Fingerprint Image Reconstruction from Standard Templates

Raffaele Cappelli, Alessandra Lumini, Dario Maio, *Member*, *IEEE*, and Davide Maltoni, *Member*, *IEEE*

Abstract—A minutiae-based template is a very compact representation of a fingerprint image, and for a long time, it has been assumed that it did not contain enough information to allow the reconstruction of the original fingerprint. This work proposes a novel approach to reconstruct fingerprint images from standard templates and investigates to what extent the reconstructed images are similar to the original ones (that is, those the templates were extracted from). The efficacy of the reconstruction technique has been assessed by estimating the success chances of a masquerade attack against nine different fingerprint recognition algorithms. The experimental results show that the reconstructed images are very realistic and that, although it is unlikely that they can fool a human expert, there is a high chance to deceive state-of-the-art commercial fingerprint recognition systems.

Index Terms—Biometric systems, security, ISO/IEC 19794-2 fingerprint standard template, fingerprint reconstruction, minutiae.

1 Introduction

Fingerprint-based biometric systems are rapidly gaining acceptance as one of the most effective technologies to authenticate users in a wide range of applications: from PC logon to physical access control and from border crossing to voters authentication [22]. A typical fingerprint verification system involves two stages: during *enrollment*, the user's fingerprint is acquired and its distinctive features are extracted and stored as a *template*; and during *verification*, a new fingerprint is acquired and compared to the stored template to verify the user's claimed identity. The distinctive features used by most fingerprint-based systems are the so-called *minutiae*, which are local characteristics of the pattern that are stable and robust to fingerprint impression conditions [22]. With the aim of achieving interoperability among different fingerprint-based recognition systems [24], an international standard for minutiae template representation has been recently defined as ISO/IEC 19794-2 [17], which is a minor modification of the earlier ANSI-INCITS 378-2004 [16].

Since a template based on minutiae is a very compact representation of the fingerprint, many researchers and practitioners in the biometric field postulated that a template does not include enough information to reconstruct the original fingerprint image (that is, the template extraction procedure has been traditionally considered a one-way function). However, this belief was recently questioned by a few works [13], [30] that explored the reversibility of minutiae templates and suggested the possibility of a *masquerade attack*, that is, using a fingerprint image reconstructed from a template for spoofing a recognition system by manufacturing a fake finger [23], [36] or directly injecting the digital image into a communication channel [28].

Considering the growing diffusion of fingerprint-based systems in large-scale applications (for example, electronic identity documents and border crossing [12]) and the efforts toward interoperability (which is certainly desirable), it is definitely urgent to carefully investigate to what extent an image reconstructed from a template may be similar to the original fingerprint. In particular, it is important to understand whether such a reconstructed fingerprint image can fool 1) a human expert examiner and 2) an automatic recognition system through a masquerade attack.

The results of such a research are certainly relevant to all the current applications and case studies where standard template storage and interoperability play an important role such as the PIV program [25], where ANSI-INCITS 378-2004 templates are stored in clear on smart cards, and the ILO Seafarers' Identity Document [15], where the ISO/IEC 19794-2 templates are printed in clear on plastic cards as 2D barcodes.

This work introduces a novel approach to reconstruct fingerprint images starting from minutiae data stored as ISO/IEC 19794-2 templates [17]. The proposed technique is based on a sequence of steps: Starting from the information available in the template, attempt to estimate various aspects of the original unknown fingerprint—the pattern area, the orientation image, and the ridge pattern; then a rendering step is finally executed to make the reconstructed fingerprint more realistic. Although specifically designed to take ISO standard templates as input, this approach may be easily adapted to work with any kind of minutiae-based template (including ANSI-INCITS 378-2004 [16]). The efficacy of the proposed approach has been assessed by estimating the success chances of a masquerade attack against nine different fingerprint recognition algorithms.

The rest of the paper is organized as follows: Section 2 briefly summarizes the previous literature and highlights the novelty of the proposed technique. Section 3 contains basic notation and definitions related to fingerprint analysis and describes the ISO/IEC 19794-2 standard. Section 4 explains the various steps of the new reconstruction approach.

- *R. Cappelli, A. Lumini, and D. Maltoni are with the Scienze dell'Informazione, Università di Bologna, via Sacchi 3, 47023 Cesena (FO), Italy. E-mail: {cappelli, lumini, maltoni}@csr.unibo.it.*
- *D. Maio is with the Department of Electronics, Informatics, and Systems, CSITE CNR, Università di Bologna, viale Risorgimento 2, 40136 Bologna, Italy. E-mail: dmaio@deis.unibo.it.*

Manuscript received 19 May 2006; revised 26 Oct. 2006; accepted 8 Dec. 2006; published online 18 Jan. 2007.
Recommended for acceptance by H. Wechsler.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0386-0506.
Digital Object Identifier no. 10.1109/TPAMI.2007.1087.

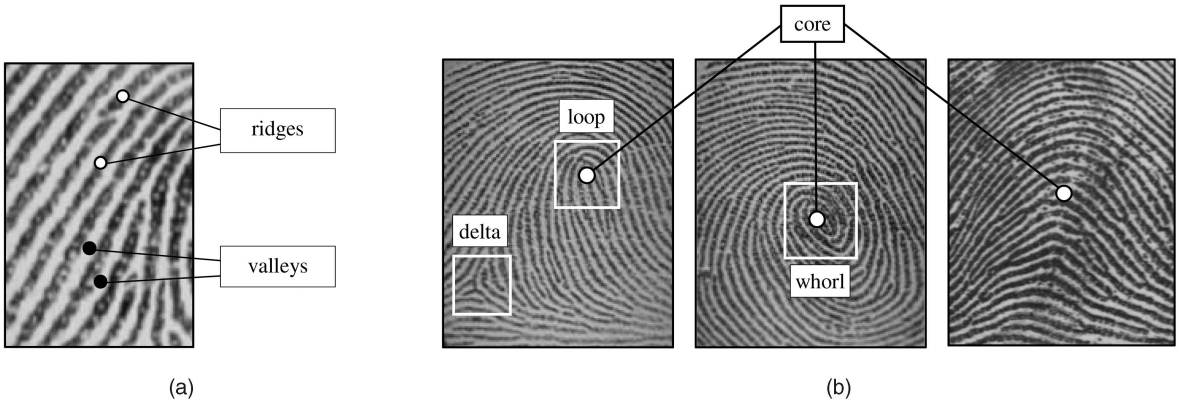


Fig. 1. (a) Ridges and valleys on a fingerprint image. (b) Singular regions (white boxes) and core points (small circles) in fingerprint images.

Section 5 reports both qualitative and quantitative experimental results. Finally, Section 6 draws some conclusions.

2 Related Works

The potential weaknesses of biometric systems have been analyzed in [28], [29], where the possible attacks are classified into four distinct categories:

- attacking the communication channels, including replay attacks on the channel between the sensor and the rest of the system,
- attacking specific software modules (for example, replacing the feature extractor or the matcher with a Trojan horse),
- attacking the database of enrolled templates, and
- presenting fake biometrics to the acquisition sensor.

Recently, the research on the last type of attack has been particularly active in the fingerprint domain. Several experiments aimed at investigating how current fingerprint-based technologies may be spoofed by fake fingers have been performed by various research groups [6], [18], [23], [27], [5] [2], [3], [32], [36]; all of them concluded that, nowadays, no commercial fingerprint scanner seems to be totally resistant to fake fingers made with the appropriate materials (for example, gelatin, silicone, and latex) and proper techniques.

The regeneration of biometric samples (or at least of their main discriminant features), with the aim of attacking a recognition system, has been discussed in various works considering different modalities: brute force attempts [29], hill-climbing attacks [1], [33], and reconstruction from templates. The last modality received less attention in the past, probably because the nonreversibility belief was quite widespread [14]. However, some pioneering works have recently addressed this issue in the fingerprint domain [13], [30] by proposing techniques to estimate the fingerprint class, the orientation image, and to draw a feasible pattern starting from the minutiae data. In [13], a neural network classifier is adopted to predict the fingerprint class; the orientation model proposed in [31] is used to estimate the orientation image, and the reconstructed pattern is created by a heuristic line-tracing approach. The approach reported in [30] estimates the orientation image starting from the minutiae triplets, infers the fingerprint class using a k-nearest neighbor classifier, and generates the final pattern by means of Gabor filters. These two works represent a first step toward the disproof of the nonreversibility belief and suggest that the information contained in fingerprint templates may allow images somewhat similar to the original fingerprints to be reconstructed. Anyway, the results achieved appear still limited: In [13], the reconstruction is performed only on fingerprints with a very simple pattern (those belonging to the arch class); the validation in [30] is mainly qualitative, and in both the above works, the visual quality of the reconstructed image is not much satisfactory.

This work addresses the same problem discussed in [13] and [30], but it substantially differentiates in

1. the template format (ISO/IEC 19794-2) on which the reversibility study is performed,
2. the mathematical models and estimation techniques on which the reconstruction approach is based,
3. the visual quality of the reconstructed fingerprints, and
4. the systematic validation of the new technique against nine different fingerprint matching algorithms.

The novel reconstruction technique was briefly introduced in an earlier work [8], where some examples of the reconstructed images are shown and visually compared to those reported in [13] and [30]; in this work, the whole approach is described, and the results of systematic experiments are reported.

