Fingerprint Analysis and Representation

3

Abstract

This chapter introduces classical domain knowledge-based and emerging learning-based techniques for feature extraction in fingerprints. Specific sections are dedicated to explain the most effective approaches for segmentation, local orientation and frequency extraction, singularity detection and pose estimation, image enhancement, and minutiae and pore detection. The computation of global and local fingerprint image quality is also reviewed. Particular emphasis is given to robust feature extraction algorithms, trained on large corpuses of real or synthetic data, that can reliably operate on images of various quality encountered in applications.

Keywords

Feature extraction • Pose estimation • Local ridge orientations • Local ridge frequencies • Segmentation • Ridge flow singularities • Enhancement • Minutiae • Pores • Fingerprint quality

3.1 Introduction

A fingerprint is the impression of the exterior appearance of the fingertip epidermis. The most evident structural characteristic of a fingerprint is a pattern of interleaved *ridges* and *valleys* (Ashbaugh, 1999; Hicklin, 2009); in a fingerprint image, ridges (also called ridge lines) are dark whereas valleys are bright (see Fig. 3.1). Ridges vary in width from 100 μm, for very thin ridges, to 300 μm for thick ridges. Generally, the period of a ridge/valley cycle is about 500 μm. Most injuries to a finger such as superficial burns, abrasions, or cuts do not affect the underlying ridge structure, and the original pattern is duplicated in any new skin that grows.

D. Maltoni et al., *Handbook of Fingerprint Recognition*,
https://doi.org/10.1007/978-3-030-83624-5_3

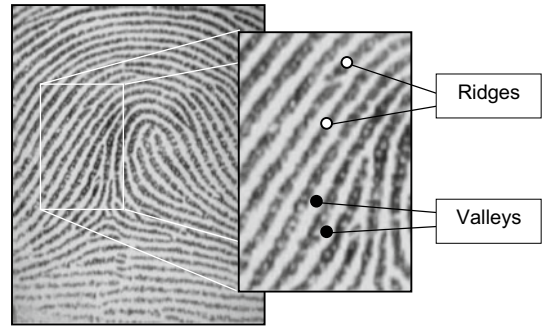

Fig. 3.1 Ridges and valleys in a fingerprint image

Ridge details are generally described in a hierarchical order at three different levels, namely Level 1 (the overall global ridge flow pattern), Level 2 (minutiae points), and Level 3 (pores, the local shape of ridge edges, etc.).

At the global level (Level 1), ridges often run smoothly in parallel but exhibit one or more regions where they assume distinctive shapes (characterized by high curvature, frequent ridge terminations, etc.). These regions, called *singularities* or *singular regions*, may be broadly classified into three typologies: *loop*, *delta*, and *whorl* (see Fig. 3.2). Singular regions belonging to loop, delta, and whorl types are typically characterized by $\cap$, Δ, and O shapes, respectively. Sometimes whorl singularities are not explicitly introduced because a whorl type can be described in terms of two loop singularities facing each other.

Fingerprint matching algorithms can pre-align fingerprint images according to a landmark or a center point, called the *core*. Henry (1900) defined the core point as "the northmost point of the innermost ridge line." In practice, the core point corresponds to the center of the northmost loop-type singularity. For fingerprints that do not contain loop or whorl singularities (e.g., those belonging to the Arch class in Fig. 3.3), it is difficult to define the core. In these cases, the core is usually associated with the point of maximum ridge-line curvature. Unfortunately, due to the high variability of fingerprint patterns, it is difficult to reliably locate a registration (core) point in all the fingerprint images. Singular regions are commonly used for fingerprint classification (see Fig. 3.3), that is assigning a

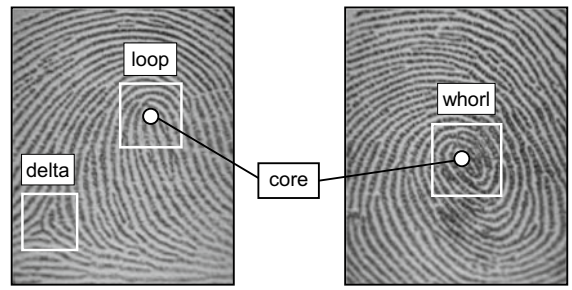

Fig. 3.2 Singular regions (white boxes) and core points (filled circles) in fingerprint images

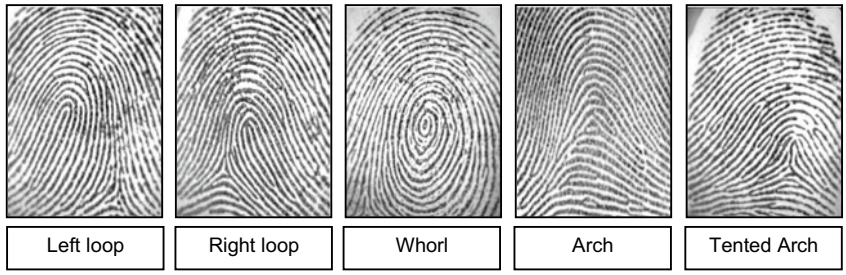

Fig. 3.3 One fingerprint from each of the five major classes defined by Henry (1900)

fingerprint to a class among a set of distinct classes, with the aim of simplifying search and retrieval (ref. Chap. 5).

At the local level (Level 2), other important features, called *minutiae*, can be found in the fingerprint patterns. Minutia means small detail; in the context of fingerprints, it refers to various ways that the ridges can be discontinuous (see Fig. 3.4). For example, a ridge can suddenly come to an end (ridge ending) or can divide into two ridges (bifurcation). Minutiae are the most commonly used features in both manual and automated fingerprint matching. Sir Francis Galton (1822–1911) was the first person to categorize minutiae and to observe that they remain unchanged over an individual's lifetime (Galton, 1892). Minutiae are sometimes called "Galton details" in his honor. In a full fingerprint (i.e., rolled impression), the number of minutiae can be more than 100; however, as discussed in Chap. 8, the spatial and angular coincidence or correspondence of a relatively small number of minutiae (e.g., as few as 12–15) may be sufficient to claim with confidence that two fingerprint impressions originate from the same finger. Some interesting statistical data on minutiae distribution can be found in Champod et al. (2016) and Stoney and Thornton (1987): in particular, average densities of 0.49 and 0.18 minutiae/mm^2 were estimated by Champod et al. (2016) inside the singular regions and outside the singular regions, respectively. Although several types of minutiae can be defined (the most common types

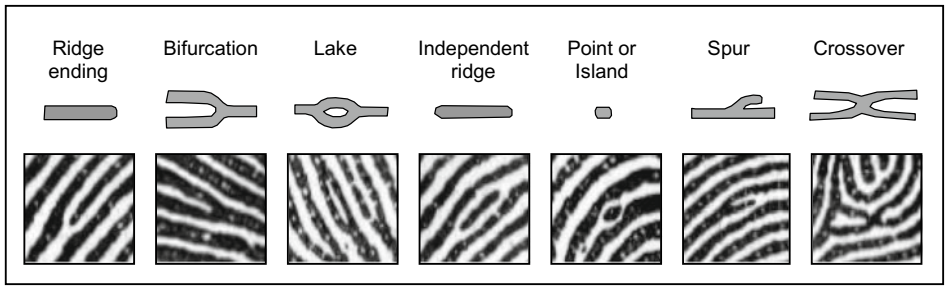

Fig. 3.4 Seven most common minutiae types

are shown in Fig. 3.4), only *ridge endings* and *bifurcations* are considered by the fingerprint encoding standards (ISO/IEC 19794-2, 2011 and ANSI/NIST-ITL 1-2011, 2015) to deal with the practical difficulty in automatically discerning the different types with high accuracy. ANSI/NIST-ITL 1-2011 (2015) recommends that all complex minutiae types such as crossovers/trifurcations should be marked as combinations of bifurcations and ridge endings.

Each minutia is denoted by its origin (i.e., x- and y-coordinates), angle, type, and quality. The angle is defined according to the ridge/valley orientation at the minutia position (Fig. 3.5). In practice, an ambiguity exists between ridge-ending and bifurcation minutiae types; depending on the finger pressure against the surface where the fingerprint impression is formed, ridge endings may appear as bifurcations and vice versa. However, given the convention used to define minutiae angle, there is no significant change in the angle if the minutia appears as a ridge ending in one impression and as a bifurcation in another impression of the same finger.

Figure 3.6a shows a portion of the fingerprint image where the ridge lines appear as dark traces on a light background; two ridge endings (1, 2) and one bifurcation (3) are shown. Note that on the negative image (Fig. 3.6b), the corresponding minutiae take the same positions, but their type is exchanged: ridge endings now appear as bifurcations

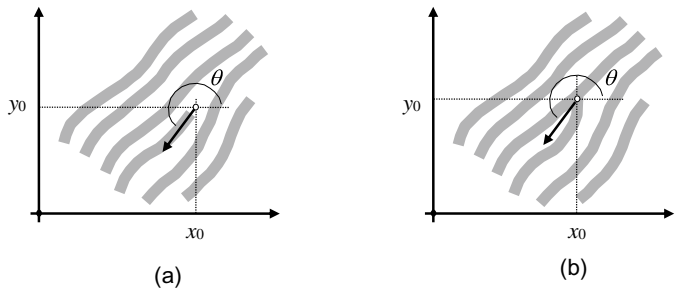

Fig. 3.5 Minutiae encoding in ISO/IEC 19794-2 (2011) and ANSI/NIST-ITL 1-2011 (2015). **a** For a ridge-ending minutia, the origin $[x_0,y_0]$ is placed at the forking point of the medial skeleton of the valley area immediately in front of the ridge ending, and θ is the angle (running up the ridge) that the ridge tangent forms with the horizontal axis; **b** for a bifurcation minutia, the origin is placed at the forking point of the medial skeleton of the ridge and θ is directed to the valley. Note that detecting a ridge ending as a bifurcation (or vice versa) does not cause an abrupt 180° change in the angle

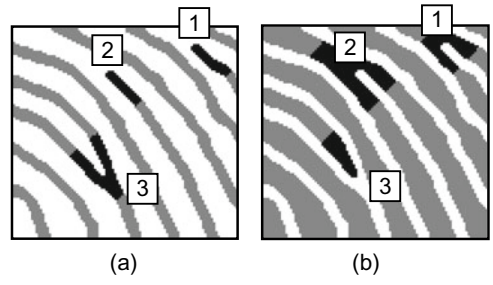

Fig. 3.6 The ridge-ending/bifurcation duality; in **a** a binary-ridge map image and **b** its negative image (i.e., dark and bright pixels are swapped)

and vice versa (this property is known as ridge-ending/bifurcation *duality*). Besides the two coordinates and the angle, other attributes can be associated with each minutia: these usually consist of features extracted from the minutia neighbors and, as discussed in Sect. 4.4, can be very useful to improve fingerprint matching accuracy.

At the very local level (Level 3), additional fine details can be extracted in the fingerprint pattern. They include all dimensional attributes of the ridges such as width, shape, edge contour (Fig. 3.7a), pores (Fig. 3.7a), incipient ridges (Fig. 3.7b), breaks, creases, and scars (Fig. 3.7c). Each ridge of the epidermis (outer skin) is dotted with *pores* (or *sweat pores*) along its entire length and anchored to the dermis (inner skin) by a double row of peglike protuberances or papillae (Roddy & Stosz, 1997). Pores may range in size from 60 to 250 μm. It was observed that the number of pores along a centimeter of ridge varies from 9 to 18. It has been claimed that 20–40 pores may be sufficient to determine the identity of a person (Ashbaugh, 1999). Although Level-3 features are highly distinctive and extremely important for latent fingerprint examiners, currently few automated matching techniques use them since their reliable detection even in high-resolution fingerprint scanners (e.g., 1,000 dpi) and good-quality fingerprint images is not guaranteed (Zhang et al., 2011a; Zhao & Jain, 2010). Thanks to the work of the Committee to Define an Extended Fingerprint Feature Set (CDEFFS, 2008) Level-3 features are now encoded in Type-9 records of ANSI/NIST-ITL 1-2011 (2015).

