3.6.5 Miscellanea

Other fingerprint enhancement approaches can be found in Asai et al. (1975), Berdan and Chiralo (1978), Nakamura, Nagaoka, and Minami (1986), Danielsson and Ye (1988), Sasakawa, Isogai, and Ikebata (1990), Kaymaz and Mitra (1993), Mehtre (1993), Bergengruen (1994), Sherstinsky and Picard (1994, 1996), Szu et al. (1995), Almansa and Lindeberg (1997), Pradenas (1997), Park and Smith (2000), Greenberg et al. (2000), Ghosal et al. (2000a, b), Simon-Zorita et al. (2001a, b), Cheng, Tian, and Zhang (2002), Connell, Ratha, and Bolle (2002), Kim, Kim, and Park (2002), Cheng et al. (2003), Pattichis and Bovik (2004), Wu, Shi, and Govindaraju (2004), Bal, El-Saba, and Alam (2005a), Greenberg and Kogan (2005), Chen and Dong (2006), Chen et al. (2006b) and Rahmes et al. (2007).

3.7 Minutiae Detection

Most automatic systems for fingerprint comparison are based on minutiae matching (see Chapter 4); hence, reliable minutiae extraction is an extremely important task and a substantial amount of research has been devoted to this topic. Most of the proposed methods require the fingerprint gray-scale image to be converted into a binary image. Some binarization processes greatly benefit from an a priori enhancement (see Section 3.6); on the other hand, some enhancement algorithms directly produce a binary output, and therefore the distinction between enhancement and binarization is sometimes faded. The binary images are usually submitted to a thinning stage which allows for the ridge line thickness to be reduced to one pixel, resulting in a skeleton image (Figure 3.39). A simple image scan then allows the detection of pixels that correspond to minutiae.

Some authors have proposed minutiae extraction approaches that work directly on the gray-scale images without binarization and thinning. This choice is motivated by the following considerations:

- A significant amount of information may be lost during the binarization process.
- Binarization and thinning are time consuming; thinning may introduce a large number of spurious minutiae.
- In the absence of an a priori enhancement step, most of the binarization techniques do not provide satisfactory results when applied to low-quality images.

3.7.1 Binarization-based methods

The general problem of image binarization has been widely studied in the fields of image processing and pattern recognition (Trier and Jain, 1995). The simplest approach uses a *global*

threshold t and works by setting the pixels whose gray-level is lower than *t* to 0 and the remaining pixels to 1.

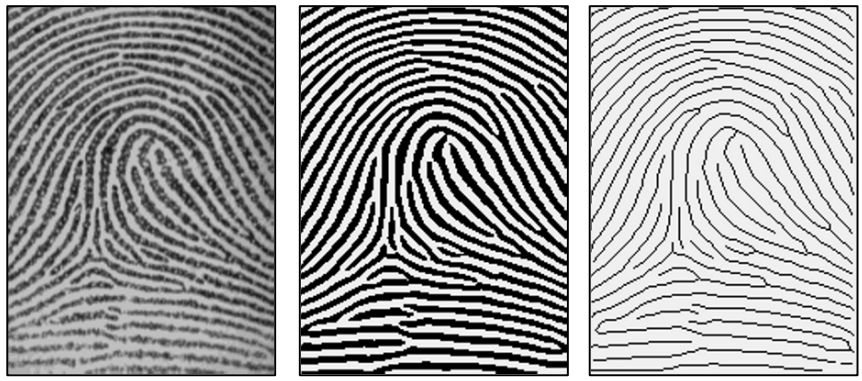


Figure 3.39. a) A fingerprint gray-scale image; b) the image obtained after binarization of the image in a); c) skeleton image obtained after a thinning of the image in b). Reprinted with permission from Maio and Maltoni (1997). © IEEE.

In general, different portions of an image may be characterized by different contrast and intensity and, consequently, a single threshold for the entire image is not sufficient for a correct binarization. For this reason, the *local threshold* technique changes *t* locally, by adapting its value to the average local intensity. In the specific case of fingerprint images, which are sometimes of very poor quality, a local threshold method cannot always guarantee acceptable results and more effective fingerprint-specific solutions are necessary. In the rest of this section, the commonly used binarization methods used for fingerprints are briefly summarized.

The FBI "minutiae reader" designed by Stock and Swonger (1969) (see also Stock, 1977) binarizes the image through a composite approach based on a local threshold and a "slit comparison" formula that compares pixel alignment along eight discrete directions. In fact, it is observed that for each pixel that belongs to a ridge line, there exists an orientation (the ridge line orientation) whose average local intensity is higher than those of the remaining orientations (which cross different ridges and valleys).