3 Fingerprint Patterns and the ISO Template

3.1 Fingerprint Anatomy

A fingerprint is the reproduction of a fingertip epidermis, which is produced when a finger is pressed against a flat surface. The main structural characteristic of a fingerprint is a pattern of interleaved *ridges* (also called ridgelines) and *valleys* (see Fig. 1a), which often run in parallel. At a global level, fingerprint patterns usually exhibit one or more regions where the ridgelines assume particular shapes (characterized by high curvature, frequent terminations, and so forth). These regions (called *singularities* or *singular regions*) may be classified into three types: *loop*, *delta*, and *whorl* (see Fig. 1b). Singular regions belonging to loop, delta, and whorl types are usually characterized by $\cap$, Δ, and O shapes, respectively. Singular regions are commonly used for fingerprint classification [22] (see Fig. 2), that is, assigning a fingerprint to a class

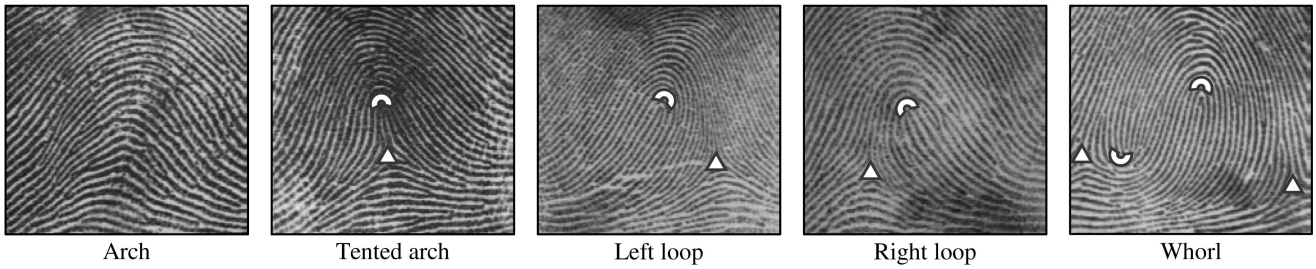


Fig. 2. The five commonly used fingerprint classes: the positions of the singularities are graphically marked.

among a set of distinct classes, with the aim of simplifying search and retrieval.

Fig. 2 shows the five most common classes of the Galton-Henry classification scheme [22] (*Arch*, *Tented arch*, *Left loop*, *Right loop*, and *Whorl*):

- Arch fingerprints have ridges that enter from one side, rise to a small bump, and go out the opposite side: no singularity is present.
- Tented arch fingerprints are similar to the arch, except that some ridgelines exhibit a high curvature, and there are only one loop and one delta (usually vertically aligned).
- Left (right) loop fingerprints have one or more ridges that enter from the left (right) side, curve back, and exit from the same side they entered; a loop and a delta singularity are present: The loop is typically located on the left (right) side of the delta with respect to a vertical axis.
- Whorl fingerprints contain two loop singularities (or a single whorl, which may be considered as two opposite loops at the same location) and two delta singularities; the whorl class is the most complex, and in some classification schemes, it is further divided into some subclasses.

Several fingerprint matching algorithms prealign fingerprint images according to a center point (*core*), typically defined as the position of the northmost loop singularity or as the point of maximum ridgeline curvature for fingerprints belonging to the arch class (Fig. 1b).

The ridgeline pattern may be effectively described by the *orientation image*, which is a discrete matrix whose elements denote the local orientation of the ridgelines (Fig. 3b). The generic element $[x, y]$ of the orientation image is defined as the angle ϕ_{xy} that the tangent to the fingerprint ridges in the corresponding local neighborhood of the image forms with the horizontal axis (Fig. 4). Analogously, the local ridgeline frequency (defined as the number of ridges per unit length) may be effectively represented by using a *frequency image* (Fig. 3c).

At a finer level, other important features called *minutiae* can be found. Minutiae are ridgeline discontinuities and may be classified into several types [22]: termination, bifurcation, island, dot, lake, and so forth. However, to deal with the practical difficulty in automatically discerning them with high accuracy, usually only a coarse classification into two types is adopted (Fig. 5): *termination* (the point where a ridge suddenly ends) and *bifurcation* (the point where a ridge divides into two ridges). A minutia point may be defined by its type, the x and y-coordinates and the direction θ (Figs. 5a and 5b).

Thanks to their stability and high discriminant power, minutiae are the most commonly adopted feature for fingerprint matching [22]. At a very fine level, other details can be also detected: These are essentially the finger sweat pores whose position and shape are considered to be highly distinctive; however, extracting pores is feasible only on high-resolution fingerprint images (for example, 1,000 dots per inch (dpi)) with a very high quality and, therefore, these kinds of features are not adopted in practice.

3.2 Fingerprint ISO Template

The ISO/IEC 19794-2:2005 standard [17] specifies data formats for minutiae-based fingerprint representation. It defines a generic record format that may include one or more templates from one or more finger impressions, and it is designed to be used in a wide range of applications where

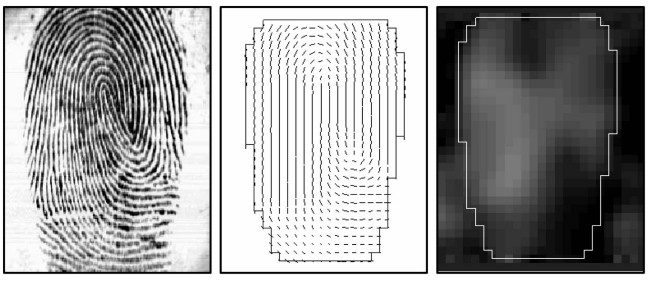


Fig. 3. (b) Orientation image and (c) frequency image of (a) fingerprint, lighter blocks in the frequency image denote regions with a higher frequency.

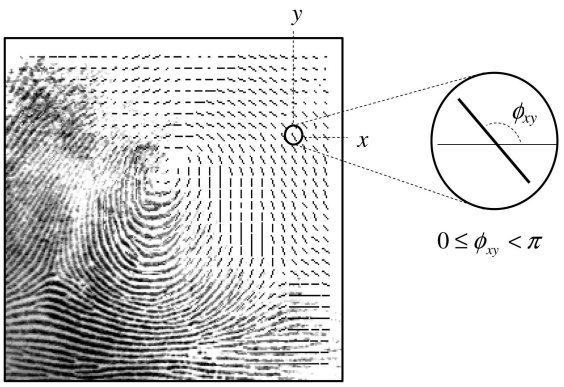
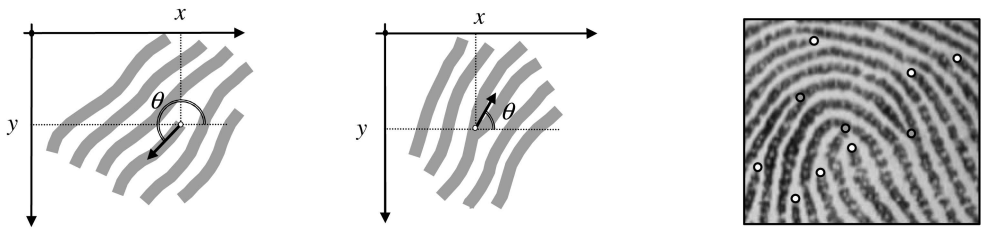


Fig. 4. A fingerprint image faded into the corresponding orientation image.



(a) (b) (c)

Fig. 5. (a) A termination minutia, where (x, y) are the minutia coordinates, θ is defined as the mean direction of the tangents to the two valleys enclosing the termination and is measured increasingly counterclockwise from the horizontal axis to the right. (b) A bifurcation minutia, where θ is defined as the mean direction of the tangents to the two ridgelines enclosing the ending valley and is measured increasingly counterclockwise from the horizontal axis to the right. (c) Terminations (white) and bifurcations (gray) in a sample fingerprint.

TABLE 1
The ISO/IEC 19794-2:2005 Fingerprint Minutiae Record Format

				Field	Size	Valid Values and Notes
Finger Minutiae Record	Record Header			Format ID	4 bytes	'F' 'M' 'R' 0
				...		
				Image Horizontal Size	2 bytes	in pixels
				Image Vertical Size	2 bytes	in pixels
				Horizontal Resolution	2 bytes	in pixels per cm
				Vertical Resolution	2 bytes	in pixels per cm
				Number of Finger Views n_V	1 byte	0 to 255
				...		
	Single Finger Record (n_V instances)	Finger Header		Finger Position	1 byte	0 to 10
				View Number	4 bits	0 to 15
				...		
				Number of Minutiae n	1 byte	0 to 255
		Finger Minutia Record (n instances)		Type	2 bits	{00=other, 01=termination, 10=bifurcation}
				Position x	14 bits	in pixels
				Reserved	2 bits	
				Position y	14 bits	in pixels
				Direction θ	1 byte	0 to 255 (resolution 1.40625 degrees)
				Quality	1 byte	1 to 100 (0=quality not reported)
		Extended Data		Extended Data Block Length	2 bytes	
			(0+ inst.)	Extended Data Area Type Code	2 bytes	only present if Extended Data Block Length>0
				Extended Data Area Length	2 bytes	
				Data Section	(prev. field)	

automated fingerprint recognition is involved. The standard defines the relevant terms, describes how to determine minutiae type, position, and orientation, and specifies the formats to be adopted for storing the data. In particular, three different data formats are defined: A record-based format, which is named *Fingerprint Minutiae Record Format* for general fingerprint template usage and interoperability, and two formats (*normal* and *compact*) for smart cards or other tokens. In this work, only the record-based format has been considered for all the tests since it is the most general template representation available in the standard; anyway, the proposed reconstruction approach may be applied to the other formats without any significant modification.

The Fingerprint Minutiae Record Format defines the fundamental data elements used for minutiae-based representation of a fingerprint and optional extended data formats for including additional data such as ridge counts and singularities location. Table 1 summarizes the structure of the records and the main fields (including all those relevant to the reconstruction approach introduced in this work). A Fingerprint Minutiae Record contains a *Record Header* that includes general information (for example, the image size) and the number of fingerprints (*Finger Views*) represented. For each Finger View, the corresponding *Single Finger Record* contains minutiae data (mandatory) and extended data (optional).