Although a few fingerprint matching techniques in the literature directly compare images through correlation-based methods (see Sect. 4.2), a representation based on the sensed gray-scale image intensities is usually considered not robust. Therefore, most of the fingerprint recognition algorithms employ a feature extraction stage for identifying salient features.

The features extracted from fingerprint images often have a direct physical counterpart (e.g., singularities or minutiae), but sometimes they are not directly related to any physical

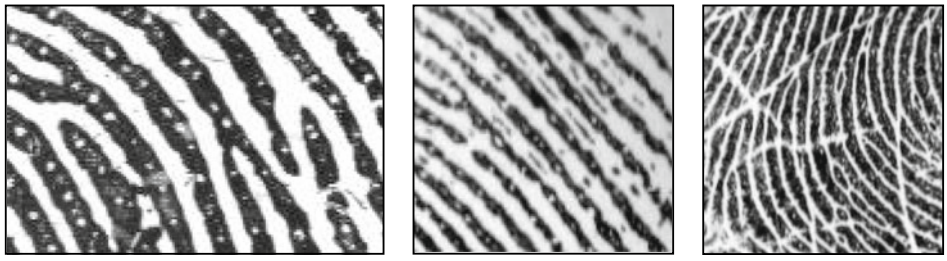

Fig. 3.7 Level-3 fingerprint features. **a** A fingerprint portion, acquired at 1000 dpi, where pores are well evident. The local variability of the ridge width and shape and the irregularity of the ridge contours is also visible; **b** incipient ridges are partially developed ridges that can occur in the valley between normal ridges: they are often fragmented and do not contain pores; and **c** creases present in a portion of a fingerprint

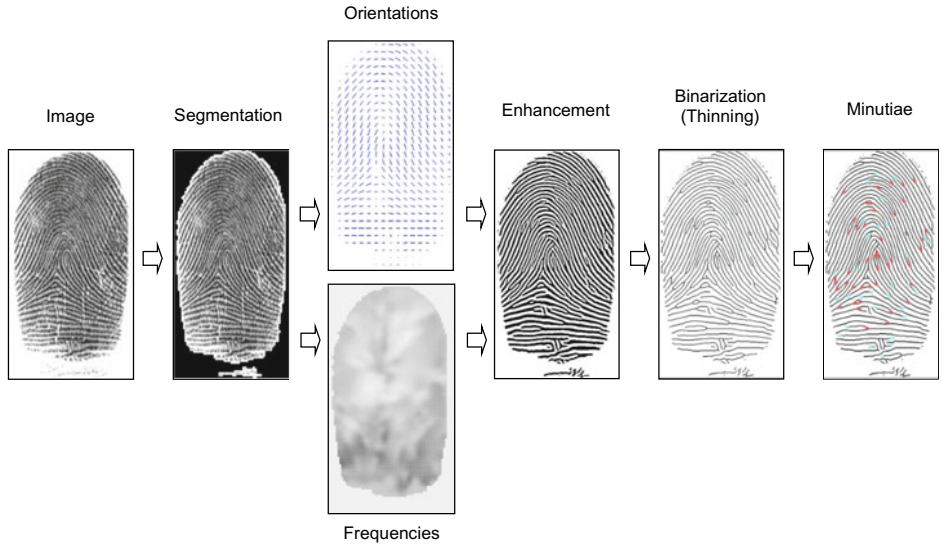

Fig. 3.8 Processing steps in a typical minutiae extraction pipeline. Details of individual steps are introduced in the following sections

attributes. This chapter mainly discusses the extraction of three-level features with clear physical meanings; other fingerprint representations used by automated matching techniques (e.g., minutiae descriptors, SIFT, and CNN-based representations) are introduced in Chap. 4. Features may be used either for matching or their computation may serve as an intermediate step for the derivation of other features. Figure 3.8 shows the typical processing steps performed to extract minutiae.

Throughout this book, a fingerprint image is represented as a two-dimensional surface. Let $\mathbf{I}$ be a gray-scale fingerprint image with g gray levels, and $\mathbf{I}[x, y]$ be the gray level of pixel $[x, y]$ in $\mathbf{I}$. Let $z = \mathrm{S}(x, y) = (g - 1 - \mathbf{I}[x, y])$ be the discrete surface corresponding to the image $\mathbf{I}$. By associating dark pixels with gray levels close to 0 and bright pixels with gray levels close to $g - 1$, the fingerprint ridge lines (appearing dark in $\mathbf{I}$) correspond to surface ridges, and the spaces between the ridge lines (appearing bright in $\mathbf{I}$) correspond to surface valleys (Fig. 3.9).

3.2 Segmentation

The term fingerprint *segmentation* is generally used to denote the separation of ridge–valley area (foreground) from the image background; an example of segmentation is shown in Fig. 3.10. Separating the background is useful to avoid the extraction of features in noisy areas that is often the background. Some authors use the term segmentation to indicate

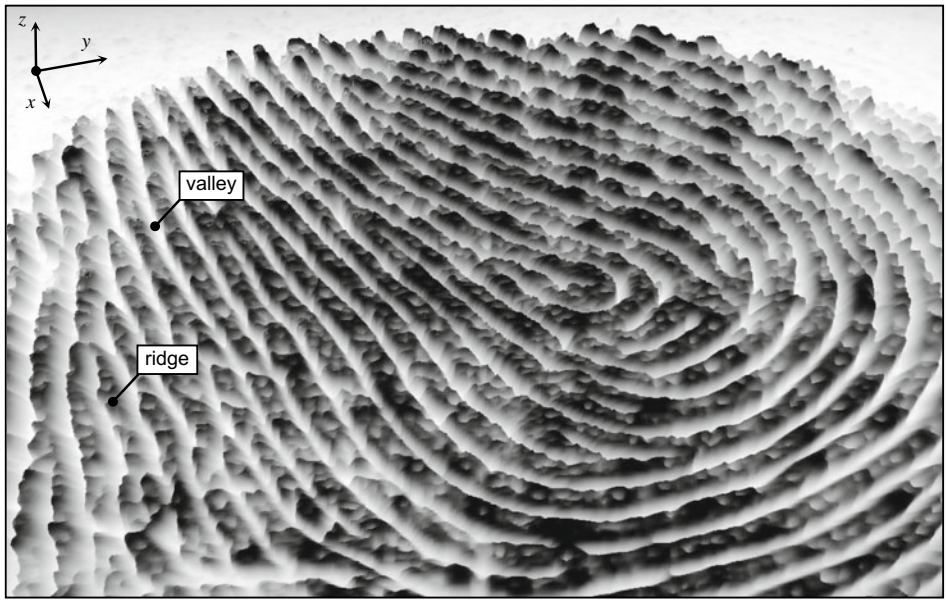

Fig. 3.9 A surface S representing a fingerprint portion

the transformation of the fingerprint image from gray-scale to binary image; throughout this book, the latter processing is referred to as *fingerprint binarization*.

Because fingerprint images are striated patterns (dark and light bands), using a global or local thresholding technique (Gonzales & Woods, 2007) does not allow the fingerprint area to be effectively isolated from the background. In fact, what really discriminates foreground and background is not the average image intensity but the presence of a striped and oriented pattern in the foreground and of an isotropic pattern (i.e., which does not have a dominant orientation) in the background. If the image background were always uniform and lighter than the fingerprint area, a simple approach based on local intensity thresholding could be effective for discriminating foreground and background; in practice, the presence of noise (such as that produced by dust and grease on the finger surface or surface of live-scan fingerprint scanners) requires more robust segmentation techniques.

Another problem related to fingerprint segmentation is *separating overlapped fingerprints*. This is of particular interest in forensic analysis to isolate two or more overlapped latent fingerprints unintentionally left on the same object. More information on this problem is provided in Chap. 6; the interested reader is also referred to Chen et al. (2011a), Feng et al. (2012), Zhao and Jain (2012), Zhang et al. (2014), Branka et al. (2019).

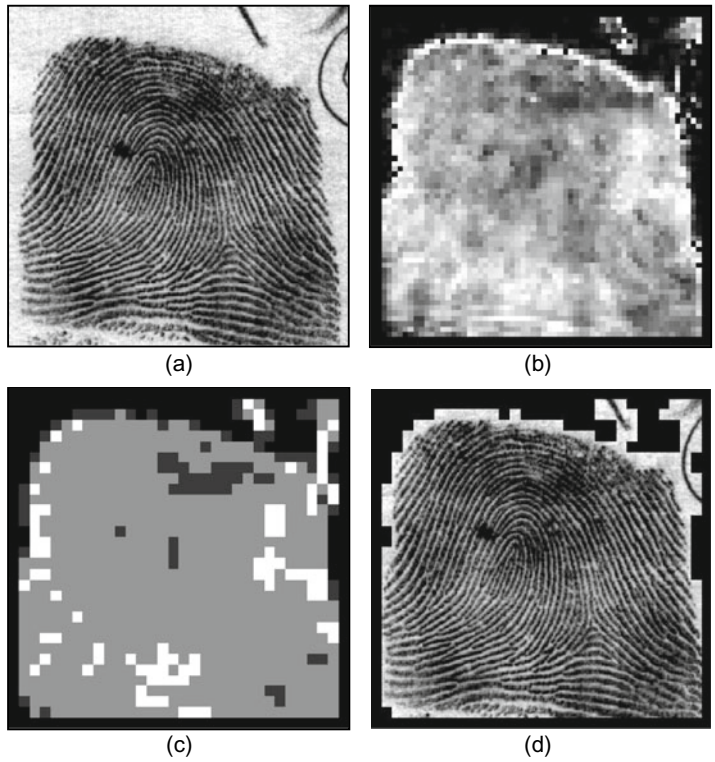

Fig. 3.10 Segmentation of a fingerprint image as proposed by Ratha et al. (1995): **a** original image; **b** variance field; **c** quality image derived from the variance field: a quality value "good," "medium," "poor", or "background" is assigned to each block according to its variance; and **d** segmented image. © Elsevier. Reprinted, with permission, from Ratha et al. (1995)

3.2.1 Segmentation Based on Handcrafted Features and Thresholding

To discriminate between striated patterns in the foreground and isotropic patterns in the background, a number of (block-wise) features have been proposed as follows:

- The presence of peaks in local histograms of ridge orientations (Mehtre et al., 1987).
- The variance of gray levels in the orthogonal direction to the ridge orientation (Ratha et al., 1995). See Fig. 3.10.
- The average magnitude of the gradient in each image block (Maio & Maltoni, 1997).
- The variance of the Gabor filter responses (Shen et al., 2001; Alonso-Fernandez et al., 2005; and Wang et al., 2005).
- The local energy in the Fourier spectrum (Pais Barreto Marques & Thome, 2005; Chikkerur et al., 2007).

Once the features have been extracted, a simple global thresholding is usually effective for the segmentation. Most of the above methods are very fast and perform reasonably well when the background is uniform and not too noisy. The methods introduced in the following subsections allow to achieve better results on difficult cases but can be practically adopted when the associated computational resource is feasible for the intended application.