Moayer and Fu (1986) proposed a binarization technique based on an iterative application of a Laplacian operator and a pair of dynamic thresholds. At each iteration, the image is convolved through a Laplacian operator and the pixels whose intensity lies outside the range bounded by the two thresholds are set to 0 and 1, respectively. The thresholds are progressively moved towards a unique value so that a guaranteed convergence is obtained. A similar

method was proposed by Xiao and Raafat (1991a, b) who, after the convolution step, employed a local threshold to deal with regions with different contrast.

A fuzzy approach to image enhancement proposed by Verma, Majumdar, and Chatterjee (1987) that uses an adaptive threshold to preserve the same number of 1 and 0 pixels for each neighborhood forms the basis of their binarization technique. The image is initially partitioned in small regions and each region is processed separately. Each region is submitted to the following steps: smoothing, fuzzy coding of the pixel intensities, contrast enhancement, binarization, 1s and 0s counting, fuzzy decoding, and parameter adjusting. This sequence is repeated until the number of 1s approximately equals the number of 0s.

Coetzee and Botha (1993) presented a binarization technique based on the use of edges in conjunction with the gray-scale image. Edge extraction is performed by using the standard Marr and Hildreth (1980) algorithm. Then, the ridges are tracked by two local windows: one in the gray-scale image and the other in the edge image; in the gray-scale domain, the binarization is performed with a local threshold, whereas in the edge-image, a blob-coloring routine is used to fill the area delimited by the two ridge edges. The resulting binary image is the logical OR of the two individual binary images.

Ratha, Chen, and Jain (1995) introduced a binarization approach based on peak detection in the gray-level profiles along sections orthogonal to the ridge orientation (see Figure 3.40). A 16×16 oriented window is centered around each pixel $[x, y]$. The gray-level profile is obtained by projection of the pixel intensities onto the central section. The profile is smoothed through local averaging; the peaks and the two neighboring pixels on either side of each peak constitute the foreground of the resulting binary image (see Figure 3.41).

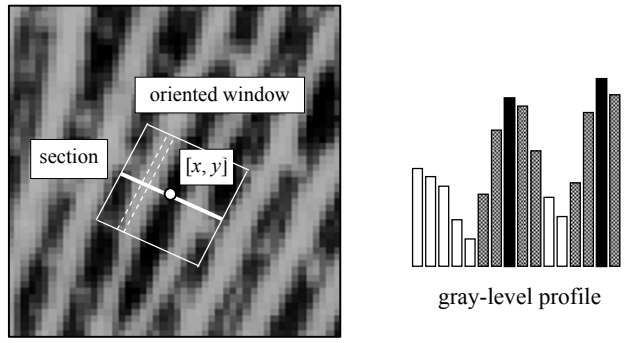


Figure 3.40. An example of a gray-level profile obtained through projection of pixel intensities on the segment centered at $[x,y]$ and normal to the local ridge orientation θ_{xy}.

Sherstinsky and Picard (1996) designed a method for fingerprint binarization that employs a dynamic non-linear system called "M–lattice." This method is based on the reaction-diffusion

model first proposed by Turing in 1952 to explain the formation of texture patterns such as zebra stripes.

Domeniconi, Tari, and Liang (1998) modeled fingerprint ridges and valleys as sequences of local maxima and saddle points. Maxima and saddle points are detected by evaluating the gradient ∇ and the Hessian matrix **H** at each point. The Hessian of a two-dimensional surface $S(x, y)$ is a 2×2 symmetric matrix whose elements are the second-order derivatives of S with respect to x^2, xy, and y^2. The eigenvectors of **H** are the directions along which the curvature of S is extremized. Let **p** be a stationary point (i.e., a point such that ∇ in **p** is **0**) and let λ_1 and λ_2 be the eigenvalues of **H** in **p**. Then **p** is a local maximum if $\lambda_1 \leq \lambda_2 < 0$ and is a saddle point if $(\lambda_1 \cdot \lambda_2) < 0$. Since ridges may gradually change their intensity values along the ridge direction, causing the surface to have either positive or negative slope, the algorithm also considers non-stationary points **p** such that the gradient directions φ_1 and φ_2 at two neighboring points $\mathbf{p}_1$ and $\mathbf{p}_2$ (located along the normal to the ridge) face each other: $angle(\varphi_1, \varphi_2) \cong 180°$. The authors also proposed an edge-linking algorithm to connect the ridge pixels detected in the first stage.