For each minutia, the corresponding *Finger Minutia Record* (6 bytes) contains

- the minutia type (termination, bifurcation, or other), where "other" is defined as a minutia type that may be matched with all the types (hence, it may denote both an unknown type or a type other than termination/bifurcation),

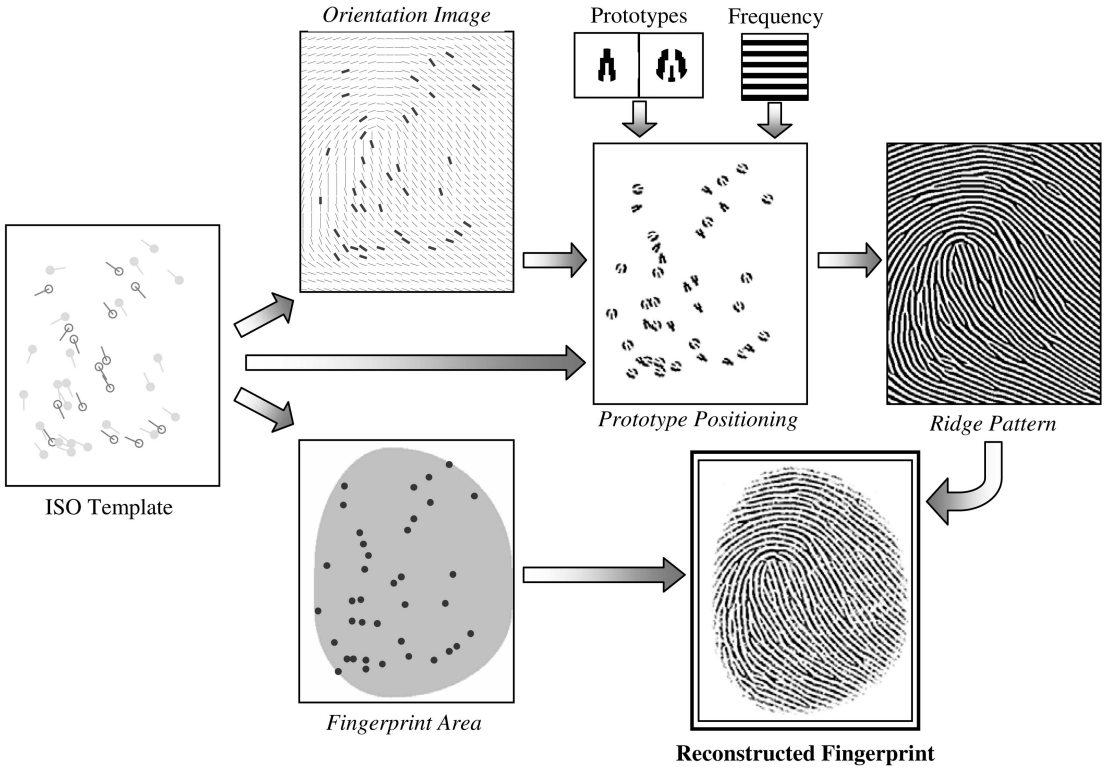


Fig. 6. A functional schema of the proposed reconstruction approach.

- the minutia x and y position expressed in pixels with respect to the coordinate system in Fig. 5,
- the minutia direction θ measured as explained in Fig. 5 and recorded as a single byte in units of 1.40625 (360/256) degrees, and
- the minutia quality in the range 1 (minimum quality) to 100 (maximum quality), or 0 if no quality information is provided.

The *Extended Data* is designed for containing additional information that may be used by the matching algorithm. More than one *Extended Data Record* may be present for each Finger View. For each record, the *Extended Data Area Type Code* denotes whether the data is in a vendor internal format or in one of the following formats defined in the standard:

- Ridge Count Data Format—optional information about the number of fingerprint ridges between pairs of minutiae,
- Core and Delta Data Format—optional information about the placement and characteristics of the loop/whorl[1] and delta singularities (see Section 3.1) on the original fingerprint image, and
- Zonal Quality Data—optional information about the quality of the fingerprint image within each cell in a grid defined on the original fingerprint image.

The approach proposed in this work is able to reconstruct a fingerprint image starting from the minutiae template obtained from a single fingerprint; therefore, for simplicity, Fingerprint Minutiae Records containing only one Finger View will be considered, but the approach could be easily applied to templates containing more Finger Views.

The following record fields are used to reconstruct the fingerprint image:

- Image Horizontal and Vertical Size,
- Image Resolution (for a typical fingerprint scanner, it is reasonable to assume the same horizontal and vertical resolution),
- Number of Minutiae n,
- Type, Position x, y, and Direction θ of each minutia, and
- Core and Delta positions (if provided in the Extended Data).

4 The Reconstruction Approach

Let W and H be the horizontal and vertical size of the image, as specified in the template, respectively, let R be the resolution of the image (in pixel per cm), and let $\mathrm{M} = \{\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_n\}$ be the set of n minutiae in the template, where each minutia is defined as a quadruple $\mathbf{m}_i = \{t_i, x_i, y_i, \theta_i\}$ that indicates its type ($t_i \in \{\mathrm{termination}, \mathrm{bifurcation}, \mathrm{other}\}$), position x_i, y_i (in pixels), and direction θ_i ($0 \leq \theta_i < 2\pi$, converted in radians from the discrete byte value in the template).

The reconstruction approach is based on a sequence of steps that, starting from the information available in the template, attempt to estimate various aspects of the original unknown fingerprint (Fig. 6): the fingerprint area, the orientation image, and the ridge pattern. A final rendering step is executed to make the reconstructed fingerprint image more realistic. The following sections describe the reconstruction steps and the related mathematical models.

1. The standard does not distinguish between loop and whorl singularities and simply names them as "cores."

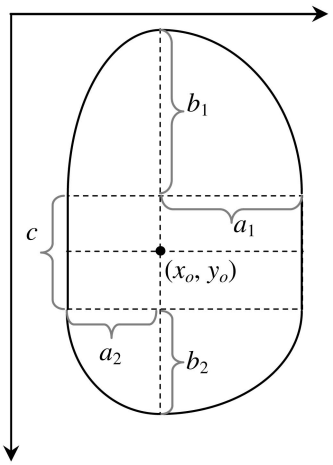


Fig. 7. The fingerprint area model.

4.1 Fingerprint Area

Depending on the finger size, position, and pressure against the acquisition sensor, acquired fingerprint images have different sizes and external shapes; this obviously affects the number and position of the minutiae reported in the template. A simple but effective mathematical model for the fingerprint area has been introduced in [7]: The model, based on four elliptical arcs and a rectangle, is controlled by five parameters (a_1, a_2, b_1, b_2, c) to define the external shape of the area and by two parameters (x_o, y_o) defining the center (see Fig. 7).

In order to estimate the minimal area enclosing all the minutiae in the template, a greedy heuristic algorithm has been adopted to find reasonable values for the model parameters (Fig. 8). The algorithm chooses the center as the centroid of the minutiae set and fixes the value of parameter c to $R/6$, which makes the initial shape slightly elongated in the vertical direction, then, starting from an area containing most of the minutiae, iteratively enlarges it until all the minutiae are enclosed. Finally, the shape is slightly enlarged on all the sides (adding a fixed offset of $R/60$) to guarantee a minimum border around the most external minutiae.

The experimental results (see Fig. 9 for some examples) showed that the area model is appropriate, its degrees of freedom allow the typical fingerprint area shapes to be covered, and the optimization approach is effective. The algorithm is also very fast, being the number of iterations comparable to the number of minutiae. It is worth noting that the model assumes no significant rotation of the fingerprint with respect to the vertical axis; since templates obtained during enrollment have usually a proper positioning, it has not been considered convenient adding this further degree of freedom.

Set $x_o = \frac{1}{n}\sum_{i=1}^{n} x_i$, $y_o = \frac{1}{n}\sum_{i=1}^{n} y_i$, $c = \frac{R}{6}$

Initialize the other parameters as follows:

$a_1 = \max_{i=1..n}\left\{x_i \,|\, |y_i - y_o| \le \frac{c}{2}\right\} - x_o$, $a_2 = x_o - \min_{i=1..n}\left\{x_i \,|\, |y_i - y_o| \le \frac{c}{2}\right\}$

$b_1 = \max\left\{0, \left(y_0 - \frac{c}{2}\right) - \min_{i=1..n}\{y_i\}\right\}$, $b_2 = \max\left\{0, \max_{i=1..n}\{y_i\} - \left(y_0 + \frac{c}{2}\right)\right\}$

while (at least one minutia is external to the area)
{
  Let m_j be the external minutia closer to the centroid (x_o, y_o),
  Let a_h and b_k, $h,k \in \{1, 2\}$, be the semi-axes of the elliptical arc closer to m_j.
  Find $a' \ge a_h$ and $b' \ge b_k$ as follows:
    $(a', b') = FindMinArc(a_h, b_k, m_j)$
  Set $a_h = a'$ and $b_k = b'$
}
Set $a_1 = a_{1+\frac{R}{60}}$, $a_2 = a_{2+\frac{R}{60}}$, $b_1 = b_{1+\frac{R}{60}}$, $b_2 = b_{2+\frac{R}{60}}$

Fig. 8. Pseudocode of the greedy algorithm used to derive appropriate parameters for the model. Function $FindMinArc()$ finds the semiaxes of the elliptical arc with a minimum area that contains m_j: Since minutiae positions and semiaxes are discretized in pixels, a simple exhaustive search is performed.

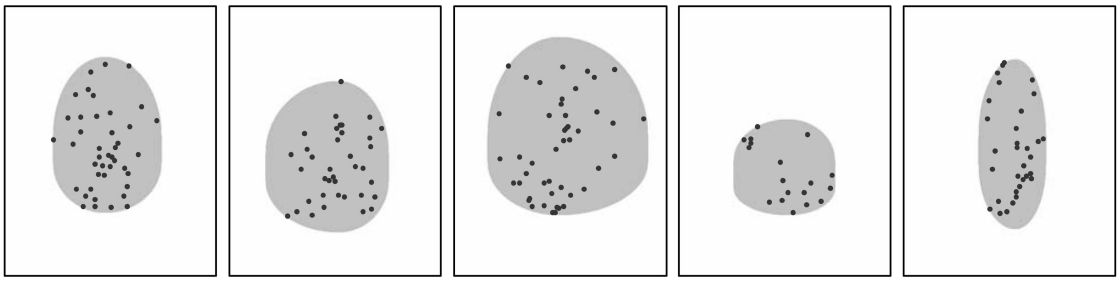


Fig. 9. Fingerprint area as estimated for some minutiae sets.