3.2.2 Learning-Based Segmentation with Simple Classifiers

Learning-based techniques were introduced to obtain more accurate segmentations with respect to approaches based on feature value thresholding:

- Bazen and Gerez (2001) proposed a pixel-wise method, where three features (gradient coherence, intensity mean, and intensity variance) are computed in the neighborhood of each pixel, and a linear classifier associates the pixel with the background or the foreground. A supervised technique is used to learn the optimal parameters for the linear classifier for each specific fingerprint sensor. A final morphological post-processing step (Gonzales & Woods, 2007) is performed to eliminate holes in both the foreground and background and to regularize the external silhouette of the fingerprint area. The same feature vectors were used by Yin et al. (2005) for point-wise segmentation, but adopting a quadratic separation surface (non-linear classifier) instead of a hyperplane (linear classifier) causes a relevant reduction of the pixel classification rate.
- Chen et al. (2004a) trained a linear classifier to select foreground blocks based on (i) the block clusters degree, (ii) the difference of local block intensity mean and global image intensity mean, and (iii) the block variance. The block cluster degree is a measure of clustering of the ridge (dark) and valley (bright) gray levels. Morphology is then applied during post-processing to regularize the results. Some examples of segmentation are shown in Fig. 3.11.
- The basics of Zhu et al. (2006) technique are discussed in Sect. 3.3.4 in the context of local orientation correction.
- Paiva and Tasdizen (2012) argued that fingerprint patches seen in a high-dimensional space form a simple and highly regular circular manifold. Therefore, they introduced some features to characterize the manifold topology and trained a Fisher linear classifier (Duda et al., 2000) to discriminate between foreground and background blocks.

Fig. 3.11 Some examples of segmentation (marked by the solid boundary) with the method proposed by Chen et al. (2004a). © Springer Nature. Reprinted, with permission, from Chen et al. (2004a)

3.2.3 Total Variation Models

Total Variation (TV) models have been widely used in the context of image decomposition (Aujol et al., 2006; Buades et al., 2010). Some researchers successfully applied these techniques to latent fingerprints (Zhang et al., 2013; Cao et al., 2014), whose segmentation is particularly challenging and usually assisted by forensic experts through manual markup of the ROI (Region of Interest) as discussed in Chap. 6.

The TV models decompose the input image into two layers: cartoon and texture. In the context of fingerprint segmentation, the cartoon layer typically contains structured noise (drawings, characters, stain, speckle, etc., in the background region), while the texture layer contains the oscillatory or textured component characterizing the fingerprint pattern (foreground). The cartoon–texture decomposition facilitates the process of segmentation, as the region of interest can be easily detected from the texture layer using traditional segmentation methods.

The methods proposed by Zhang et al. (2012a, b, 2013) extend the classical TV methods by locally adjusting the cartoon/texture relative strength according to the background noise level and by imposing directional information. A three-layer decomposition (cartoon, texture, and noise) was proposed by Thai and Gottschlich (2016) whose method enforces sparsity and smoothness in the texture layer. As shown in Fig. 3.12, TV methods are quite effective to deal with difficult segmentation cases; on the other hand, the iterative approaches used to solve the underlying optimization problem are computationally demanding.

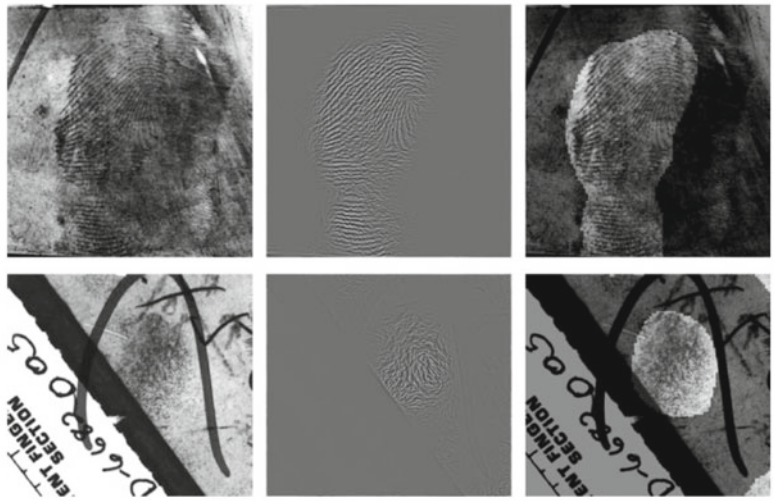

Fig. 3.12 Some examples of latent fingerprint image segmentation with the method proposed by Zhang et al. (2013). **a** Input image, **b** the texture layer after the decomposition, and **c** the segmentation mask imposed over the image. © IEEE. Reprinted, with permission, from Zhang et al. (2013)

3.2.4 Deep Learning Models

Deep learning approaches have been proposed for latent fingerprint segmentation by Zhu et al. (2017), Ezeobiejesi and Bhanu (2017), and Nguyen et al. (2018a):

- Zhu et al. (2017) trained four CNNs by using multi-scale patches and fused the corresponding output scores to improve segmentation accuracy.
- Ezeobiejesi and Bhanu (2017) used a stack of restricted Boltzmann machines (RBMs) to implement a generative feature learning model. For each fingerprint patch, the extracted features are then passed to a simple binary classifier.

Both these methods process patches separately (i.e., through a sliding window technique), and this makes them computationally inefficient. On the other hand, the Nguyen et al. (2018a) method combines fully convolutional neural network and detection-based approaches to process the entire input latent image in one shot. A visual attention mechanism was specifically designed to focus only on the latent fingerprint regions. More details on this approach are reported in Sect. 6.4.3.

3.3 Local Ridge Orientation Estimation

The estimation of local ridge orientations, also called ridge flow, is one of the most important steps of fingerprint image processing. In fact, the availability of reliable orientations can greatly improve subsequent tasks (in particular, fingerprint enhancement by contextual filtering).

The local ridge orientation at a pixel $[x, y]$ is the angle θ_{xy} that the fingerprint ridges, crossing through an arbitrary small neighborhood centered at $[x, y]$, form with the horizontal axis. Because fingerprint ridges are not directed, θ_{xy} is an unoriented direction lying in $[0\ldots180°[$. In the rest of the book, we use the term *orientation* to denote an unoriented direction in $[0\ldots180°[$ and the term *direction* to indicate an oriented direction in $[0\ldots360°[$.

Instead of computing local ridge orientation at each pixel, most of the fingerprint processing and feature extraction methods estimate the local ridge orientation at discrete positions (this reduces computational efforts and still allows estimates at other pixels through interpolation). The fingerprint *orientation field* (also called *orientation image* or *directional image*), first introduced by Grasselli (1969), is a matrix **D** whose elements encode the local orientation of the fingerprint ridges. Each element θ_{ij}, corresponding to the node $[i, j]$ of a grid located over the pixel $[x_i,y_j]$, denotes the average orientation of the fingerprint ridges in a neighborhood of $[x_i,y_i]$ (see Fig. 3.13). An additional value r_{ij} is often associated with each element θ_{ij} to denote the reliability (or consistency) of the orientation. The value r_{ij} is low for noisy and seriously corrupted regions and high for good-quality regions in the fingerprint image.

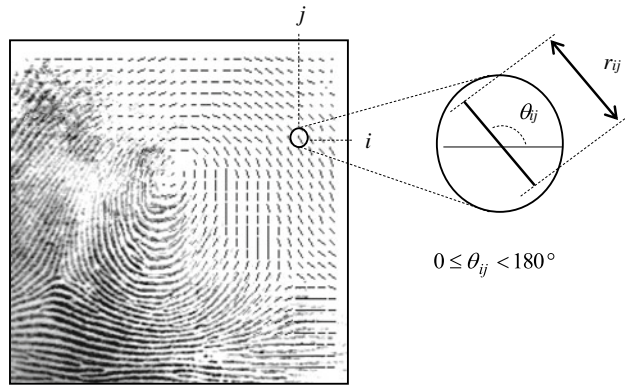

Fig. 3.13 A fingerprint image faded into the corresponding orientation image computed over a square-meshed grid of size 16×16. Each element denotes the local orientation of the fingerprint ridges; the element length is proportional to its reliability

3.3.1 Gradient-Based Approaches

The simplest and most natural approach for extracting local ridge orientation is based on the computation of gradients in the fingerprint image. The gradient $\nabla(x, y)$ at point $[x, y]$ of $\mathbf{I}$ is a two-dimensional vector $[\nabla_x(x, y),\nabla_y(x, y)]$, where ∇_x and ∇_y components are the derivatives of $\mathbf{I}$ at $[x, y]$ with respect to the x- and y-directions, respectively. It is well known that the gradient phase angle denotes the direction of the maximum intensity change. Therefore, the direction θ of a hypothetical edge that crosses the region centered at $[x, y]$ is orthogonal to the gradient phase angle at $[x, y]$. This method, although simple and efficient, has some drawbacks. First, using the classical Prewitt or Sobel convolution masks (Gonzales & Woods, 2007) to determine ∇_x and ∇_y components of the gradient and computing θ according to the arctangent of the ∇_y/∇_x ratio present problems due to the non-linearity and discontinuity around 90°. Second, a single orientation estimate reflects the ridge–valley orientation at too fine a scale and is generally very sensitive to the noise in the fingerprint image; on the other hand, simply averaging gradient estimates is not meaningful due to the circularity of angles: the average orientation between 5° and 175° is not 90° (as an arithmetic average would suggest) but 0°. Furthermore, the concept of average orientation is not always well defined; consider the two orthogonal orientations 0° and 90°; is the correct average orientation 45° or 135°?

Kass and Witkin (1987) proposed a simple but elegant solution to the above problem, which allows local gradient estimates to be averaged. Their basic idea is to double the angles, so that each single orientation estimate is encoded by the vector:

$$\mathbf{d} = [r \cdot \cos(2\theta), r \cdot \sin(2\theta)], \tag{3.1}$$

where 2θ is used in place of θ to discount the circularity of angles and the reliability r is proportional to the orientation estimate strength (e.g., the squared norm of the gradient: ${\nabla_x}^2 + {\nabla_y}^2$). Averaging the angles in a local $n \times n$ window W to obtain a more robust estimate $\bar{\mathbf{d}}$ can be performed by separately averaging the two (x and y) components:

$$\bar{\mathbf{d}} = \left[\frac{1}{n^2}\sum_W r \cdot \cos(2\theta), \frac{1}{n^2}\sum_W r \cdot \sin(2\theta)\right]. \tag{3.2}$$

Computing the average between two orthogonal orientations with Eq. (3.2) involves summing two vectors facing each other, and therefore, the length of the resulting vector is zero. This indicates that the vector is meaningless, independent of its orientation.

Based on the above idea, an effective method may be derived for computing the fingerprint orientation image (Bazen & Gerez, 2002; Rao, 1990; Ratha et al., 1995). For example, Ratha et al. (1995) computed the dominant ridge orientation θ_{ij} by combining multiple gradient estimates within a 17×17 window W centered at $[x_i,y_j]$:

$$\theta_{ij} = 90^\circ + \frac{1}{2}\text{atan2}\big(2G_{xy}, G_{xx} - G_{yy}\big), \tag{3.3}$$

$$G_{xy} = \sum_{h=-8}^{8}\sum_{k=-8}^{8} \nabla_x\big(x_i + h, y_j + k\big) \cdot \nabla_y\big(x_i + h, y_j + k\big),$$

$$G_{xx} = \sum_{h=-8}^{8}\sum_{k=-8}^{8} \nabla_x\big(x_i + h, y_j + k\big)^2,$$

$$G_{yy} = \sum_{h=-8}^{8}\sum_{k=-8}^{8} \nabla_y\big(x_i + h, y_j + k\big)^2,$$

where ∇_x and ∇_y are the x- and y-gradient components computed through 3×3 Sobel masks, and atan2(y,x) calculates the arctangent of the two variables y and x: it is similar to calculating the arctangent of y/x, except that the signs of both arguments are used to determine the quadrant of the result. An example of local orientation image computed with Eq. (3.3) is shown in Fig. 3.14b. Bazen and Gerez (2002) have shown that this method is mathematically equivalent to the principal component analysis of the autocorrelation matrix of the gradient vectors. Another gradient-based method, independently proposed by Donahue and Rokhlin (1993), relies on least-squares minimization to perform the averaging of orientation estimates and leads to equivalent expressions.