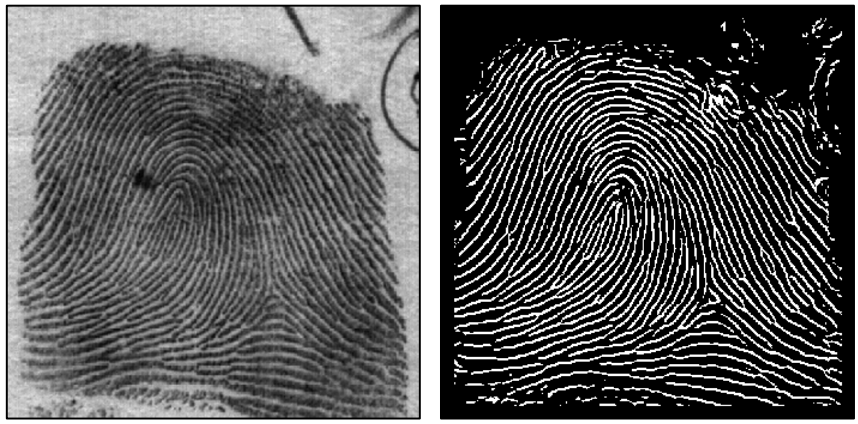


Figure 3.41. An example of fingerprint binarization using the Ratha, Chen, and Jain (1995) method. © Elsevier.

A slightly different topological approach was proposed by Tico and Kuosmanen (1999a) who treated a fingerprint image as a noisy sampling of the underlying continuous surface (Figure 3.8) and approximated it by orthogonal Chebyshev polynomials. Ridge and valley regions are discriminated by the sign of the maximal normal curvature of this continuous surface (Wang and Pavlidis, 1993). The maximal normal curvature along a given direction **d** may be computed as $\mathbf{d}^T \mathbf{H} \mathbf{d}$, where **H** is the Hessian matrix; this means that the second-order derivatives have to be estimated at each point.

Other binarization approaches can be found in Abutaleb and Kamel (1999), Govindaraju, Shi, and Schneider (2003), Mieloch, Mihailescu, and Munk (2005) and Zhang and Xiao (2006).

Most of the enhancement algorithms based on contextual filtering (discussed in Section 3.6.2) may produce a clear binary image for appropriately chosen parameters. In any case, even when the output of the contextual filtering is a gray-scale image, a simple local thresholding technique often results in satisfactory binarization. Examples of binary images obtained through the O'Gorman and Nickerson (1989) approach are shown in Figures 3.39b and 3.42a. Analogous results may be obtained through the contextual filtering techniques proposed by Donahue and Rokhlin (1993), Mehtre (1993), Sherlock, Monro, and Millard (1992, 1994), Watson, Candela, and Grother (1994) and Hong, Wan, and Jain (1998). Figure 3.42 shows the results obtained by binarizing a portion of a good quality fingerprint image through some of the methods described in this section; contextual filtering-based methods (a) and (d) produced the most regular binary ridge patterns.

Minutiae detection from binary images is usually performed after an intermediate thinning step that reduces the width of the ridges to one pixel. Unfortunately, thinning algorithms are rather critical and the aberrations and irregularity of the binary-ridge boundaries have an adverse effect on the *skeletons* (i.e., the one-pixel-width ridge structure), resulting in "hairy" growths (spikes) that lead to the detection of spurious minutiae. With the aim of improving the quality of the binary images before the thinning step, some researchers have introduced regularization techniques which usually work by filling holes (see Figure 3.43), removing small breaks, eliminating bridges between ridges, and other artifacts. For this purpose, Coetzee and Botha (1993) identify holes and gaps by tracking the ridge line edges through adaptive windows and removing them using a simple blob-coloring algorithm. Hung (1993) uses an adaptive filtering technique to equalize the width of the ridges; narrow ridges in under-saturated regions are expanded and thick ridges in over-saturated regions are shrunk. Wahab, Chin, and Tan (1998) correct the binary image at locations where orientation estimates deviate from their neighboring estimates. This correction is performed by substituting the noisy pixels according to some oriented templates. Luo and Tian (2000) implement a two-step method, where the skeleton extracted at the end of the first step is used to improve the quality of the binary image based on a set of structural rules; a new skeleton is then extracted from the improved binary image.

Mathematical morphology (Gonzales and Woods, 2007) is a powerful and elegant tool of digital topology that allows a regularization of the shape of binary objects. Some authors propose morphology-based techniques for regularizing binary fingerprint images:

- Fitz and Green's (1996) and Ikeda et al. (2002) remove small lines and dots both in the ridges and valleys of binary images through the application of morphological operators.
- To remove the spikes that often characterize the thinned binary images, Ratha, Chen, and Jain (1995) implement a morphological "open" operator whose structuring element is a small box oriented according to the local ridge orientation.