	(a)	(b)
$\hat{g}_{l_1}(0)$	0	0+π/4
$\hat{g}_{l_1}(1)$	π/4	π/4+π/6
$\hat{g}_{l_1}(2)$	π/2	π/2
$\hat{g}_{l_1}(3)$	3π/4	3π/4
$\hat{g}_{l_1}(4)$	π	π
$\hat{g}_{l_1}(5)$	5π/4	5π/4
$\hat{g}_{l_1}(6)$	3π/2	3π/2
$\hat{g}_{l_1}(7)$	7π/4	7π/4+π/6

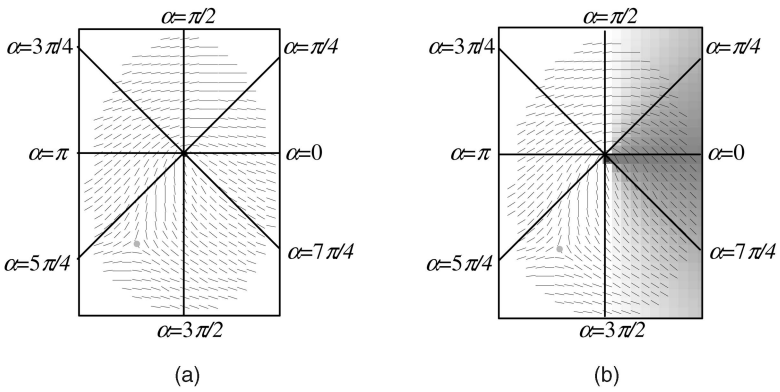


Fig. 10. The effect of modifying some of the control points $\hat{g}_{l_1}(q)$ that define the values of the piecewise linear function $g_{l_1}(\alpha)$ corresponding to the loop singularity of an orientation image obtained with $n_s = 1$; in (b), the orientation differences with respect to (a) are highlighted.

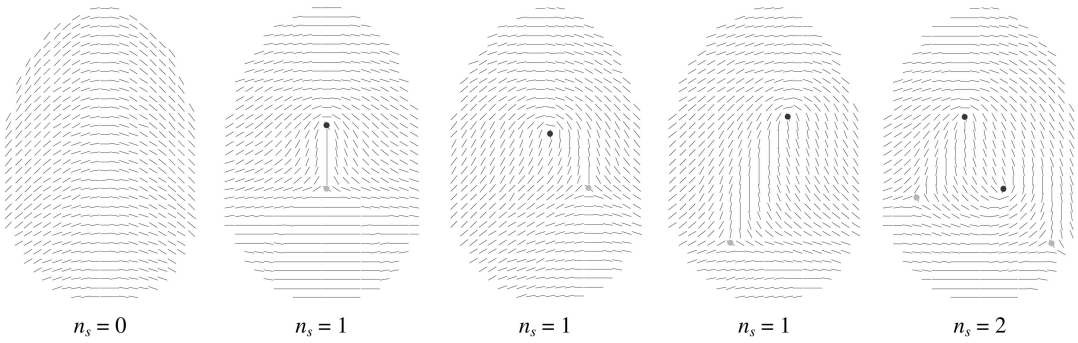


Fig. 11. Examples of orientation images corresponding to the five fingerprint classes obtained with the orientation model adopted.

4.2 Orientation Image

The orientation image defines the overall ridgeline flow that forms the fingerprint pattern; hence, in order to achieve a good reconstruction of the fingerprint, it is crucial to derive an orientation image as much similar as possible to the real one. The only information contained in the ISO template that may help to infer the orientation image are the directions θ of each minutia (see Section 3.2): From the set M of minutiae, it is then possible to obtain partial sparse information on the orientation image (Fig. 6). Deriving a complete orientation image from such partial information is a challenging task: In [30], sets of minutiae triplets are used to estimate the local orientation in triangular fingerprint regions, and a final local averaging step is performed to obtain a smooth result. In this work, in order to make the orientation estimation robust even if the number of minutiae is small or they do not uniformly cover the fingerprint area, a model-based approach is proposed, where some parameters are optimized to fit the minutiae directions in the template.

The orientation model adopted in this work was originally proposed in [34] and extended in [7] to enable the generation of synthetic orientation images. This particular model has been considered better suited to this optimization task among the various ones proposed in the literature [31], [34], [4] [37], [19] since it is able to effectively represent most of the orientation patterns with a small number of parameters. The local orientation is defined as a function of

- the number n_s of loop and delta singularities: Note that, in the typical fingerprint classes, the number of loop and delta singularities is always the same (see Section 3.1),
- the locations of the loop and delta singularities: $\mathbf{l}_j = [lx_j, ly_j]^{\mathrm{T}}$ and $\mathbf{d}_j = [dx_j, dy_j]^{\mathrm{T}}$, $j = 1 \ldots n_s$, and
- other parameters whose number and meaning vary according to n_s (as detailed below).

The orientation ϕ_{xy} at a given point $\mathbf{z} = [x, y]^{\mathrm{T}}$ is defined as

$$\phi_{xy} = \begin{cases} \arctan\left(\max\left\{0, k_{arch} - 3\cdot\frac{y}{\mathrm{H}}\right\}\cdot\cos\left(\frac{x}{\mathrm{W}}\pi\right)\right) & \text{if } n_s{=}0 \text{ (arch class)} \\ \frac{1}{2}\left[\sum_{j=1}^{n_s} g_{d_j}\left(\arg\left(\mathbf{z}-\mathbf{d}_j\right)\right) - \sum_{j=1}^{n_s} g_{l_j}\left(\arg\left(\mathbf{z}-\mathbf{l}_j\right)\right)\right] & \text{if } n_s{=}1 \text{ or } n_s{=}2, \end{cases}$$

where

- the function $arg(\mathbf{p})$ returns the phase angle of vector $\mathbf{p}$ (treated as a complex number),
- k_{arch} is a parameter controlling the curvature for arch class fingerprints, and
- $g_s(\alpha)$, with $s \in \{l_1, l_2, d_1, d_2\}$, are piecewise linear functions defined as

$$g_s(\alpha) = \hat{g}_s(q) + \left(\frac{4\alpha}{\pi} - q\right)(\hat{g}_s((q+1) mod\ 8) - \hat{g}_s(q)),\ q = \left\lfloor \frac{4\alpha}{\pi} \right\rfloor.$$

Each function $g_s(\alpha)$ is defined by the eight control points $\{\hat{g}_s(q) | q = 0 \ldots 7\}$, which set the value of the function for fixed α angles $(\frac{q\pi}{4}, q = 0 \ldots 7)$. Fig. 10 shows the effect of modifying some of the control points $\hat{g}_s(q)$ on a sample orientation image.

Fig. 11 shows an example of orientation image generated by the above model for each of the five fingerprint classes (see Section 3.1).

TABLE 2
Control Points $\hat{g}_s(q)$ as Functions of u_s and v_s for the Different Types of Singularities s

$s = l_1, s = l_2$	$s = d_1, s = d_2$
$\hat{g}_s(0) = u_s$	$\hat{g}_s(0) = \frac{2}{3}u_s$
$\hat{g}_s(1) = \frac{1}{4}\pi + \frac{2}{3}u_s$	$\hat{g}_s(1) = \frac{1}{4}\pi$
$\hat{g}_s(2) = \frac{1}{2}\pi$	$\hat{g}_s(2) = \frac{1}{2}\pi$
$\hat{g}_s(3) = \frac{3}{4}\pi + \frac{2}{3}v_s$	$\hat{g}_s(3) = \frac{3}{4}\pi$
$\hat{g}_s(4) = \pi + v_s$	$\hat{g}_s(4) = \pi + \frac{2}{3}v_s$
$\hat{g}_s(5) = \frac{5}{4}\pi + \frac{2}{3}v_s$	$\hat{g}_s(5) = \frac{5}{4}\pi + v_s$
$\hat{g}_s(6) = \frac{3}{2}\pi$	$\hat{g}_s(6) = \frac{3}{2}\pi + \frac{2}{3}(u_s + v_s)$
$\hat{g}_s(7) = \frac{7}{4}\pi + \frac{2}{3}u_s$	$\hat{g}_s(7) = \frac{7}{4}\pi + u_s$

The set P of unknown parameters that have to be estimated depends on the number of singularities, in particular,

$$\mathrm{P} = \begin{cases} \{k_{arch}\} & \text{if } n_s = 0 \\ \{\mathbf{l}_1, \mathbf{d}_1, u_{l_1}, v_{l_1}, u_{d_1}, v_{d_1}\} & \text{if } n_s = 1 \\ \{\mathbf{l}_1, \mathbf{l}_2, \mathbf{d}_1, \mathbf{d}_2, u_{l_1}, v_{l_1}, u_{l_2}, v_{l_2}, u_{d_1}, v_{d_1}, u_{d_2}, v_{d_2}\} & \text{if } n_s = 2. \end{cases}$$

To reduce the model complexity, for each piecewise linear function $g_s(\alpha)$, P contains just two parameters u_s and v_s: In fact, the eight control points $\hat{g}_s(q)$ for each singularity s are derived from u_s and v_s according to Table 2.

Table 3 reports the ranges of variations for the model parameters.

To make explicit the dependency of the orientation image on the parameters in P, the orientation at a given point $[x, y]^{\mathrm{T}}$ will be denoted here as $\phi_{xy}(\mathrm{P})$.

