate in deceiving a fingerprint identification system, even higher than the identification rate of type-II attack with the image template (92.5 percent). This indicates that the reconstructed fingerprints are very similar to the original ones. The rank-1 identification rate of type-I attack is improved to 100 percent when the orientation field is used in addition to minutiae. Similarly to the verification experiment in Section 4.1, inclusion of singular points shows negligible impact on the performance.

The rank-1 identification rate of type-I attack with minutiae-only template reported in [13] is only 23 percent. Since the same fingerprint database has been used and the same minutiae extraction and matching algorithms have been employed, we can conclude that the proposed reconstruction algorithm performs better than the algorithm of Ross et al. [13].
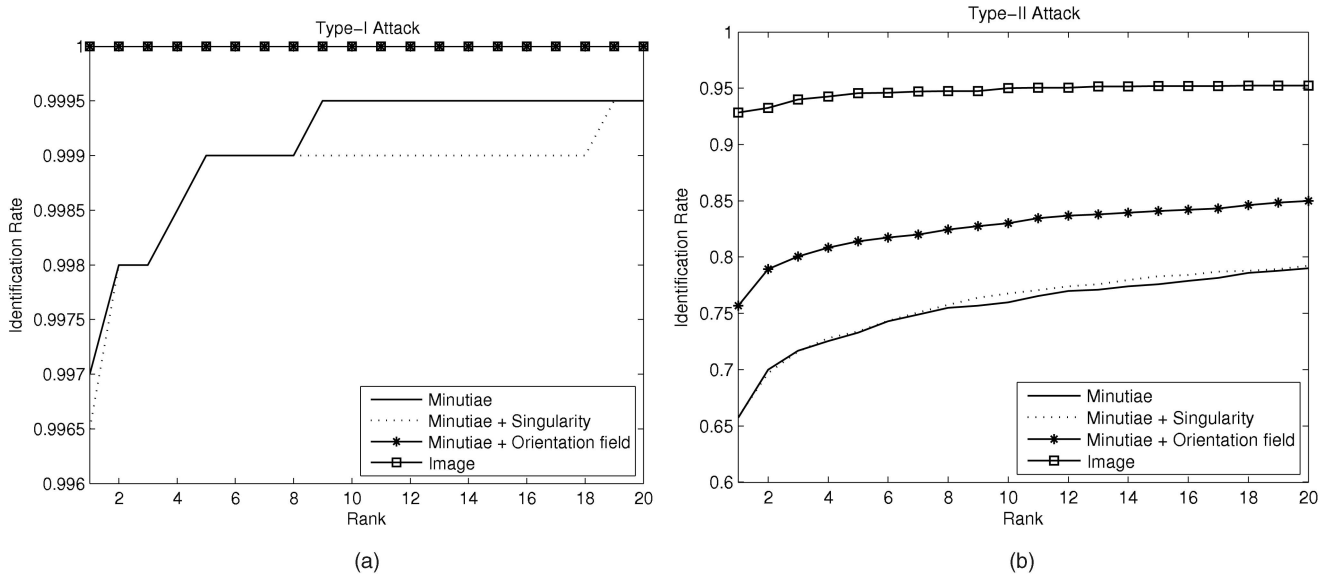
Fig. 17. CMC curves of the proposed reconstruction algorithm on NIST SD4. (a) Type-I attack and (b) type-II attack.

4.3 Computational Requirements

The reconstruction of fingerprints with the minutiae-only template in FVC2002 DB1 and NIST SD4 takes around 1.40 and 4.16 seconds per image, respectively, when tested on a PC with 3 GB of RAM and a 3 GHz Pentium 4 CPU. The algorithm is currently implemented in MATLAB and we expect the computational costs to be significantly reduced after optimization.

5 CONCLUSION AND FUTURE WORK

A novel fingerprint reconstruction scheme has been proposed which is based on converting the minutiae representation to the phase representation. The phase is composed of the continuous phase and the spiral phase. A reconstructed fingerprint is obtained by reconstructing the orientation field, reconstructing the continuous phase, and combining the continuous phase with the spiral phase. The experimental results show that the reconstructed image is very consistent with the original fingerprint and that there is a high chance of deceiving a state-of-the-art commercial fingerprint recognition system.

The reconstructed fingerprints still contain a few spurious minutiae, especially in the high-curvature regions. To overcome this problem, a better model for the continuous phase of fingerprints of any pattern type should be developed. To obtain reconstructed images that are even more consistent with the original fingerprints, ridge frequency and minutiae type should be utilized. To make the reconstructed fingerprints appear visually more realistic, brightness, ridge thickness, pores, and noise should be modeled. The accept rate of the reconstructed fingerprints can be further improved by reducing the image quality around the spurious minutiae. To reduce the risk of attacks using reconstructed fingerprints, robust fingerprint template security [33] and spoof detection techniques [34] should be developed.