The reliability r of the estimate θ can be derived by the concordance (or coherence) of the orientation vectors **d** in the local window W (Bazen & Gerez, 2002; Kass & Witkin, 1987). In fact, due to the continuity and smoothness of fingerprint ridges, sharp orientation

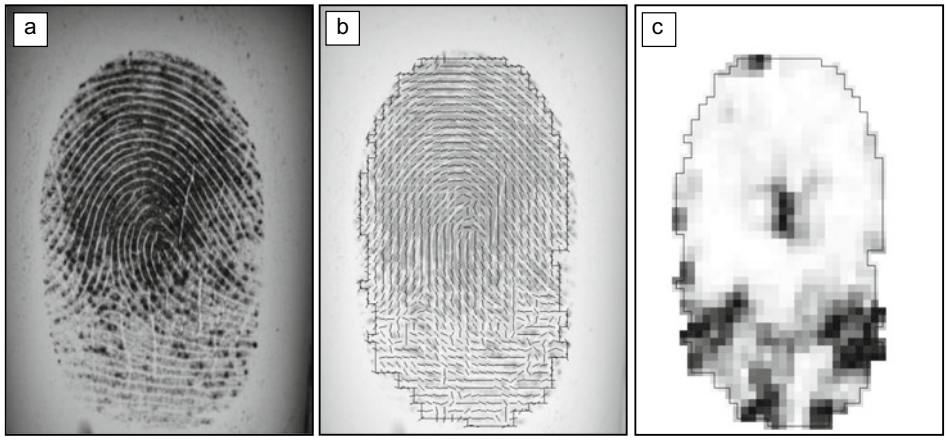

Fig. 3.14 A fingerprint image **a** local ridge orientation field, **b** computed with Eq. (3.3) and its local coherence map, and **c** computed with Eq. (3.4) over 3×3 blocks

changes often denote unreliable estimation. Kass and Witkin (1987) define the coherence as the norm of the sum of orientation vectors divided by the sum of their individual norms; this scalar always lies in [0, 1]: its value is 1 when all the orientations are parallel to each other (maximum coherence) and 0 if they point in opposite directions (minimum coherence):

$$r = \text{coherence}(\theta) = \frac{\left|\sum_W \mathbf{d}\right|}{\sum_W |\mathbf{d}|}. \tag{3.4}$$

An example of a local coherence map computed with Eq. (3.4) is shown in Fig. 3.14c. For the gradient-based approach corresponding to Eq. (3.3), it can be shown that Eq. (3.4) simplifies to

$$r_{ij} = \text{coherence}(\theta_{ij}) = \frac{\sqrt{(G_{xx} - G_{yy})^2 + 4G_{xy}^2}}{G_{xx} + G_{yy}}. \tag{3.5}$$

Jain et al. (1997) computed the concordance of the orientations according to their variance in 5×5 neighborhoods whereas Donahue and Rokhlin (1993) computed this according to the residual of the least-square minimization.

The major flaw of gradient-based orientation estimators is their failure in the near-zero gradient regions, namely ridge peaks and valley bottoms. In fact, in these regions, the small values of both the x- and y-components of the gradient imply high noise sensitivity. For this reason, some authors recommend to look beyond the first-order derivatives; see Larkin (2005) for a comprehensive review. Using second-order derivatives only partially solves the problem since the high noise sensitivity is moved to the zero crossing regions (i.e., inflexion points) where all the second-order derivatives and the Hessian are null. The method by Da Costa et al. (2001) is based on both first- and second-order derivatives: for each region, a binary decision on which operators to use is taken according to the local coherence of the two operators.

3.3.2 Slit- and Projection-Based Approaches

The first slit-based approach dates back to the 1960s, but some variants have been recently introduced. The basic idea is to define a fixed number (n_S) of reference orientations or slits S_k:

$$S_k = k\frac{\pi}{n_S}, \quad k = 0 \ldots n_S - 1$$

and to select the best slit $S_{k_{opt}}$ based on the pixel gray-values along the slits. The local orientation at $[x_i, y_j]$ is the orientation $\theta_{ij} = S_{k_{opt}}$ computed in a local window W centered at $[x_i, y_j]$.

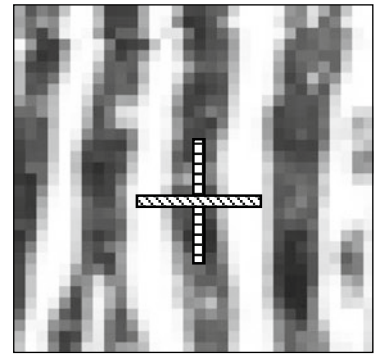

Fig. 3.15 A pair of orthogonal slits with high standard deviation contrast is shown for a local region of a fingerprint

Stock and Swonger (1969) sum the pixel gray values along eight slits and select the minimum-sum slit or maximum-sum slit for ridge or valley pixels, respectively: in fact, for pixels lying on ridges (dark), the sum of gray values along the ridge orientation is small, whereas for pixel lying on valleys (bright), the sum is high.

Based on the observation that the total fluctuation of the gray-scale is the smallest along the orientation of the ridges and largest in the orthogonal direction, similar methods have been proposed by Mehtre et al. (1987), He et al. (2003), and Oliveira and Leite (2008). In particular, Oliveira and Leite (2008) compute the standard deviation $stdev(S_k)$ of the gray-scale of the pixels corresponding to each slit S_k and select the optimal slit according to the maximum standard deviation contrast between a slit and its orthogonal slit (see Fig. 3.15).

Sherlock (2004) suggests projecting the ridge lines inside a local window along a number of discrete orientations (see Fig. 3.21 for an example of gray-scale projection): the projection that exhibits the smallest variation corresponds to the orientation of the ridges within the window. In the Ji and Yi (2008) approach, all the ridge lines except the central one are removed from the local window before computing the projections.

The computational complexity of slit- and projection-based approaches is usually higher than gradient-based techniques, and quantization might produce a coarser angular resolution. However, these methods allow to assign a probability value to each quantized orientation that can be useful to further process noisy regions. In other words, a gradient-based technique leads to a winner-take-all decision where just the optimal orientation is carried over, whereas in slit- and projection-based methods, one can also exploit the probability of the non-winning orientations for subsequent regularization or post-processing.

3.3.3 Orientation Estimation in the Frequency Domain

The Kamei and Mizoguchi (1995) method (well described in Kamei, 2004) is based on the application of 16 directional filters in the frequency domain. The optimal orientation at each pixel is then chosen not only according to the highest filter response but also taking local smoothing into consideration. Analogous results can be achieved in the spatial domain by using the Gabor-like filters, as proposed by Hong et al. (1996) and Nakamura et al. (2004).

The Chikkerur et al. (2007) approach is based on Short-Time Fourier Transform (STFT) analysis. The image is divided into partially overlapped blocks whose intensity values are cosine tapered moving from the center toward the borders. For each block, the Fourier Transform $F(u, v)$ is computed and its spectrum $|F(u, v)|$ is mapped to polar coordinates $|F(r, \theta)|$. The probability of a given θ value (within the block) is then computed as the marginal density function:

$$p(\theta) = \int_r p(r, \theta)\, dr, \quad \text{where } p(r, \theta) = \frac{|F(r, \theta)|}{\iint_{r,\theta} |F(r, \theta)|\, dr\, d\theta}.$$

The expected value of θ for the block is finally estimated, according to Eq. (3.2), as

$$E\{\theta\} = 90^\circ + \frac{1}{2}\text{atan2}\left(\int_\theta p(\theta)\sin(2\theta)d\theta,\ \int_\theta p(\theta)\cos(2\theta)d\theta\right). \tag{3.6}$$

Larkin (2005) proposed two energy-based operators that provide uniform and scale-invariant orientation estimation. The second operator, the most robust one, is based on spiral phase quadrature (or the Rietz transform). Although both the operators can be applied also in the spatial domain through convolution, the most natural and simpler implementation of these operators is in the frequency domain.

3.3.4 Orientation Image Regularization

The orientation image **D**, computed from poor-quality fingerprints, may contain several unreliable elements due to creases, local scratches, or cluttered noise. In this situation, a local smoothing can be very useful in enhancing **D**. This can be done by (re)converting the angles in orientation vectors **d** (Eq. (3.1)) and by averaging them through Eq. (3.2). Figure 3.16 shows an example of orientation image smoothing. However, such a simple averaging has some limitations (Fig. 3.16b):

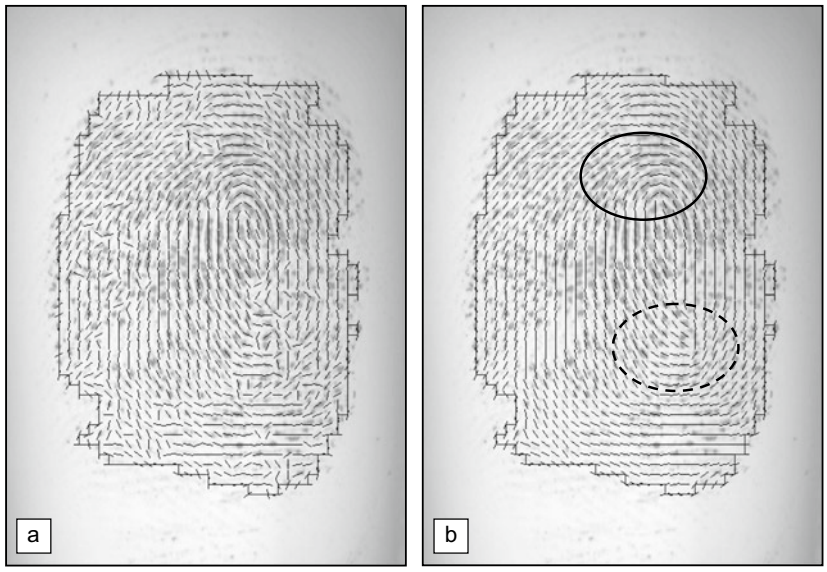

Fig. 3.16 **a** Estimation of local ridge orientation in a fingerprint through the gradient-based approach corresponding to Eq. (3.3): in the noisy regions, the estimate is unreliable; **b** two iterations of local (3×3) smoothing are applied, resulting in a more consistent representation; it is worth noting that while the smoothing recovered the correct orientation at several places (e.g., inside the solid circle), it altered the average orientation inside the region denoted by the dashed circle where incorrect orientations were dominating the correct one

1. It is ineffective when the incorrect orientations dominate the correct ones.
2. Smooths out high curvature values, especially in singular point regions.
3. Slightly shifts the loop singularities.

To overcome the undesired effects described above, more elaborate approaches than a simple average have been proposed:

- Jiang et al. (2004) noted that when a noisy region is smoothed, the local coherence tends to increase. On the other hand, if a high curvature region is smoothed, the local coherence remains low. To prevent smoothing out high curvature regions, the size of the smoothing window is then chosen according to a hierarchical coherence analysis. A similar approach was proposed by Can and Lin (2009).
- Liu et al. (2004) also argued that since the noise is often caused by scars and breaks in ridges, it can be modeled as an impulse function. To suppress such kind of noise, a simple averaging (i.e., a linear filtering) is not effective and a non-linear approach, similar to a median filtering, performs better.