The estimation of optimal values for the parameters is then carried out through an optimization process by minimizing the cost function $f_{cost}(\mathrm{P})$

$$f_{cost}(\mathrm{P}) = \frac{1}{n}\sum_{i=1}^{n} \mathrm{orientDiff}\big(\theta_i \bmod \pi, \phi_{x_i y_i}(\mathrm{P})\big),$$

where θ_i is the direction of the ith minutia in the template, $\phi_{x_i y_i}(\mathrm{P})$ is the orientation given by the orientation model at the coordinates $[x_i, y_i]^{\mathrm{T}}$ of the ith minutia, and $orientDiff(\varphi_1, \varphi_2) \in \left[0, \frac{\pi}{2}\right]$ evaluates the absolute difference between two orientations:

$$orientDiff(\varphi_1, \varphi_2) = min\{|\varphi_1 - \varphi_2|, \pi - |\varphi_1 - \varphi_2|\},\ \varphi_1, \varphi_2 \in [0, \pi).$$

Three independent minimizations are performed (one for each value of $n_s \in \{0, 1, 2\}$) by using a direct search algorithm: The Nelder-Mead simplex [26]. This particular algorithm has been chosen since it employs only function evaluations and does not rely on derivative information, which would be difficult to calculate due to the complexity of the cost function. The overall minimum among the three n_s values is chosen and the corresponding orientation image is finally postprocessed in order to better fit the minutiae orientations. For each minutia i, the postprocessing consists of replacing $\phi_{x_i y_i}$ with the original minutiae orientation $(\theta_i \bmod \pi)$ and then locally averaging the orientations in a small neighborhood to produce a smooth result.

TABLE 3
Ranges of Variation for the Model Parameters

Parameter	Range
k_{arch}	[2.0, 5.0]
$\mathbf{l}_1, \mathbf{l}_2$	Any position within the $W \times H$ image
$\mathbf{d}_1, \mathbf{d}_2$	
u_s, v_s	$\left[-\frac{2}{3}\pi, \frac{2}{3}\pi\right]$

Fig. 12 compares some reconstructed orientation images with the orientation images calculated from the original fingerprints using the approach proposed in [10].

In case the ISO template contains information about the position of the singularities in the Extended Data (see Section 3.2), the above optimization is simpler; in fact, it is carried out with a fixed value for n_s and with a reduced set of parameters, because estimating the singularity positions $\mathbf{l}_j$ and $\mathbf{d}_j$ is no longer necessary.

4.3 Ridge Pattern

The global characteristics of the ridge pattern may be described by the orientation image and the frequency image (see Section 3.1). Unfortunately, local frequency information is not among the mandatory data required by the ISO template; in case the template contains ridge count information in the Extended Data, an effective technique may be based on the interpolation of the ridge count values between different pairs of minutiae; however, such an additional investigation is outside the aims of this work. On the other hand, reconstructing a frequency image from the minutiae information seems to be almost impracticable: under some simplifying hypothesis, one may try to infer something from the relative position of minutiae pairs and the singularity positions, but usually, the number of minutiae is too low to come to any robust conclusion. For this reason, the approach here proposed simply assumes a constant frequency ν for the whole fingerprint and, instead of attempting to estimate it, reconstructs four fingerprint images with different frequency values in a range determined according to the image resolution R: $\nu = \left(\frac{2.54}{500} R \cdot T\right)^{-1}$, with period $T = 6, 7, 8, 9$ pixels.[2] This range of variation allows to cover typical ridgeline frequencies in nature [22]; adding intermediate values did not lead to better results in our preliminary tests on 500 dpi images.

Given the minutiae set M, the estimated orientation image $[\phi_{xy}]$, and the frequency ν, the ridge pattern reconstruction involves the following steps:

1. minutiae prototype positioning and
2. iterative pattern growing.

Step 1 starts from an empty image and, for each minutia $\mathbf{m}_i = \{t_i, x_i, y_i, \theta_i\}$ in M, places at position (x_i, y_i) a small prototype (that is, a small raster image resembling the characteristics of a minutia) corresponding to minutia type t_i, scaled according to ν and rotated according to

$$\tilde{\theta}_i = \begin{cases} \phi_{x_i y_i} & \text{if } \min\{|\phi_{x_i y_i} - \theta_i|, 2\pi - |\phi_{x_i y_i} - \theta_i|\} < \frac{\pi}{2} \\ \phi_{x_i y_i} + \pi & \text{otherwise,} \end{cases}$$

2. In a 500-dpi image, parameter T is exactly the ridgeline period in pixels.

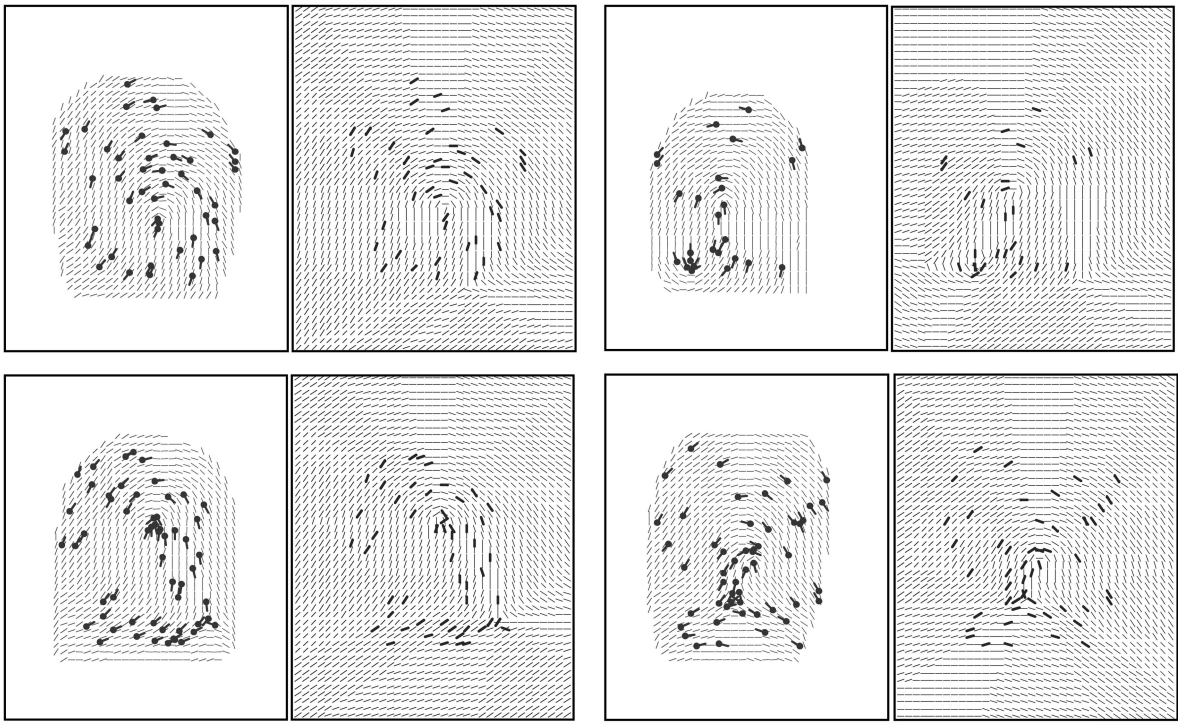


Fig. 12. Four examples of reconstructed orientation images: for each of them, the original orientation image is shown on the left and the reconstructed one on the right; the minutiae orientations are superimposed to both. The orientation images have been obtained starting only from minutiae positions and orientations.

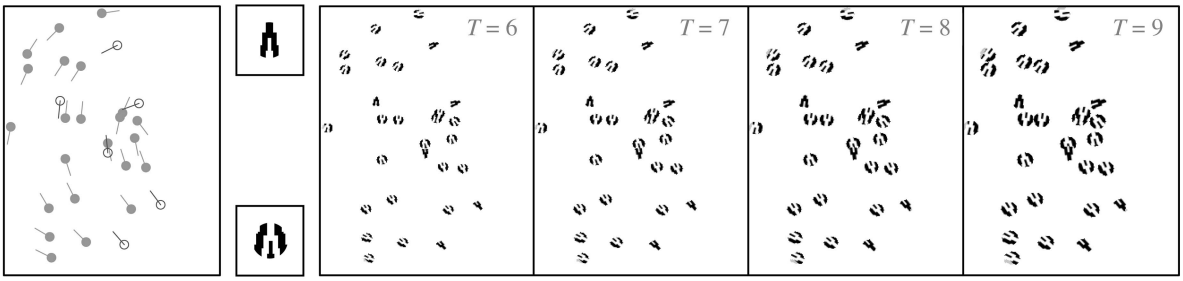


Fig. 13. From left to right: A set of minutiae, the two prototypes (bifurcation and termination), and the four results of Step 1 for different frequencies ν $(T = 6, 7, 8, 9)$.

where angle $\tilde{\theta}_i$ has the estimated orientation $\phi_{x_i y_i}$ and the direction closer to the minutia direction θ_i.

Fig. 13 shows the two prototypes (bifurcation and termination) and some examples of the result of Step 1 for different frequencies ν. In case $t_i = \text{other}$ (minutiae type unknown or not reported), the bifurcation prototype is used.

Step 2 iteratively grows the minutiae prototypes by applying at each pixel (x, y) a Gabor filter adjusted according to the frequency ν and the local orientation ϕ_{xy}:

$$gabor(r, s : \phi_{xy}, \nu) = e^{-\frac{(r+s)^2}{2\sigma^2}} \cdot \cos\left[2\pi\nu(r\sin\phi_{xy} + s\cos\phi_{xy})\right].$$

The parameter σ, which determines the bandwidth of the filter, is adjusted according to the frequency so that the filter does not contain more than three effective peaks (see Fig. 14).

This pattern growing technique is analogous to the approach proposed in [7] for the generation of synthetic ridgeline patterns: At each iteration, the application of the Gabor filters makes the nonempty regions in the image grow (Fig. 15) until they merge generating a uniform ridgeline pattern; the process terminates when the whole image has been covered. An efficient implementation of the filtering can be achieved by using separable Gabor filters, which can be precomputed according to a discretized set of frequencies (4) and orientations (256).

4.4 Rendering

The output of the previous step is a "perfect" pattern with black ridges and white valleys; the noising and rendering step proposed in [7] is finally applied in order to

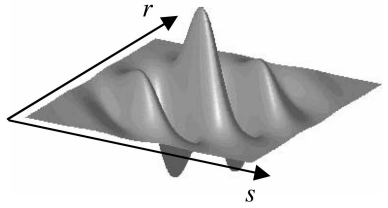


Fig. 14. An example of the Gabor filter used in Step 2: Note that the bandwidth is adjusted so that the filter does not contain more than three peaks.