Fingerprint reconstruction may also be used for improving the interoperability among minutiae encoders and matchers from different vendors, which was identified as a problem in the NIST MINEX testing [35]. By reconstructing fingerprint images from standard templates encoded by vendor A, vendor B may extract and utilize proprietary features from the reconstructed images which have the potential to provide better performance than standard templates. But, we suggest that only the reconstructed orientation field should be used since the additional features generated by our current algorithm are less reliable.

While the proposed algorithm is designed for fingerprint reconstruction, its underlying ideas, namely, representing fingerprints using phase, decomposing phase into continuous phase and spiral phase, and modeling the continuous phase with piecewise polynomials, may also have use in fingerprint enhancement and matching. Currently, we are trying to apply the reconstruction method to the difficult and important problem of latent fingerprint restoration. See Fig. 18 for a direct use of the proposed reconstruction algorithm to a latent fingerprint in NIST SD27. Both the ridge flow and minutiae in the reconstructed fingerprint match the original fingerprint well. But, apparently, the reconstructed ridge pattern does not match the original ridge skeleton exactly. The current reconstruction algorithm has to be significantly modified to utilize all of the available input in the latent fingerprint restoration problem, such as the grayscale image and other features (such as ridge orientation and skeleton) manually marked by the latent expert.
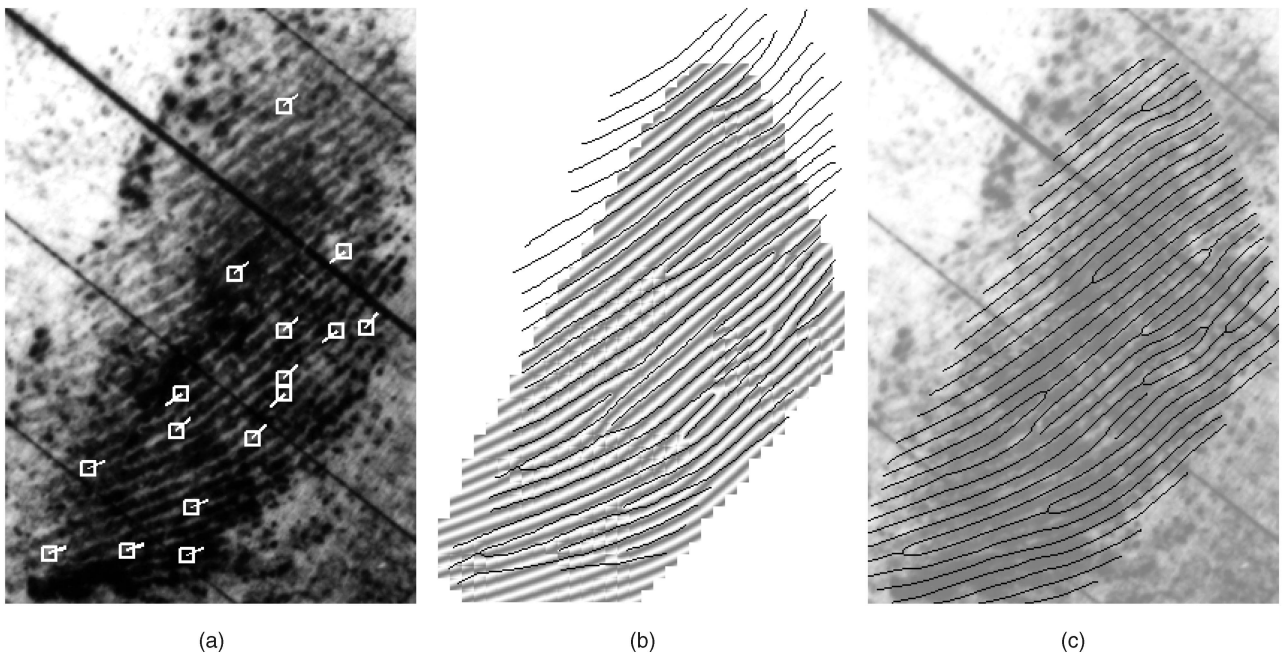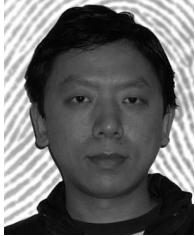
Fig. 18. Reconstruction of latent fingerprint. (a) Latent fingerprint (NIST SD27, G001) with manually marked minutiae, (b) reconstructed fingerprint and the skeleton of the original fingerprint, and (c) the original fingerprint and the skeleton of the reconstructed fingerprint.