- Liang and Asano (2006a) recommend using Generalized Morphology Operators (GMO) that may increase the robustness of the algorithms to noise and small intrusions, especially when medium size structuring elements are used. An efficient implementation of GMO-based techniques for removing salt and pepper noise and small islands is proposed based on the distance transform (Gonzales and Woods, 2007) and the integral image (Viola and Jones, 2001).

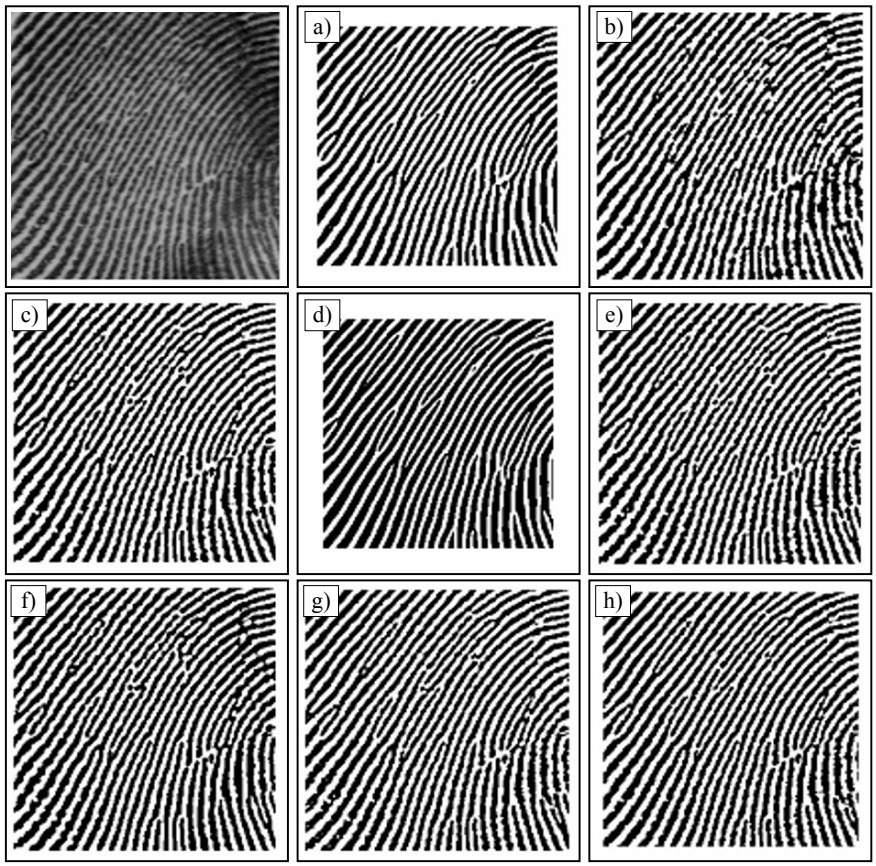


Figure 3.42. A portion of a good quality fingerprint image and its binarization through some of the methods discussed in this section: a) O'Gorman and Nickerson (1989); b) Verma, Majumdar, and Chatterjee (1987); c) local threshold approach; d) Sherlock, Monro, and Millard (1994); e) Xiao and Raafat (1991b); f) Moayer and Fu (1986); g) Stock and Swonger (1969); h) Watson, Candela, and Grother (1994).

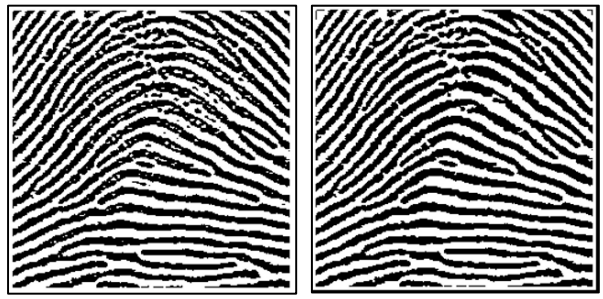


Figure 3.43. The result of eliminating small holes from both the ridges and valleys of the binary image; input image is shown on the left and the output is shown on the right. The filtering is performed by computing the connected components of the image and by removing the components whose area (number of pixels) is smaller than a given threshold.