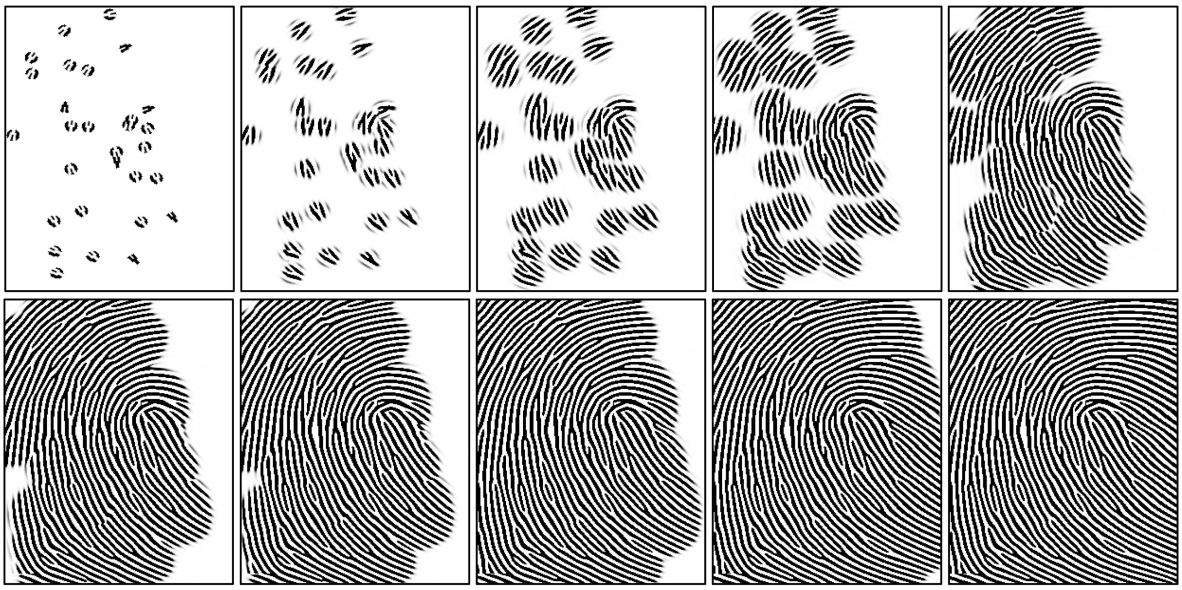


Fig. 15. An example of Step 2 of the ridge pattern reconstruction.

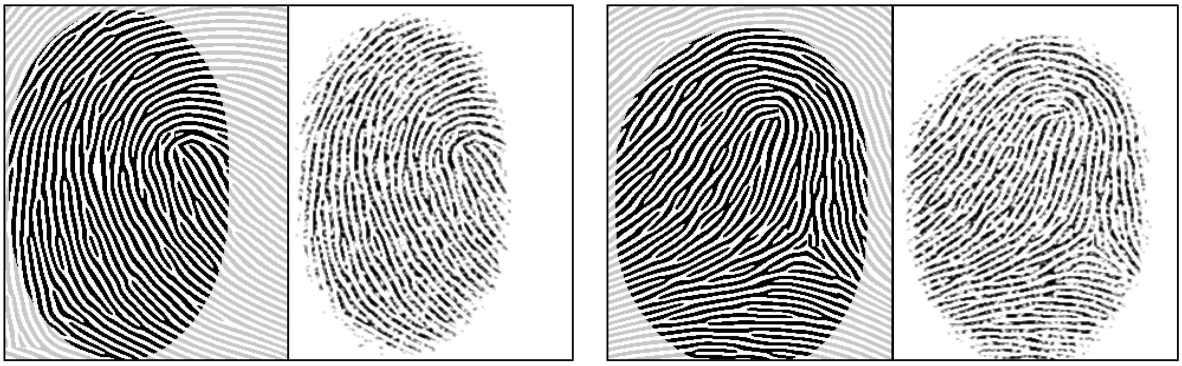


Fig. 16. Two examples of the noising and rendering step.

Fig. 17. Sample fingerprint images from the data set used in the experimentation.

- make the images more realistic to the human eye,
- avoid potential problems that matching algorithms may have when processing images with very sharp edges, and
- prevent automatic algorithms to reject reconstructed images by simply detecting the absence of noise.

The rendering step is aimed at simulating some of the factors that contribute to deteriorate the quality of real fingerprints, thus producing gray-scale noisy images: irregularity of the ridges and their different contact with the sensor surface, presence of small pores within the ridges, presence of very small prominence ridges, gaps, and cluttering noise due to nonuniform pressure of the finger against the sensor, and so forth. The noising and rendering step sequentially performs the following operations:

1. Isolate the valley white pixels into a separate layer.
2. Add noise in the form of small white blobs of variable size and shape. The amount of noise increases with the inverse of distance from the boundary of the fingerprint area (estimated as explained in Section 4.1).
3. Smooth the resulting image with a 3×3 averaging box filter.
4. Superimpose the valley layer to the image obtained.
5. Remove any part of the pattern lying outside the fingerprint area.

Fig. 16 shows two examples of the noising and rendering step. It is worth noting that the image background is completely white: this imitates the output of a clean optical sensor such as the one used in the experimentation (see Fig. 17); the background-generation approach proposed in [7] may be used to add sensor-specific backgrounds.

Fig. 18. From left to right: an original fingerprint, a reconstructed fingerprint from the corresponding ISO template, and an overlay of the two images. An animation that better highlights the similarities between an original and a reconstructed image is available in the supplemental material, which can be found at http://computer.org/tpami/archives.htm.

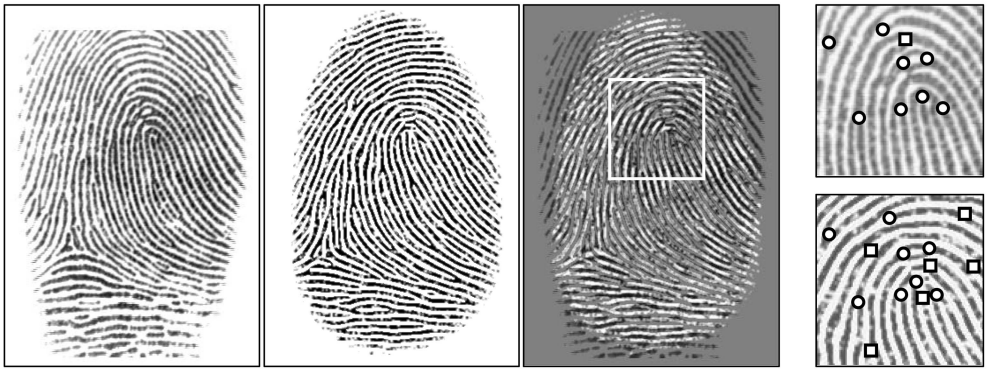


Fig. 19. The comparison of an original and a reconstructed image: minutiae in the core region have been manually marked (circles and squares denote matching and nonmatching minutiae, respectively).

5 Experimental Results

The proposed fingerprint reconstruction approach has been evaluated on fingerprint images (388×374 pixels) acquired through a 500 dpi optical scanner during the collection of Second International Competition for Fingerprint Verification Algorithms (FVC 2002) database 1 (DB1) [11]. FVC2002 data collection involved 30 volunteers in three different sessions [21]: During the first one, the volunteers were asked to place their fingers over the acquisition sensor "naturally," whereas during the other two, specific perturbations were added (exaggerated displacement and rotation, and moistened and dried fingers). For each volunteer, fingerprints from four different fingers (forefinger and middle finger of each hand) were acquired. In order to create a data set containing fingerprint images as much similar as possible to those typically acquired during the enrollment stage of a generic application, the first impression of each finger acquired at the first session has been selected, thus obtaining 120 different fingerprints. Fig. 17 shows some sample fingerprints from the data set. For each fingerprint, an optimized version of the minutiae extraction algorithm described in [20] has been adopted to create a corresponding ISO template, which has been used as input for reconstructing four fingerprint images (with different frequency values; see Section 4.3) using the proposed approach.

In Section 5.1, some reconstructed fingerprint images are compared to the original ones and visually analyzed, and then the results are discussed. In Section 5.2, the effectiveness of the proposed approach is validated by simulating attacks against state-of-the-art fingerprint matching algorithms.

5.1 Examples of Reconstructed Images

Fig. 18 compares an original fingerprint with one reconstructed from the corresponding ISO template: The similarity between the two images is high. The two ridgeline patterns are extremely close in most of the common area, as it may be observed in the overlay image. It is worth noting that the reconstruction has been performed starting only from the minutiae data; no additional information has been considered (for example, position of the singularities). This means that the orientation model and the optimization procedure introduced in Section 4.2 have been able to properly reconstruct the orientation image.

A further analysis shows that the results are good also at a minutiae level: Most of the original minutiae are present in the reconstructed image with the correct position and orientation, although the reconstructed image often contains more minutiae than the original one. Figs. 19 and 20 report two examples where the minutiae in the core region have been manually marked and matched. In Fig. 19, the number of matching minutiae is eight (over nine in the

Fig. 20. The comparison of an original and a reconstructed image: Minutiae in the core region have been manually marked (circles and squares denote matching and nonmatching minutiae, respectively).

original fingerprint and 14 in the reconstructed one); in Fig. 20, the number of matching minutiae is three (over four in the original fingerprint and 13 in the reconstructed one). The worse result of Fig. 20 is probably due to the low number of minutiae in the ISO template, which caused a wrong estimation of the orientation image in the core region and, as a consequence, a quite different ridgeline pattern generated. However, it should be noted that, even in this case, most of the original minutiae find a good match in the reconstructed image.

In spite of the high similarity of the reconstructed fingerprint patterns with respect to the original images, at a fine level of detail, several differences exist between the patterns: this is due not only to the extra minutiae inserted in the reconstructed images but also to some details like the local shape of the minutiae, the presence of evident pores, the presence of scratches or other imperfections, the structure around the core region (which is very characteristic), and so forth.