- Zhu et al. (2006) trained a neural network to classify orientation image elements into two classes: correct and incorrect, based on an 11-dimensional feature vector. For each element, the 11 features, including gradient magnitude, gray-scale variance, gray-scale variance projected along the ridge orientation, inter-ridge distance, and variance of the peak/valley heights, are extracted from the corresponding image block. After classification, the incorrect ridge orientations are corrected using the orientation of the neighboring elements.
- Zhang and Yan (2007) define "invalid regions" in the foreground as the sets of connected elements with low coherence value and use the contours of these regions to build a constrained Delaunay triangulation that is used to correct the orientations through interpolation.
- In Oliveira and Leite (2008), correction is based on multi-scale analysis. In particular, they compute the orientation image at two different scales (fine scale and coarse scale) and correct only the elements whose value substantially differs between the two representations; in case of no substantial difference, the fine scale value is retained; otherwise, the coarse scale value is used to correct the fine scale orientation image. Another multi-resolution approach was introduced by Mei et al. (2009).
- Some authors proposed to regularize orientations through diffusions models (Perona, 1998): Hou and Yau (2010) variational approach tries to simultaneously smooth the orientations and preserve the singularity positions; Cao et al. (2012) model considers the distribution of divergence and curl of the orientation vector field. In general, these techniques require to solve non-linear differential equations through iterative numerical methods.

3.3.5 Global Models of Ridge Orientations

A global "mathematical" model for ridge orientation can be very useful for several purposes such as orientation image correction, fingerprint data compression, and synthetic fingerprint generation (see Chap. 7). Hereafter, we make a distinction among the models defined according to the singularities (type and position) from those independent of the singularities.

Models Based on Singularities

Sherlock and Monro (1993) proposed a mathematical model to synthesize a fingerprint orientation image from the position of loops and deltas alone. This approach is also known as the zero–pole model since it takes a loop as a zero and a delta as a pole in the complex plane. But, the model makes some simplifying assumptions and it does not cover all possible fingerprint patterns (fingerprints with different ridge patterns may present the same singularities at the same locations). Improvements to this method have

been proposed by Vizcaya and Gerhardt (1996) and Araque et al. (2002). In particular, Vizcaya and Gerhardt (1996) improved the zero–pole model by using a piece-wise linear approximation around singularities to adjust the zero and pole's behavior. These new models introduce additional degrees of freedom to better cope with fingerprint pattern variability. Vizcaya and Gerhardt (1996) also proposed an optimization technique, based on gradient descent, to determine the model parameters starting from an estimation of the orientation image. More details on the Sherlock and Monro (1993) and the Vizcaya and Gerhardt (1996) methods can be found in Sect. 7.2.2 where they are used for the synthetic generation of local orientation images.

In the above-mentioned methods, the influence of a singularity on the orientation of a given point does not depend on the distance of the point from the singularity; this can cause errors in regions far from singular points. Furthermore, these models cannot deal with fingerprints with no singularities such as Arch-type fingerprints. Zhou and Gu (2004b) developed a rational complex model, which generalizes the zero–pole model by adding some pseudozeros and pseudopoles as the control points: the pseudozeros are the roots of the additional polynomial in the numerator and the pseudopoles are the roots of the additional polynomial in the denominator. The model parameters can be derived by Weighted Least Square optimization (WLS) starting from the position of the singularities and estimation of the orientation image. Further variants were proposed by Zhou and Gu (2004a) and Gu et al. (2006).

Li et al. (2006) argued that a good orientation model should not only be accurate in approximating the underlying orientation image, but also should have prediction capability where the ridge information is not available (e.g., due to excessive noise). The method proposed by Li et al. (2006) uses a first-order phase portrait approach to compute the predicted orientation. This allows a reconstruction of the orientation using the data around the singular points. To increase the prediction accuracy, the initial estimation of the orientation fields (computed through a gradient-based approach) is refined by replacing the unreliable orientations with the predicted orientations. The refined orientation is then used to obtain the final orientation model using a constrained non-linear phase portrait approach.

Huckemann et al. (2008) argued that most of the global models proposed after Sherlock and Monro's model are controlled by too many parameters, making it critical to extract stable parameter values from a given orientation image. Their model, denoted as Quadratic Differential (QD), extends the basic Sherlock and Monro (1993) zero–pole model, by using as control parameters (besides the singularity positions) five values with clear geometric meaning: three parameters control the finger placement with respect to the image (i.e., origin and angle of the reference axes) and the remaining two parameters define the horizontal and vertical size of the finger. An extended version of the above method, named XQD, was proposed by Gottschlich et al. (2017).

All the above-mentioned modeling methods rely on the position of singularities and the underlying optimization techniques, but provide no guidance on whether to first compute

the singularity positions based on a noisy orientation map or improve the orientations based on inaccurate singularities. An elegant approach to overcome the above problem was proposed by Dass (2004) whose Bayesian formulation allows the simultaneous extraction of orientation image and singularities. The orientation image is iteratively updated by taking into account the spatial smoothness in a local neighborhood, the gradient values, and contributions to the local orientation given by the singularities. Obtaining orientation image and singularity information simultaneously offers the additional advantage of interleaved updating: the orientation image can be dynamically improved based on current singularity information and the singularity extraction can be performed with higher accuracy thanks to the improved orientation image. Unfortunately, the numerical optimization underlying the joint estimation can be very time-consuming.

Models Independent of the Singularities
Ridge orientations are modeled based on orthogonal polynomials which are defined independently of the singularities. Here, the model coefficients are determined as a data fitting problem starting from an initial coarse estimation:

- Wang et al. (2007a) proposed a fingerprint orientation model based on 2D Fourier series expansion (FOMFE) in the phase plane. The model can seamlessly summarize global features, including high curvature regions around singularities. The extended version by Wang and Hu (2011) can reconstruct the global orientation map starting from a partial fingerprint. Further variants (Tashk et al., 2009; Tao et al., 2010) were introduced to improve the quality of reconstruction in areas of high curvature and poor quality.
- The Ram et al. (2010) model uses the Legendre polynomials to independently model sine and cosine components of the orientation map. In the variant proposed by Jirachaweng et al. (2011), the regions where the basic model does not fit well the data (e.g., around singularities) are refined by using the higher order Legendre polynomials.
- Discrete Cosine Transform (DCT) was used as a basis function by Liu and Liu (2012) and Liu et al. (2014). In the former approach, the orientation field is reconstructed by a linear combination of DCT atoms, and sparse coding is used to improve robustness against noise.
- Bian et al. (2014) proposed a method for reconstructing the orientation field by using the best quadratic approximation by orthogonal polynomials in two discrete variables in the sine domain. The proposed basis functions are true orthogonal in the case of the finite discrete data (unlike for other polynomials such as the Legendre) and the underling optimization is claimed to be numerically more stable. An extension of this method combined with a final orientation diffusion is the basis of the approach described in Bian et al. (2017a).

Some researchers argued that modeling orientations based on orthogonal polynomials are closer to an approximation method rather than statistical modeling, since the natural variability of fingerprint patterns is not encoded in the model, and, when a large region is dominated by the noise, the recovery ability is limited. Learning-based methods discussed in the following subsection were conceived to overcome this limit.

3.3.6 Learning-Based Methods

One of the first learning-based models was introduced by Lee and Prabhakar (2008) whose approach computes the orientation image based on an MRF (Markov Random Field) made up of two components; one incorporates a global mixture model of orientation fields learned from training fingerprint examples and the other enforces a smoothness constraint over the orientation image in the neighboring regions. Although Lee and Prabhakar (2008) implementation is computationally intensive, it demonstrates the effectiveness of model-based estimation techniques.

Inspired by spelling correction techniques used in natural language processing, Feng et al. (2013) proposed an effective method to exploit the prior knowledge of fingerprint orientations (see Fig. 3.17). A dictionary of reference orientation patches is built off-line using a set of true orientation fields. During the online phase, a first estimation is achieved through one of the known techniques (e.g., STFT), then the orientation map is divided into overlapping patches, and for each patch, a dictionary lookup retrieved the six nearest neighbors dictionary "words". Finally, an energy minimization process, solved by loopy

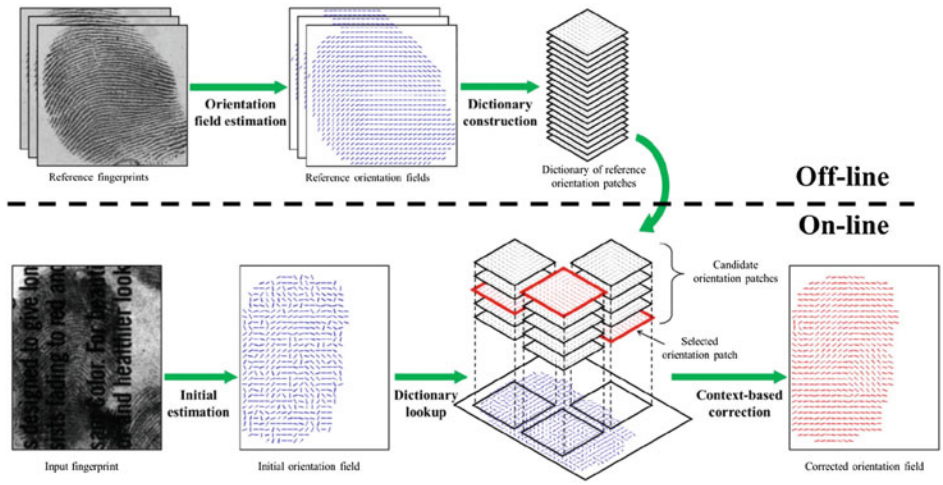

Fig. 3.17 The global-dictionary approach proposed by Feng et al. (2013). © IEEE. Reprinted, with permission, from Feng et al. (2013)

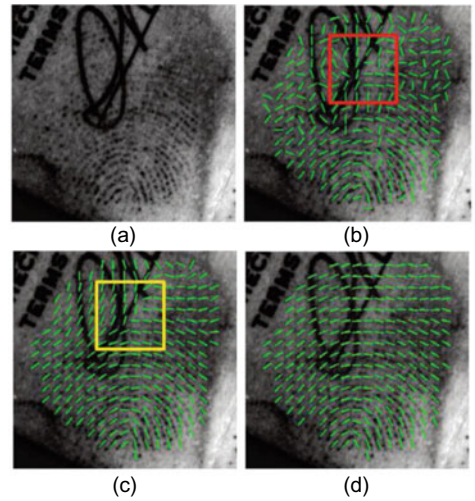

Fig. 3.18 **a** A latent fingerprint; **b** initial orientation extraction by STFT method; **c** global-dictionary approach correction is made through a dictionary patch which is feasible in general but unrealistic at that specific position; and **d** the local-dictionary approach properly handles this case. © IEEE. Reprinted, with permission, from Yang et al. (2014)

belief propagation, is applied to smooth out inconsistencies across neighboring patches when selecting the optimal assignments. Variants of this approach have been introduced by Chen et al. (2016), and Liu and Yang (2017), based on multi-scale analysis and sparse coding, respectively.

Yang et al. (2014) observed that in noisy images, some dictionary patches could be selected at places which are unrealistic given the overall structure of fingerprints (see Fig. 3.18) and proposed to use localized dictionaries for different regions. However, this approach requires fingerprints to be coarsely pre-aligned, which is a critical step for poor-quality and partial images. Therefore, a robust alignment was introduced based on probabilistic voting (i.e., the Hough transform) of all local orientation patches. The local-dictionary approach outperformed the global one in a number of experiments carried out by the authors.