As far as thinning techniques are concerned (Lam, Lee, and Suen, 1992), a large number of approaches are available in the literature due to the central role of this processing step in many pattern recognition applications: character recognition, document analysis, and map and drawing vectorization. Hung (1993) used the algorithm by Arcelli and Baja (1984); Ratha, Chen, and Jain (1995) adopted a technique included in the HIPS library (Landy, Cohen, and Sperling, 1984), Mehtre (1993) employed the parallel algorithm described in Tamura (1978) and Coetzee and Botha (1993) used the method by Baruch (1988). In Ji et al. (2007) the skeleton is computed through a constrained PCNN (Pulse Coupled Neural Network) where the orientation image is used to constrain the thinning direction of PCNN thus allowing to reduce bothersome artifacts such as the short spikes that conventional thinning algorithms often produce. Sudiro, Paindavoine, and Kusuma (2007) noted that in the binarized image, valleys are often thinner than ridges, and since the time taken by a thinning algorithm increases with the initial thickness of the objects, they propose extracting minutiae from valleys to reduce the computation time.

Once a binary skeleton has been obtained, a simple image scan allows the pixels corresponding to minutiae to be detected according to the ANSI/NIST-ITL 1 (2007) coordinate models shown in Figure 3.5: in fact the pixels corresponding to minutiae are characterized by a *crossing number* different from 2. The crossing number $cn(\mathbf{p})$ of a pixel $\mathbf{p}$ in a binary image is defined (Arcelli and Baja, 1984) as half the sum of the differences between pairs of adjacent pixels in the 8-neighborhood of $\mathbf{p}$:

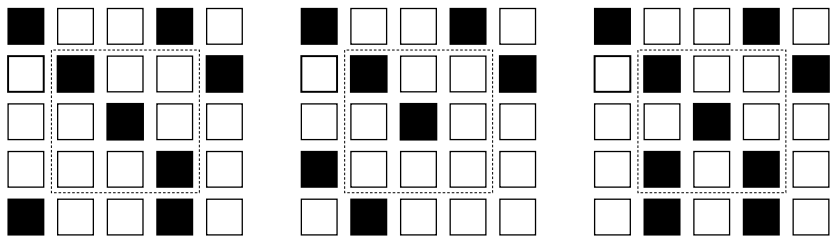
$$cn(\mathbf{p}) = \frac{1}{2} \sum_{i=1\ldots8} \left| val(\mathbf{p}_{i \bmod 8}) - val(\mathbf{p}_{i-1}) \right|,$$

where $\mathbf{p}_0$, $\mathbf{p}_1$, ...$\mathbf{p}_7$ are the pixels belonging to an ordered sequence of pixels defining the eight-neighborhood of $\mathbf{p}$ and $val(\mathbf{p}) \in \{0,1\}$ is the pixel value. It is simple to note (Figure 3.44) that a pixel $\mathbf{p}$ with $val(\mathbf{p}) = 1$:

- Is an intermediate ridge point if $cn(\mathbf{p}) = 2$.
- Corresponds to a ridge ending minutia if $cn(\mathbf{p}) = 1$.
- Corresponds to a bifurcation minutia if $cn(\mathbf{p}) = 3$.
- Defines a more complex minutia (e.g., crossover) if $cn(\mathbf{p}) > 3$.

Some techniques have been proposed in the literature to extract minutiae from binary images without using the crossing number to check the pixel connectivity on the skeleton resulting from a thinning step: Leung et al.'s (1991) method extracts the minutiae from thinned binary images using a three-layer perceptron neural network. The algorithm by Gamassi, Piuri, and Scotti (2005) is a variant of the crossing number method that can work with thick binary ridges: in fact, for each point the algorithm counts the black-white transitions along a square path centered at that point and large enough to touch two ridges. Approaches by Weber (1992), Govindaraju, Shi, and Schneider (2003) and Shi and Govindaraju (2006) work on thick binary ridges and exploit special ridge tracking algorithms. Shin, Hwang, and Chien (2006) encode thick binary ridges with Run Length Code (RLC), and extract minutiae by searching for the termination or bifurcation points of ridges in the RLC. Miao, Tang, and Fu (2007) encode the skeleton by means of principal curves extracted through the principal graph algorithms; principal curves are self-consistent smooth curves that are suitable to approximate noisy data. Finally, the foundation of the method developed by Székely and Székely (1993) lies in the use of a divergence operator capable of discerning fingerprint pattern discontinuities that correspond to minutiae.

Minutiae direction θ (see Figure 3.5), in addition to the minutiae coordinates, is used by most of the matching algorithms, to enforce minutiae pairing. A simple way to determine the minutiae direction is to start from the local ridge orientation at the minutia origin and to convert this orientation into a direction (i.e., deciding the quadrant) by looking at the departing ridge(s).



a) $cn(\mathbf{p}) = 2$ b) $cn(\mathbf{p}) = 1$ c) $cn(\mathbf{p}) = 3$

Figure 3.44. a) intra-ridge pixel; b) ridge ending minutia; c) bifurcation minutia.