5.2 Attacking Automatic Fingerprint Recognition Systems

This section reports the results of experiments aimed at exploring the feasibility of a masquerade attack against eight state-of-the-art commercial fingerprint recognition algorithms[3] (referred to in the following as $\mathrm{A1}, \mathrm{A2}, \ldots, \mathrm{A8}$) and against the algorithm available in the National Institute of Standards and Technology (NIST) Fingerprint Image Software 2 (NFIS2) [35] (referred to in the following as NIST). To the best of our knowledge, all the eight commercial algorithms (whose implementation details are industrial secrets) use minutiae as the main feature; on the other hand, it is likely that they also exploit other features to improve the performance [9].

For each algorithm:

- Three operating thresholds τ have been selected to force the algorithm to operate at different security levels, corresponding to False Match Rate $(\mathrm{FMR}) = 1$ percent, $\mathrm{FMR} = 0.1$ percent, and $\mathrm{FMR} = 0$ percent; to this purpose, the above error rates have been a priori computed over the whole FVC2002 DB1 according to the FVC2002 protocol [11].
- The 120 templates corresponding to the original fingerprint images in the data set have been created (the internal template format of the algorithm has been used, which does not necessarily correspond to the ISO template and may contain additional data).
- The four fingerprint images, reconstructed from each ISO template with different frequency values, have been matched against the corresponding template created by the algorithm: The attack has been considered successful if at least one of the four reconstructed images obtained a matching score higher than τ.
- The percentage of successful attacks over the 120 fingerprints has been reported for the three security levels ($\mathrm{FMR} = 1$ percent, $\mathrm{FMR} = 0.1$ percent, and $\mathrm{FMR} = 0$ percent).

The attacks have been simulated under the following hypotheses:

- BASE. Only mandatory fields are present in the ISO template, and no minutiae type information is available $(t_i = \text{other for each } i)$.
- MINTYPE. Only mandatory fields are present in the ISO template, but minutiae type information is available $(t_i \in \{\text{bifurcation}, \text{termination}\}$ for each $i)$.
- SINGPOS. Same as BASE, but with additional information about Core and Delta positions in the Extended Data.
- ORIMG. Same as BASE, but with the whole orientation image stored in the Extended Data using a vendor internal format; in this case, the orientation image estimation described in Section 4.2 is not performed, and the original orientations are used to guide the ridge pattern generation (Section 4.3).

Tables 4, 5, 6, and 7 report the results obtained under the four above hypotheses; Fig. 21 highlights the worst/average/best performance at each security level.

The percentage of successful attacks from fingerprints reconstructed with the proposed approach is surprising and beyond all expectations. From the analysis of the results in the four different hypotheses, the following observations may be made:

- With the sole knowledge of the minutiae position and orientation (BASE hypothesis) at a security level of

3. The names of the commercial systems tested are not disclosed here to avoid any form of undesired publicity.

TABLE 4
Percentage of Successful Attacks under the BASE Hypothesis

BASE		Algorithms									Average
		A1	A2	A3	A4	A5	A6	A7	A8	NIST	
Security levels	FMR=1.0%	100%	100%	100%	86.67%	98.33%	100%	100%	81.67%	91.67%	95.37%
	FMR=0.1%	100%	95.83%	98.33%	80.83%	88.33%	95.83%	100%	76.67%	79.17%	90.56%
	FMR=0.0%	97.50%	84.17%	97.50%	68.33%	80%	86.67%	88.33%	64.17%	66.67%	81.49%

TABLE 5
Percentage of Successful Attacks under the MINTYPE Hypothesis

MINTYPE		Algorithms									Average
		A1	A2	A3	A4	A5	A6	A7	A8	NIST	
Security levels	FMR=1.0%	100%	98.33%	99.17%	86.67%	95.83%	98.33%	100%	79.17%	95.00%	94.72%
	FMR=0.1%	100%	93.33%	99.17%	74.17%	83.33%	96.67%	98.33%	73.33%	87.50%	89.54%
	FMR=0.0%	100%	80%	99.17%	67.50%	72.50%	88.33%	81.67%	57.50%	67.50%	79.35%

TABLE 6
Percentage of Successful Attacks under the SINGPOS Hypothesis

SINGPOS		Algorithms									Average
		A1	A2	A3	A4	A5	A6	A7	A8	NIST	
Security levels	FMR=1.0%	100%	98.33%	98.33%	94.17%	95%	97.50%	100%	91.67%	91.67%	96.30%
	FMR=0.1%	99.17%	91.67%	96.67%	87.50%	87.50%	93.33%	97.50%	86.67%	83.33%	91.48%
	FMR=0.0%	96.67%	80.83%	95.83%	79.17%	77.50%	86.67%	93.33%	69.17%	70.00%	83.24%

TABLE 7
Percentage of Successful Attacks under the ORIMG Hypothesis

ORIMG		Algorithms									Average
		A1	A2	A3	A4	A5	A6	A7	A8	NIST	
Security levels	FMR=1.0%	100%	100%	100%	100%	99.17%	100%	100%	99.17%	98.33%	99.63%
	FMR=0.1%	99.17%	100%	100%	99.17%	95%	100%	100%	99.17%	98.33%	98.98%
	FMR=0.0%	96.67%	100%	100%	97.50%	94.17%	100%	100%	94.17%	97.50%	97.78%

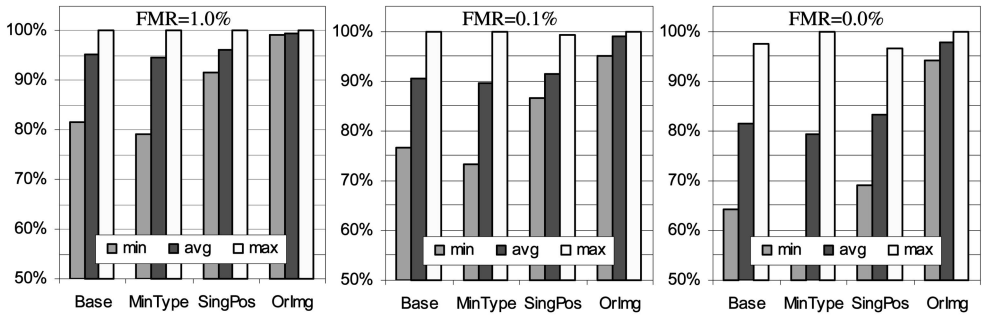


Fig. 21. A bar graph for each security level: each graph shows the minimum, average, and maximum result obtained by the nine matching algorithms for each of the four hypotheses.

0.1 percent FMR (typical of medium-security applications), the average percentage of successful attacks is higher than 90 percent, and each algorithm tested accepts at least 75 percent of the reconstructed fingerprints. At a much higher security level (corresponding to 0 percent FMR measured on FVC2002 DB1), the average percentage of successful attacks is higher than 80 percent.

- Contrary to what one may expect, knowing minutiae type information (MINTYPE hypothesis) seems not to improve the chance of a successful attack: in fact, although in some cases the percentage of success grows, for most of the algorithms, it drops. This may be explained by considering that, on one hand, this information does not add any benefit against those algorithms (probably, the vast majority) that do not

consider minutiae type information during matching, whereas, on the other hand, the termination prototype is more complex than the bifurcation one (see Fig. 13), and it may lead to an excessive proliferation of minutiae during the ridgeline pattern iterative growth. Considering all minutiae types as "unknown" and, thus, using always the bifurcation prototype (as explained in Section 4.3) may help to reconstruct patterns that have a higher chance to be accepted by most algorithms.

- The knowledge of the singularity positions (SINGPOS hypothesis) increases the average percentage of successful attacks only slightly: this indicates that the proposed optimization approach for estimating the orientation image is effective even when such information is not available.
- If the whole fingerprint orientation image is available in the template (ORIMG hypothesis), the probability of a successful attack drastically increases: An average of 97.78 percent has been achieved at the highest security level considered. It is important to remind that, in this experimentation, each algorithm adopted its internal template format, which may contain additional data with respect to the ISO standard; the extremely good results obtained in the ORIMG hypothesis may suggest that some of the algorithms added information about the whole orientation image or part of it to their templates.

6 Conclusions

The most important aim of this research was to study the reconstruction of fingerprint images from standard templates in order to understand whether it is possible

1. to fool a human expert (for instance placing a reconstructed fingerprint image on a crime scene) and
2. to perform masquerade attacks against a state-of-the-art automatic fingerprint recognition system (for instance, injecting a reconstructed image in a communication channel or manufacturing a fake finger).

From the novel reconstruction method developed and the results of the systematic experiments carried out, the following conclusions may be drawn:

1. It is very unlikely to fool a human expert: this does not seem to be simply due to a limitation of the proposed approach (for example, the generation of a relatively high number of minutiae) but rather to a lack of information in the standard template itself, which does not allow details such as the local shape of the minutiae or evident pores to be reconstructed.
2. It is definitely possible to successfully attack state-of-the-art automatic recognition systems, provided that one is able to present reconstructed images to the system. In the experiments performed starting from the sole minutiae positions and orientations (mandatory in the ISO template), the average percentage of successful attacks against nine different systems was 81 percent at a high security level and 90 percent at a medium security level.

The latter result points out the need for new efforts aimed at making current systems

- more robust against fake fingers placed on the acquisition sensor,
- adequately protected along all the communication channels (for example, by encryption and challenge response techniques), and
- aware of the feasibility of such masquerade attacks and, hence, equipped with appropriate countermeasures. For instance, since a reconstructed image usually contains more minutiae than the original one, in the presence of high-quality minutiae with no mate in the template, the matching score might be considerably decreased.

The proposed reconstruction approach could be further improved to make the success probability of a masquerade attack even higher. Some of the possible enhancements are

- adopting more effective optimization algorithms and models for estimating the orientation image (which the experimental results are confirmed to be fundamental for the reconstruction),
- exploiting ridge-count information (when available) to estimate the local frequency, and
- attempting to remove the spurious minutiae added during the ridge pattern generation.