In the method introduced by Cao et al. (2014), the dictionary words are not orientation patches but ridge patches (see Fig. 3.19a). This allows to simultaneously estimate local orientations and frequencies in order to properly apply contextual enhancements (Sect. 3.6.2).

Deep learning techniques based on CNN were proposed by Cao and Jain (2015), and Schuch et al. (2017a); training and inference are quite different in the two approaches:

- In the method by Cao and Jain (2015), (i) a dictionary of 128 orientation patches (of size 10×10) is created by unsupervised clustering good-quality orientation patches extracted from NIST SD4 (Watson & Wilson, 1992); (ii) a training set of 1.28 million good-quality image patches (of size 160×160) is then built from NIST SD14 (Watson,

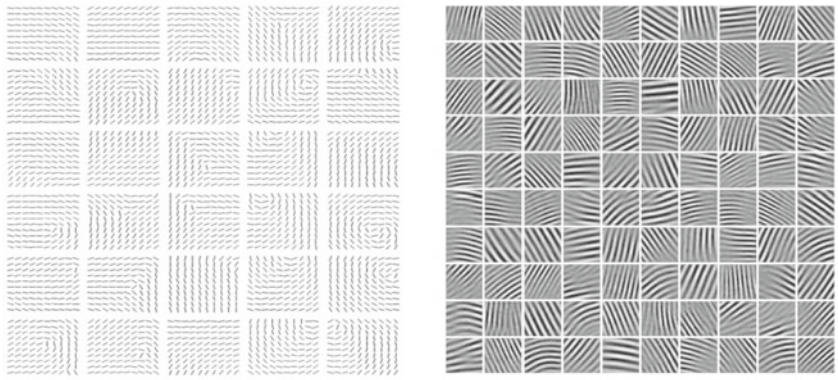

Fig. 3.19 **a** Orientation patches versus **b** ridge patches. © IEEE. Reprinted, with permission, from Cao et al. (2014)

1993) by labeling each image patch with the index of the closest dictionary word[1]; (iii) the training set patches are corrupted by line-like texture noise and block cancelation and used to train a CNN to classify an input patch into one of the 128 classes; (iv) during inference, the input image is divided into overlapping patches and each of them is classified by the CNN; and (v) the final orientation of each element in the input is determined by fusing the orientations of the covering patches.

- In the CNN approach by Schuch et al. (2017a), training is supervised and based on the manually marked orientations made available in the FVC-onGoing FOE training set (see Sect. 3.3.7). This approach is closer to a typical CNN-based semantic segmentation (Minaee et al., 2021) where a fully convolutional architecture is exploited to simultaneously process the whole image (with no need of sliding a window over local patches). The hierarchical arrangement of convolutional layers increases the receptive field size as one moves from the input to the output level, so that the prediction of each output element depends on a sufficiently large image portion. The authors empirically show that a classification network with a soft fusion of output labels (i.e., deep expectation) outperforms a similar model trained for orientation regression.

3.3.7 Benchmarking Fingerprint Orientation Extraction

While many researchers evaluated the efficacy of their orientation extraction approach on indirect tasks (e.g., the improvement of singularity detection or fingerprint recognition),

[1] The similarity between an orientation patch and an image patch is computed by extracting the orientations from the image patch (with a local method) and using a squared cosine distance.

Table. 3.1 A comparison of known approaches on FVC-onGoing FOE. The average error on the bad-quality partition (AvgError Bad) is the main indicator

Method	AvgError bad (°)	AvgError good (°)	Computation time on Intel Xeon E5410 (s)
CNN DEX-OF (Schuch et al., 2017a)	6.73	4.84	1.776
Local dictionary (Yang et al., 2014)	9.66	6.08	5.987
ROF (Div-Curl model) (Cao et al., 2012)	11.20	5.24	0.762
Adaptive polynomials (Turroni et al., 2011)	13.27	5.93	4.772
FOMFE (Wang et al., 2007a)	21.44	6.70	1.996
Gradient (baseline) (Turroni et al., 2011)	21.83	5.86	0.074

we believe that assessing this important module in isolation is very valuable because it can provide useful insights on the residual errors and guide future improvements.

FVC-onGoing FOE (Fingerprint Orientation Extraction) is a benchmark[2] specifically designed to evaluate the accuracy of orientation extraction algorithms. Even if the fingerprints used in FVC-onGoing FOE are plain impressions, methods conceived to work on latent prints can be applied to this dataset as well since recovering orientations on noisy latent fingerprints is not much different than processing poor-quality plain impressions. Manual labeling of orientations at selected places and subsequent interpolation is used to create a ground truth for FVC-onGoing FOE: even if this procedure is prone to inaccuracies (as pointed out by Schuch et al., 2017b), looking at the orientation error on the poor-quality partition of this benchmark gives a better overview of the state of the art (see Table 3.1). From Table 3.1, the advantages of learning-based techniques (Local Dictionary and CNN) on poor-quality images are evident and their robustness is extremely useful for latent fingerprint processing; however, their computational complexity is typically much higher than local techniques such as the gradient baseline, whose accuracy on reasonable-quality samples is comparable and more than satisfactory for several applications.

Further comparisons of fingerprint orientation techniques can be found in Turroni et al. (2011) and in the recent survey by Bian et al. (2019).

[2] https://www.biolab.csr.unibo.it/fvcongoing/UI/Form/BenchmarkAreas/BenchmarkAreaFOE.aspx.

3.4 Local Ridge Frequency Estimation

The local ridge frequency (or density) f_{xy} at point $[x, y]$ is the number of ridges per unit length along a hypothetical segment centered at $[x, y]$ and orthogonal to the local ridge orientation θ_{xy}. A frequency image **F**, analogous to the orientation image **D**, can be defined if the frequency is estimated at discrete positions and arranged into a matrix.

The local ridge frequency varies across different fingers and may also noticeably vary across different regions of the same fingerprint (see Fig. 3.20). Orczyk and Wieclaw (2011), based on manual marking by a forensic expert, reported that for 500 dpi fingerprints, most local ridge distances lie in the range 5–10 pixels with an average of 7.71.

Hong et al. (1998) estimate local ridge frequency by counting the average number of pixels between two consecutive peaks of gray levels along the direction normal to the local ridge orientation (see Fig. 3.21). For this purpose, the surface S corresponding to the fingerprint is sectioned with a plane parallel to the z-axis (see Fig. 3.9) and orthogonal to local ridge orientation. The frequency f_{ij} at $[x_i, y_j]$ is computed as follows.

1. A 32×16 *oriented window* centered at $[x_i, y_j]$ is defined in the ridge coordinate system (i.e., rotated to align the y-axis with the local ridge orientation).
2. The *x-signature* of the gray levels is obtained by accumulating, for each column x, the gray levels of the corresponding pixels in the oriented window. This is a sort of

Fig. 3.20 Two fingerprint images and the corresponding frequency image computed with the method proposed by Maio and Maltoni (1998a). A local 3×3 averaging is performed after frequency estimation to reduce noise. Light blocks denote higher frequencies. It is quite evident that significant changes may characterize different fingerprint regions and different average frequencies may result from different fingers

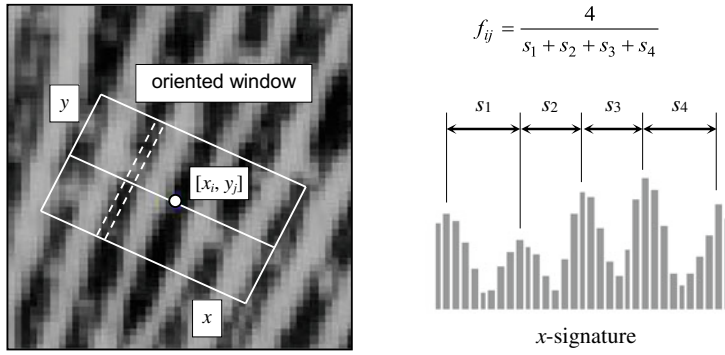

Fig. 3.21 An oriented window centered at $[x_i, y_j]$; the dashed lines show the pixels whose gray levels are accumulated for a given column of the *x*-signature (Hong et al., 1998). The *x*-signature on the right clearly exhibits five peaks; the four distances between consecutive peaks are averaged to determine the local ridge frequency

averaging that makes the gray-level profile smoother and prevents ridge peaks from being obscured due to small ridge breaks or pores.

3. f_{ij} is determined as the inverse of the average distance between two consecutive peaks of the *x*-signature.

The method is simple and fast. However, it is difficult to reliably detect consecutive peaks of gray levels in the spatial domain in noisy fingerprint images. In this case, the authors suggest using interpolation and low-pass filtering. An alternative way to extract ridge distances from the *x*-signature makes use of a fitting method based on the first- and second-order derivatives (Yang et al., 2003).

Jiang (2000) and Orczyk and Wieclaw (2011) also compute the local ridge frequency starting from the *x*-signatures. Jiang (2000), instead of measuring the distances in the spatial domain, makes use of a high-order spectrum technique called *mix-spectrum*. The ridge patterns in a fingerprint image are noisy periodic signals; when they deviate from a pure sinusoid shape, their energy is distributed to their fundamental frequency and harmonics. The mix-spectrum technique enhances the fundamental frequency of the signal by exploiting the information contained in the second and third harmonic.

Gottschlich (2012) argued that most of the frequency estimation errors of *x-signature*-based methods occur in curved regions (e.g., close to singularities) where a single dominant orientation cannot be defined. Therefore, he proposed to locally rectify ridges based on ridge flow curvature before computing the *x-signature* (Fig. 3.22). The rectification also allows to increase the window size.

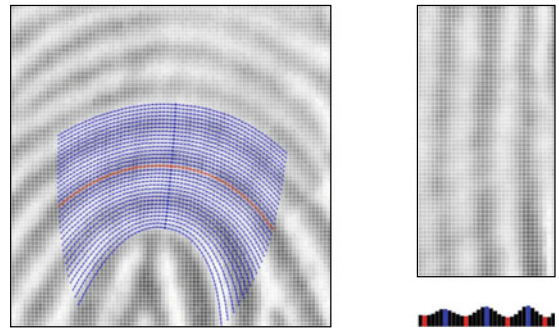

Fig. 3.22 The curvature-based rectification of ridges proposed by Gottschlich (2012) to improve *x-signature*-based local frequency estimation. © IEEE. Reprinted, with permission, from Gottschlich (2012)

In the method proposed by Maio and Maltoni (1998a), the ridge pattern is locally modeled as a sinusoidal-shaped surface, and the variation theorem is exploited to estimate the unknown frequency. The variation V of a function h in the interval $[x_1, x_2]$ is the amount of "vertical" change in h:

$$V(h) = \int_{x_1}^{x_2} \left| \frac{dh(x)}{dx} \right| \cdot dx.$$

If the function h is periodic at $[x_1, x_2]$ or the amplitude changes within the interval $[x_1, x_2]$ are small, the variation may be expressed as a function of the average amplitude α_m and the average frequency f (see Fig. 3.23).

$$V(h) = (x_2 - x_1) \cdot 2\alpha_m \cdot f.$$

Therefore, the unknown frequency can be estimated as

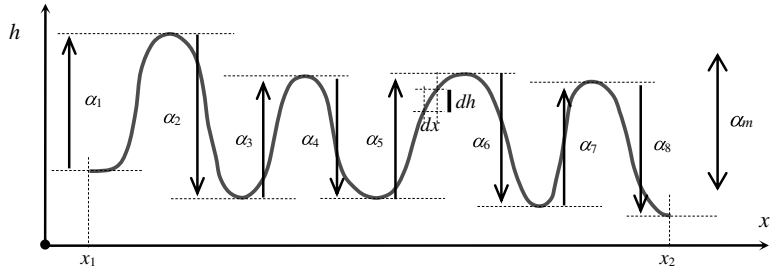

Fig. 3.23 The variation of the function h in the interval $[x_1, x_2]$ is the sum of amplitudes $\alpha_1, \alpha_2, \ldots, \alpha_8$ (Maio & Maltoni, 1998a). If the function is periodic or the function amplitude does not change significantly within the interval of interest, the average amplitude α_m can be used to approximate the individual α values. Then the variation may be expressed as $2\alpha_m$ multiplied by the number of periods of the function over the interval

$$f = \frac{V(h)}{2 \cdot (x_2 - x_1) \cdot \alpha_m}. \tag{3.7}$$

Maio and Maltoni (1998a) proposed a practical method based on the above analysis. The variation and the average amplitude of a two-dimensional ridge pattern are estimated from the first- and second-order partial derivatives and the local ridge frequency is computed from Eq. (3.7). Two examples of frequency images computed using this method are shown in Fig. 3.20.