REFERENCES

[1]   J. Feng and A.K. Jain, "FM Model Based Fingerprint Reconstruction from Minutiae Template" *Proc. Second Int'l Conf. Biometrics,* pp. 544-553, June 2009.

[2]   A.M. Bazen, G.T.B. Verwaaijen, S.H. Gerez, L.P.J. Veelenturf, and B.J. van der Zwaag, "A Correlation-Based Fingerprint Verification System," *Proc. 11th Ann. Workshop Circuits Systems and Signal Processing,* pp. 205-213, Nov. 2000.

[3]   L.R. Thebaud, "Systems and Methods with Identity Verification by Comparison and Interpretation of Skin Patterns Such as Fingerprints," US Patent No. 5,909,501, 1999.

[4]   J. Feng, Z. Ouyang, and A. Cai, "Fingerprint Matching Using Ridges," *Pattern Recognition,* vol. 39, no. 11, pp. 2131-2140, 2006.

[5]   M. Hara and H. Toyama, "Method and Apparatus for Matching Streaked Pattern Image," US Patent No. 7,295,688, 2007.

[6]   N.K. Ratha, R.M. Bolle, V.D. Pandit, and V. Vaish, "Robust Fingerprint Authentication Using Local Structural Similarity," *Proc. Fifth IEEE Workshop Applications of Computer Vision,* pp. 29-34, 2000.

[7]   A.M. Bazen and S.H. Gerez, "Fingerprint Matching by Thin-Plate Spline Modelling of Elastic Deformations," *Pattern Recognition,* vol. 36, no. 8, pp. 1859-1867, Aug. 2003.

[8]   FVC2004, the Third Int'l Fingerprint Verification Competition, http://bias.csr.unibo.it/fvc2004/, 2010.

[9]   M. Tico and P. Kuosmanen, "Fingerprint Matching Using an Orientation-Based Minutia Descriptor," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 25, no. 8, pp. 1009-1014, Aug. 2003.

[10] J. Feng, "Combining Minutiae Descriptors for Fingerprint Matching," *Pattern Recognition,* vol. 41, no. 1, pp. 342-352, 2008.

[11] K. Asai, H. Izumisawa, K. Owada, S. Kinoshita, and S. Matsuno, "Method and Device for Matching Fingerprints with Precise Minutia Pairs Selected from Coarse Pairs," US Patent No. 4,646,352, 1987.

[12] C. Hill, "Risk of Masquerade Arising from the Storage of Biometrics," master's thesis, Australian Nat'l Univ., 2001.

[13] A. Ross, J. Shah, and A.K. Jain, "From Template to Image: Reconstructing Fingerprints from Minutiae Points," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 4, pp. 544-560, Apr. 2007.

[14] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, "Fingerprint Image Reconstruction from Standard Templates," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 9, pp. 1489-1503, Sept. 2007.

[15] S.O. Novikov and G.N. Glushchenko, "Fingerprint Ridges Structure Generation Models," *Proc. SPIE Int'l Workshop Digital Image Processing and Computer Graphics,* pp. 270-274, 1997.

[16] J.L. Araque, M. Baena, B.E. Chalela, D. Navarro, and P.R. Vizcaya, "Synthesis of Fingerprint Images," *Proc. 16th Int'l Conf. Pattern Recognition,* pp. 422-425, Aug. 2002.

[17] D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition.* second ed. Springer-Verlag, 2009.

[18] B.G. Sherlock and D.M. Monro, "A Model for Interpreting Fingerprint Topology," *Pattern Recognition,* vol. 26, no. 7, pp. 1047-1055, 1993.

[19] W. Bicz, "The Idea of Description (Reconstruction) of Fingerprints with Mathematical Algorithms and History of the Development of This Idea at Optel," Optel, http://www.optel.pl/article/english/idea.htm, 2003.

[20] P.R. Vizcaya and L.A. Gerhardt, "A Nonlinear Orientation Model for Global Description of Fingerprints," *Pattern Recognition,* vol. 29, no. 7, pp. 1221-1231, 1996.

[21] A. Witkin and M. Kass, "Reaction-Diffusion Textures," *ACM SIGGRAPH Computer Graphics,* vol. 25, no. 4, pp. 299-308, 1991.