However, the above (or other) improvements are outside the aims of this work, whose main purpose is not to design a perfect way to fool biometric systems, but to encourage developers of algorithms and systems to seriously take into account this kind of attack and to implement specific protections and countermeasures.

Acknowledgments

This work was partially supported by European Commission (BioSecure-FP6 IST-2002-507634).

References

[1] A. Adler, "Can Images Be Regenerated from Biometric Templates," *Proc. Biometrics Consortium Conf.,* Sept. 2003.
[2] A. Antonelli, R. Cappelli, D. Maio, and D. Maltoni, "Fake Finger Detection by Skin Distortion Analysis," *IEEE Trans. Information Forensics and Security,* vol. 1, no. 3, pp. 360-373, Sept. 2006.
[3] D. Baldisserra, A. Franco, D. Maio, and D. Maltoni, "Fake Fingerprint Detection by Odor Analysis," *Proc. Int'l Conf. Biometric Authentication,* Jan. 2006.
[4] A.M. Bazen and S.H. Gerez, "Systematic Methods for the Computation of the Directional Fields and Singular Points of Fingerprints," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 7, pp. 905-919, July 2002.
[5] BioSec European Research Project-FP6 IST-2002-001766, http://www.biosec.org, 2005.
[6] J. Blommé, "Evaluation of Biometric Security Systems against Artificial Fingers," master's thesis, 2003.
[7] R. Cappelli, "Synthetic Fingerprint Generation," *Handbook of Fingerprint Recognition,* D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar, eds., Springer, 2003.
[8] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, "Can Fingerprints be Reconstructed from ISO Templates," *Proc. Ninth Int'l Conf. Control, Automation, Robotics and Vision,* Dec. 2006.
[9] R. Cappelli, D. Maio, D. Maltoni, J.L. Wayman, and A.K. Jain, "Performance Evaluation of Fingerprint Verification Systems," *IEEE Trans. Pattern Analysis Machine Intelligence,* vol. 28, no. 1, pp. 3-18, Jan. 2006.
[10] M. Donahue and S. Rokhlin, "On the Use of Level Curves in Image Analysis," *Image Understanding,* vol. 57, no. 3, pp. 185-203, 1993.
[11] *Proc. Second Int'l Competition for Fingerprint Verification Algorithms (FVC 2002),* http://bias.csr.unibo.it/fvc2002, 2002.

[12] US General Accounting Office, "Using Biometrics for Border Security," Technical Report GAO-03-174, Government Accountability Office, 2002.
[13] C. Hill, "Risk of Masquerade Arising from the Storage of Biometrics," master's thesis, Australian Nat'l Univ., 2001.
[14] Int'l Biometric Group, "Generating Images from Templates," white paper, IBG, 2002.
[15] ILO SID-0002, "Finger Minutiae-Based Biometric Profile for Seafarers' Identity Documents," Int'l Labour Organization, 2006.
[16] ANSI-INCITS 378-2004, *Information Technology—Finger Minutiae Format for Data Interchange,* 2004.
[17] ISO/IEC 19794-2:2005, *Information Technology—Biometric Data Interchange Formats—Part 2: Finger Minutiae Data,* 2005.
[18] H. Kang, B. Lee, H. Kim, D. Shin, and J. Kim, "A Study on Performance Evaluation of the Liveness Detection for Various Fingerprint Sensor Modules," *Proc. Seventh Int'l Conf. Knowledge-Based Intelligent Information and Engineering Systems,* pp. 1245-1253, 2003.
[19] J. Li, W.Y. Yau, and H. Wang, "Constrained Nonlinear Models of Fingerprint Orientations with Prediction," *Pattern Recognition,* vol. 39, no. 1, pp. 102-114, 2006.
[20] D. Maio and D. Maltoni, "Direct Gray-Scale Minutiae Detection in Fingerprints," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 1, pp. 27-40, Jan. 1997.
[21] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, and A.K. Jain, "Second Int'l Competition for Fingerprint Verification Algorithms (FVC 2002)," *Proc. 16th Int'l Conf. Pattern Recognition,* vol. 3, pp. 811-814, Aug. 2002.
[22] D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition.* Springer, 2003.
[23] T. Matsumoto, H. Matsumoto, K. Yamada, and S. Hoshino, "Impact of Artificial 'Gummy' Fingers on Fingerprint Systems," *Proc. Int'l Soc. Optical Eng. (SPIE),* vol. 4677, Jan. 2002.
[24] *NIST Minutiae Interoperability Exchange Test (MINEX),* http://fingerprint.nist.gov/minex, 2006.
[25] NIST Special Publication 800-76, "Biometric Data Specification for Personal Identity Verification," Feb. 2005.
[26] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge Univ. Press, 1988.
[27] T. Putte and J. Keuning, "Biometrical Fingerprint Recognition: Don't Get Your Fingers Burned," *Proc. IFIP TC8/WG8.8 Fourth Working Conf. Smart Card Research and Advanced Applications,* pp. 289-303, 2000.
[28] N.K. Ratha, J.H. Connell, and R.M. Bolle, "Enhancing Security and Privacy in Biometrics-Based Authentication Systems," *IBM Systems J.,* vol. 40, no. 3, pp. 614-634, 2001.
[29] N.K. Ratha, J.H. Connell, and R.M. Bolle, "An Analysis of Minutiae Matching Strength," *Proc. Third Int'l Conf. Audio and Video-Based Biometric Person Authentication,* pp. 223-228, 2001.
[30] A. Ross, J. Shah, and A.K. Jain, "Toward Reconstructing Fingerprints from Minutiae Points," *Proc. Int'l Soc. Optical Eng. (SPIE), Biometric Technology for Human Identification II,* A.K. Jain and N.K. Ratha, eds., pp. 68-80, Mar. 2005.
[31] B. Sherlock and D. Monro, "A Model for Interpreting Fingerprint Topology," *Pattern Recognition,* vol. 26, no. 7, pp. 1047-1055, 1993.
[32] L. Thalheim and J. Krissler, "Body Check: Biometric Access Protection Devices and Their Programs Put to the Test," *c't Magazine,* Nov. 2002.
[33] U. Uludag and A.K. Jain, "Attacks on Biometric Systems: A Case Study in Fingerprints," *Proc. Int'l Soc. Optical Eng. (SPIE), Security, Steganography, and Watermarking of Multimedia Contents VI,* E.J. Delp III and P.W. Wong, eds., pp. 622-633, June 2004.
[34] P. Vizcaya and L. Gerhardt, "A Nonlinear Orientation Model for Global Description of Fingerprints," *Pattern Recognition,* vol. 29, no. 7, pp. 1221-1231, 1996.
[35] C. Watson and M. Garris, "NIST Fingerprint Image Software 2 (NFIS2)," Nat'l Inst. of Standards and Technology, http://fingerprint.nist.gov/NFIS, 2006.
[36] A. Wiehe, T. Sondrol, O.K. Olsen, and F. Skarderud, "Attacking Fingerprint Sensors," NISlab/Gjovik Univ. College, technical report, Dec. 2004, http://olekasper.no/articles/attacking_fingerprint_sensors.pdf.
[37] J. Zhou and J. Gu, "Modeling Orientation Fields of Fingerprints with Rational Complex Functions," *Pattern Recognition,* vol. 37, no. 2, pp. 389-391, 2004.

Raffaele Cappelli received the Laurea degree (cum laude) in computer science from the University of Bologna, Italy, in 1998 and the PhD degree in computer science and electronic engineering from the Department of Electronics, Informatics, and Systems (DEIS), University of Bologna, in 2002. He is an associate researcher at the University of Bologna. His research interests include pattern recognition, image retrieval by similarity, and biometric systems (fingerprint classification and recognition, synthetic fingerprint generation, face recognition, and performance evaluation of biometric systems).

Alessandra Lumini received the Laurea degree (cum laude) in computer science from the University of Bologna, Italy, in 1996 and the PhD degree in computer science and electronic engineering from the Department of Electronics, Informatics, and Systems (DEIS), University of Bologna, in 2001, for her work on image databases. She is an associate researcher at the II Faculty of Engineering of the University of Bologna. Her research interests include pattern recognition, biometric systems, image databases, and multidimensional data structures.

Dario Maio received the degree in electronic engineering from the University of Bologna in 1975. He is a full professor at the University of Bologna. He is the chair of the Cesena Campus and the director of the Biometric Systems Laboratory (Cesena, Italy). He has published more than 150 papers in numerous fields, including distributed computer systems, computer performance evaluation, database design, information systems, neural networks, autonomous agents, and biometric systems. He is the author of the books *Biometric Systems, Technology, Design and Performance Evaluation* (Springer, 2005) and *Handbook of Fingerprint Recognition* (Springer, 2003), received the PSP award from the Association of American Publishers. Before joining the University of Bologna, he received a fellowship from the Italian National Research Council (CNR) for working on the Air Traffic Control Project. He is a member of the IEEE. He is with the Department of Electronics, Informatics, and Systems (DEIS) and IEIIT-CNR. He teaches database and information systems.

Davide Maltoni is an associate professor at the Department of Electronics, Informatics, and Systems (DEIS), University of Bologna. He teaches "computer architectures" and "pattern recognition" at the Department of Computer Science, University of Bologna, Cesena, Italy. His research interests are in the area of Pattern Recognition and Computer Vision. In particular, he is active in the field of Biometric Systems (fingerprint recognition, face recognition, hand recognition, and performance evaluation of biometric systems). He is a codirector of the Biometric Systems Laboratory (Cesena, Italy), which is internationally known for its research and publications in the field. He is the author of the books *Biometric Systems, Technology, Design and Performance Evaluation* (Springer, 2005) and *Handbook of Fingerprint Recognition* (Springer, 2003), which received the PSP award from the Association of American Publishers. He is currently an associate editor of the journals *Pattern Recognition* and *IEEE Transactions on Information Forensic and Security*. He is a member of the IEEE.