Kovacs-Vajna et al. (2000) proposed a two-step procedure: first, the average ridge distance is estimated for each 64×64 sub-block of the image that is of sufficient quality and then this information is propagated, according to a diffusion equation, to the remaining regions. Two methods are considered in the first step: geometric and spectral. In the geometric approach, the central points of the ridges are computed on a regular grid and the ridge distances are measured on straight lines passing through these points. Unlike the *x*-signature approach, distances are directly measured in the two-dimensional image; several estimates on the same image block are performed to compensate for the noise. The second method is based on a search of the maxima in the Fourier power spectrum of each sub-block. Here too, the method works on two-dimensional signals. The invariance with respect to the local ridge orientations is obtained by performing the maxima search in radial directions: in fact, all the components (harmonics) having the same distance from the origin denote the same frequency.

Another geometrical method was introduced by Lehtihet et al. (2014); gray-level minima (lying on ridges) are used as the vertex for a Delaunay triangulation and the resulting graph is locally parsed to estimate distances of adjacent ridges.

Almansa and Lindeberg (1997, 2000) use scale space theory to locally estimate ridge width; their method relies upon combinations of normalized derivatives computed pointwise.

The approach by Chikkerur et al. (2007), already discussed in Sect. 3.3.3 for the estimation of the local ridge orientations, is based on Short-Time Fourier Transform (STFT) analysis. The probability of a given r value (corresponding to the average frequency f within the block) is computed as the marginal density function:

$$p(r) = \int_\theta p(r, \theta)\, d\theta, \quad \text{where } p(r, \theta) = \frac{|F(r, \theta)|}{\iint\limits_{r,\theta} |F(r, \theta)|\, dr\, d\theta}$$

and the expected value of r for the block is estimated as

$$E\{r\} = \int_r p(r)\, r\, dr. \tag{3.8}$$

Zhan et al. (2006) compared frequency estimation approaches operating in the spatial domain versus Fourier domain and concluded that the former can be implemented more efficiently but the latter seems to be more robust to noise.

3.5 Singularity Detection and Pose Estimation

Singularities are the most evident Level-1 features in the fingerprint pattern. They can be useful for coarse fingerprint classification (Chap. 5) and to define stable points for pose estimation and absolute registration. Most of the approaches proposed in the literature for singularity detection operate on the fingerprint orientation image. In the rest of this section, the main approaches are presented. These include algorithms based on the Poincaré index, local characteristics of the orientation image, partition of the orientation image, and global model of the orientation image. Each of these four families of algorithms is described in Sects. 3.5.1–3.5.4.

3.5.1 Poincaré

An elegant and practical method based on the Poincaré index was proposed by Kawagoe and Tojo (1984). Let **G** be a vector field and C be a curve immersed in **G**; then the Poincaré index $P_{\mathbf{G},C}$ is defined as the total rotation of the vectors of **G** along C (see Fig. 3.24).

If **G** is the discrete vector field associated with a fingerprint orientation image (Note that a fingerprint orientation image is not a true vector field in as much as its elements are unoriented directions (i.e., in the range [0…180°[)) **D**, and $[i,j]$ is the position of the element θ_{ij} in the orientation image, then the Poincaré index $P_{\mathbf{G},C}(i,j)$ at $[i,j]$ is computed as follows.

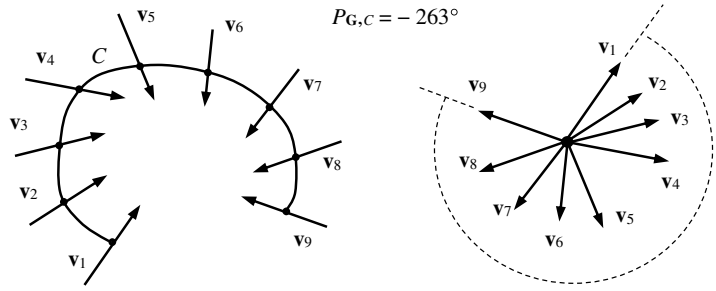

Fig. 3.24 The Poincaré index computed over a curve C immersed in a vector field **G**

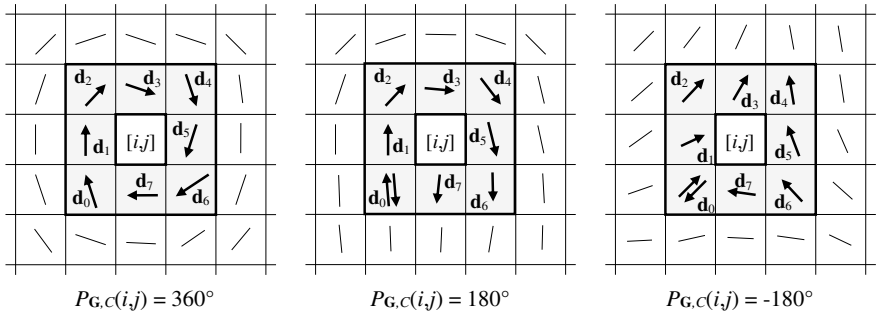

Fig. 3.25 Examples of the Poincaré index computation in the eight-neighborhood of points belonging (from left to right) to a whorl, loop, and delta singularity, respectively. Note that, for the loop and delta examples (center and right), the sense of $\mathbf{d}_0$ is first chosen upward (to compute the angle between $\mathbf{d}_0$ and $\mathbf{d}_1$) and then successively downward (when computing the angle between $\mathbf{d}_7$ and $\mathbf{d}_0$)

- The curve C is a closed path defined as an ordered sequence of some elements of $\mathbf{D}$, such that $[i, j]$ is an internal point.
- $P_{\mathbf{G},C}(i, j)$ is computed by algebraically summing the orientation differences between the adjacent elements of C. Summing orientation differences requires a *sense*[3] (among the two possible values) to be associated at each orientation, in order to transform it into a direction. A solution to this problem is to randomly select the sense of the first element and assign the direction closest to that of the previous element to each successive element. It is well known and can be easily shown that, on closed curves, the Poincaré index assumes only one of the discrete values: $0°, \pm 180°$, and $\pm 360°$. In the case of fingerprint singularities,

$$P_{\mathbf{G},C}(i,j) = \begin{cases} 0° & \text{if } [i,j] \text{ does not belong to any singular region} \\ 360° & \text{if } [i,j] \text{ belongs to a whorl type singular region} \\ 180° & \text{if } [i,j] \text{ belongs to a loop type singular region} \\ -180° & \text{if } [i,j] \text{ belongs to a delta type singular region} \end{cases}$$

Figure 3.25 shows three portions of the orientation image. The path defining C is the ordered sequence of the eight elements $\mathbf{d}_k$ $(k = 0, \ldots, 7)$ surrounding $[i, j]$. The direction of the elements $\mathbf{d}_k$ is chosen as follows: $\mathbf{d}_0$ is directed upward; $\mathbf{d}_k$ $(k = 1, \ldots, 7)$ is directed so that the absolute value of the angle between $\mathbf{d}_k$ and $\mathbf{d}_{k-1}$ is less than or equal to 90°. The Poincaré index is then computed as

[3] The sense of a vector is specified by the order of two points on a line parallel to the vector. Orientation and sense together determine the direction of a vector.

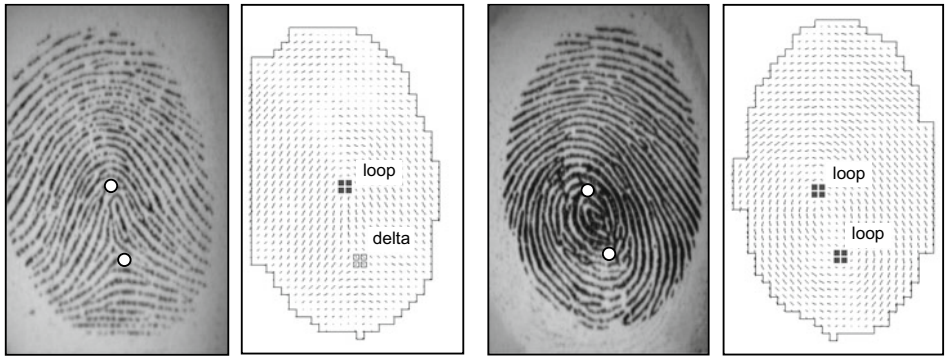

Fig. 3.26 Singularity detection using the Poincaré index for two images. The elements whose Poincaré index is 180° (loop) or −180° (delta) are enclosed by small boxes. Usually, more than one point (four points in these examples) is found for each singular region: hence, the center of each singular region can be defined as the barycenter of the corresponding points. Note that the position of the loop singularities is slightly moved toward the borders because of the local smoothing of the orientation image

$$P_{\mathbf{G},C}(i,j) = \sum_{k=0\ldots7} \text{angle}\big(\mathbf{d}_k, \mathbf{d}_{(k+1) \bmod 8}\big).$$

Some examples of singularities detected by the above method are shown in Fig. 3.26.

An interesting implementation of the Poincaré method for locating singular points was proposed by Bazen and Gerez (2002): according to Green's theorem, a closed line integral over a vector field can be calculated as a surface integral over the rotation of this vector field; in practice, instead of summing angle differences along a closed path, the authors compute the "rotation" of the orientation image (through a further differentiation) and then perform a local integration (sum) in a small neighborhood of each element. Bazen and Gerez (2002) also provided a method for associating an orientation with each singularity; this is done by comparing the orientation image around each detected singular point with the orientation image of an ideal singularity of the same type.

Singularity detection in noisy or low-quality fingerprints is difficult and the Poincaré method may lead to the detection of false singularities (Fig. 3.27). Regularizing the orientation image through a local averaging, as discussed in Sect. 3.3.4, is often quite effective in preventing the detection of false singularities, even if it can lead to the slight displacement of the loop position toward the borders. Wang et al. (2007b) propose a posteriori correction of loop location to compensate for the offset introduced by fingerprint image smoothing.

Based on the observation that only a limited number of singularities can be present in a fingerprint, Karu and Jain (1996) proposed to iteratively smooth the orientation image (through averaging) until a valid number of singularities is detected by the Poincaré index.