[22] M. Kücken and A.C. Newell, "Fingerprint Formation," *J. Theoretical Biology,* vol. 235, no. 1, pp. 71-83, 2005.

[23] H. Cummins and M. Midlo, *Finger Prints, Palms and Soles: An Introduction to Dermatoglyphics.* Dover Publications, 1961.

[24] Neurotechnology Inc., VeriFinger, http://www.neurotechnology.com, 2010.

[25] K.G. Larkin and P.A. Fletcher, "A Coherent Framework for Fingerprint Analysis: Are Fingerprints Holograms?" *Optics Express,* vol. 15, pp. 8667-8677, 2007.

[26] D.C. Ghiglia and M.D. Pritt, *Two-Dimensional Phase Unwrapping: Theory, Algorithms, and Software.* John Wiley and Sons, 1998.

[27] J. Bigun and G.H. Granlund, "Optimal Orientation Detection of Linear Symmetry," *Proc. First Int'l Conf. Computer Vision,* pp. 433-438, June 1987.

[28] J. Zhou and J. Gu, "Modeling Orientation Fields of Fingerprints with Rational Complex Functions," *Pattern Recognition,* vol. 37, no. 2, pp. 389-391, 2004.

[29] R.M. Goldstein, H.A. Zebker, and C.L. Werner, "Satellite Radar Interferometry: Two Dimensional Phase Unwrapping," *Radio Science,* vol. 23, no. 4, pp. 713-720, 1988.

[30] K.G. Larkin, "Natural Demodulation of 2D Fringe Patterns," *Proc. Fourth Int'l Workshop Automatic Processing of Fringe Patterns,* 2001.

[31] FVC2002, the Second Int'l Fingerprint Verification Competition, http://bias.csr.unibo.it/fvc2002/, 2010.

[32] NIST Special Database 4, NIST 8-Bit Gray Scale Images of Fingerprint Image Groups (FIGS), http://www.nist.gov/srd/nistsd4.htm, 2010.

[33] K. Nandakumar, A.K. Jain, and S. Pankanti, "Fingerprint-Based Fuzzy Vault: Implementation and Performance," *IEEE Trans. Information Forensics and Security,* vol. 2, no. 4, pp. 744-757, Dec. 2007.

[34] K.A. Nixon and R.K. Rowe, "Multispectral Fingerprint Imaging for Spoof Detection," *Biometric Technology for Human Identification II,* A.K. Jain and N.K. Ratha, eds., pp. 214-225. SPIE, 2005.

[35] NIST Minutiae Interoperability Exchange Test (MINEX), http://fingerprint.nist.gov/minex04/, 2010.

Jianjiang Feng received the BS and PhD degrees from the School of Telecommunication Engineering, Beijing University of Posts and Telecommunications, China, in 2000 and 2007, respectively. From 2008 to 2009, he was a postdoctoral researcher in the Pattern Recognition and Image Processing Laboratory at Michigan State University. He is currently an assistant professor in the Department of Automation at Tsinghua University, Beijing. His research interests include fingerprint recognition, palmprint recognition, and structural matching. He is a member of the IEEE.

Anil K. Jain is a university distinguished professor in the Department of Computer Science and Engineering at Michigan State University. His research interests include pattern recognition and biometric authentication. He received the 1996 *IEEE Transactions on Neural Networks* Outstanding Paper Award and the Pattern Recognition Society Best Paper Awards in 1987, 1991, and 2005. He served as the editor-in-chief of the *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1991-1994). He is a fellow of the AAAS, the ACM, the IEEE, the IAPR, and the SPIE, and a member of the IEEE Computer Society. He has received Fulbright, Guggenheim, Alexander von Humboldt, IEEE Computer Society Technical Achievement, IEEE Wallace McDowell, and IAPR King-Sun Fu Awards. The holder of six patents in the area of fingerprints, he is the author of a number of books, including the *Handbook of Biometrics* (2007), *Handbook of Multibiometrics* (2006), *Handbook of Face Recognition* (2005), *Handbook of Fingerprint Recognition* (2009), *BIOMETRICS: Personal Identification in Networked Society* (1999), and *Algorithms for Clustering Data* (1988). ISI has designated him a highly cited researcher. According to Citeseer, his book *Algorithms for Clustering Data* (Prentice-Hall, 1988) is ranked #93 in the most cited articles in computer science. He currently serves as an associate editor of the *IEEE Transactions on Information Forensics and Security* and the *ACM Transactions on Knowledge Discovery in Data*. He is a member of the Defense Science Board and The National Academies committees on Whither Biometrics and Improvised Explosive Devices.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.