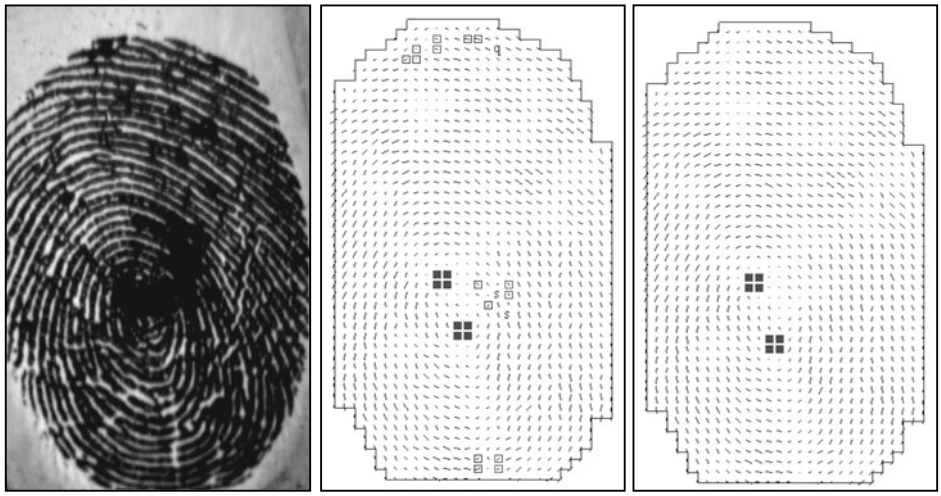

Fig. 3.27 **a** A poor-quality fingerprint; **b** the singularities of the fingerprint in (**a**) are extracted through the Poincaré method (circles highlight the false singularities); and **c** the orientation image has been regularized and the Poincaré method no longer provides false alarms

In fact, a simple analysis of the different fingerprint classes (refer to Chap. 5 for a more detailed discussion) shows the following:

- Arch fingerprints do not contain singularities.
- Left loop, right loop, and tented arch fingerprints contain one loop and one delta.
- Whorl fingerprints contain two loops (or one whorl) and two deltas.

The above constraints are nicely demonstrated by Zhou et al. (2007) who conclude that for each completely captured fingerprint, the number of loops and deltas are identical. A practical way to enforce this constraint is to compute the Poincaré index along the external boundary of the orientation image and then use the resulting value to limit the number of valid configurations. Another useful suggestion given by Zhou et al. (2007) is to locally change the path C for the computation of the Poincaré index according to the reliability of the underlying elements; in fact, $P_{\mathbf{G},C}$ being independent of the closed path C, if the eight-neighborhood path (shown in Fig. 3.25) includes unreliable elements, C can be progressively enlarged to include more reliable elements. In their approach, Zhou et al. (2009a) use the Poincaré index in conjunction with another similar operator, called DORIC, that looks for the presence of a single peak in the sequence of direction differences $\mathbf{d}_k$, $\mathbf{d}_{k+1}$; in Fig. 3.25, it can be noted that for loop and delta singularities, there is a single sharp change of direction between $\mathbf{d}_7$ and $\mathbf{d}_0$. After the Poincaré-based detection and DORIC-based filtering, Zhou et al. (2009a) select the optimal subset of

singular points as the set S that minimizes the difference between the estimated orientation image and a reconstruction of the orientation image through the Sherlock and Monro (1993) model with S as the set of singularities.

Finally, the spatial distributions of singularity locations in nature (Cappelli & Maltoni, 2009) can be exploited to guide singularity extraction or implement post-filtering approaches.

3.5.2 Methods Based on Local Characteristics of the Orientation Image

Some authors have proposed singularity detection approaches where the fundamental idea is to explore the orientation image regions characterized by high irregularity, curvature, or symmetry. In fact, the singularities are the only foreground regions where a dominant orientation does not exist and ridges assume high curvature.

The coherence operator, as defined by Eq. (3.4), was used by Cappelli et al. (1999) to find approximate locations of singular regions. Figure 3.28 shows an example of a coherence map computed over 3×3 neighborhoods.

Srinivasan and Murthy (1992) extract singularities according to the local histogram of the orientation image; in fact, the points where the local histogram does not exhibit a well-pronounced peak are likely to belong to singular regions. By analyzing the predominant orientation in some predefined sectors around the candidate points, Srinivasan and Murthy were able to discriminate between loop and delta singularities.

Chen et al. (2011b) compute local orientation entropy at three different quantization levels and combine these estimates at each position (by simple multiplication) to improve singularity detection robustness against noise.

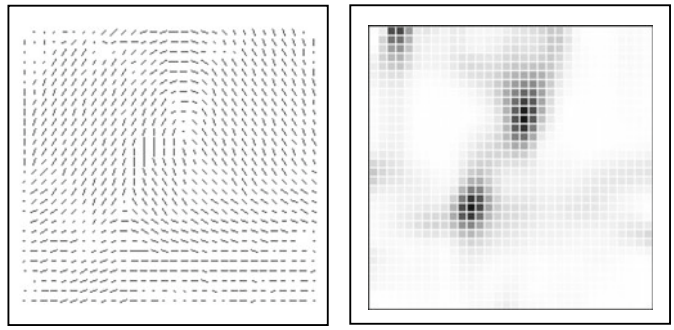

Fig. 3.28 An orientation image and the corresponding coherence map that clearly identifies the regions (dark cells $\leftrightarrow$ low coherence) containing the singularities

The Local Axial Symmetry (LAS) introduced by Liu et al. (2006) is a pixel-wise feature, derived from the orientation image that denotes the symmetry of the orientations inside circular regions centered at each pixel. Based on the observation that there are two regions with lower LAS value on the two opposite sides of each loop-type singularity, a simple binarization scheme is proposed to isolate these two regions and locate the loop(s) as midpoint(s) of their barycenters. A similar symmetry-based technique was used by Liu et al. (2005) to check the validity of singular points (both loop and delta) initially detected through the Poincaré index approach.

Zhu et al. (2016) proposed to derive Walking Directional Fields (WDFs) from the orientation map. Since WDF local vectors point to the singularity centers, a navigation algorithm can be designed to detect singularities without processing the entire orientation map.

Some researchers argued that multi-resolution analysis is an effective tool for singularity detection since it allows (i) to increase robustness against the noise if the same singularities are detected at different resolution; (ii) to locate singular points with sub-block accuracy, and (iii) to reduce computational costs when a coarse-to-fine approach is adopted:

- Koo and Kot (2001) detect the singularities with single pixel accuracy. At each resolution, they derive from the orientation image a curvature image whose blocks indicate the local degree of ridge curvature. High-curvature blocks, which denote singularities, are retained and analyzed at finer resolutions.
- Nilsson and Bigun (2002a, b, 2003) multi-resolution analysis is based on the convolution of the orientation field with two complex filters tuned to detect points of rotational symmetry. The two filters (one for loop-type singularity and one for delta-type singularity) are convolved with orientation image and the points where the response of any one filter is high are retained and analyzed at a finer resolution (see Fig. 3.29). Orientation of the detected singularities can also be estimated from the argument of the complex filter responses. This technique can be implemented in an efficient way thanks to the separability of the filters used. An improved version of this approach is described in Chikkerur and Ratha (2005).
- Liu et al. (2004) approach iteratively checks for the existence of singularities and refines their position over a sequence of resolutions, based on local features such as the Poincaré index and orientation variance.
- Wang et al. (2007b) multi-resolution approach is based on harmonic relationships between the orientation of a singular point and its neighbors.
- In the approach by Weng et al. (2011), the candidate singularities detected by an improved Poincaré index method at different resolutions (varying block size and stride) are aggregated and post-filtered according to the orientation coherence.

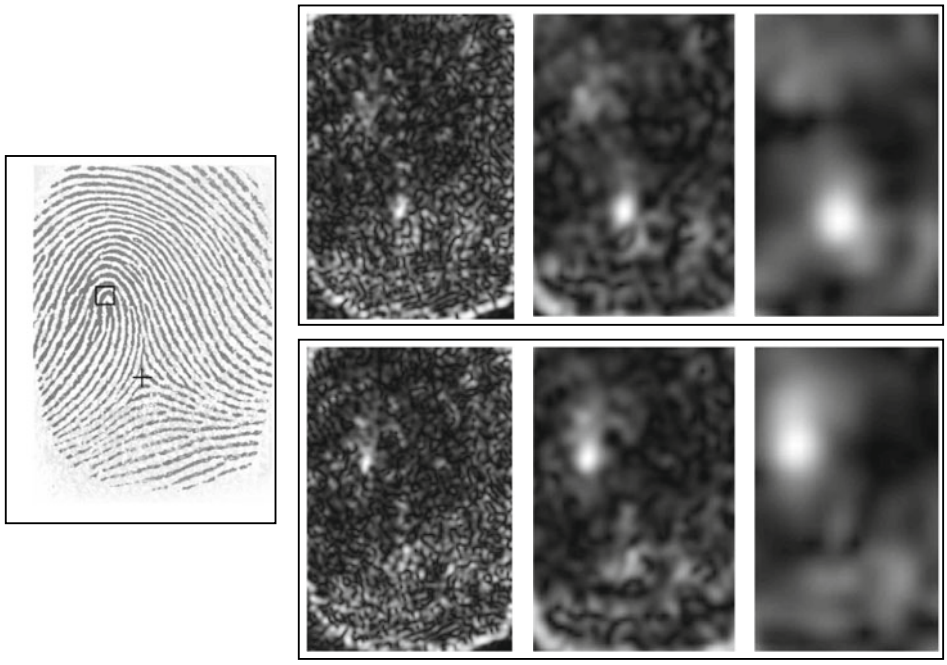

Fig. 3.29 An example of the Nilsson and Bigun (2003) approach. A fingerprint image and its responses of the loop filter (bottom row) and delta filter (top row) at three different scales are shown. Images courtesy of J. Bigun

3.5.3 Partitioning-Based Methods

Some authors noted that partitioning the orientation image in regions characterized by homogeneous orientation implicitly reveals the position of singularities. Hung and Huang (1996) and Huang et al. (2007) coarsely discretize the orientation image by using a very small number of orientation values. Each orientation value determines a region. The borderline between two adjacent regions is called a fault line. By noting that fault lines converge toward loop singularities and diverge from deltas, the authors define a geometrical method for determining the convergence and divergence points. In Maio and Maltoni (1996) and Cappelli et al. (1999), the orientation image is partitioned by using an iterative clustering algorithm and a set of dynamic masks, respectively (Fig. 3.30). Ohtsuka and Takahashi (2005), Ohtsuka and Kondo (2005), and Ohtsuka and Watanabe (2010) derive the position of singularities from the extended relational graphs modeling the orientation image. Rämö et al. (2001) and Kryszczuk and Drygajlo (2006) implicitly partition the orientation image in correspondence with the points where the x- and y-orientation components (refer to Eq. (3.1)) change sign: efficient methods are then introduced to extract singularities based on the contemporary zero crossings of both orientation components.

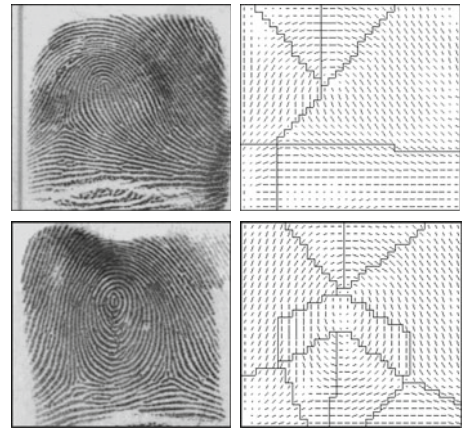

Fig. 3.30 Orientation image partitioning with the MASK approach (Cappelli et al., 1999). The intersections between region border lines denote fingerprint singularities. © IEEE. Reprinted, with permission, from Cappelli et al. (1999)

3.5.4 Methods Based on a Global Model of the Orientation Image

An effective strategy to improve robustness is to exploit a global model of the orientation image (see Sect. 3.3.5):

- Wu and Zhou (2004) used the zero–pole model and the Hough transform to detect the position of singularities starting from the orientation image. In the zero–pole model, the orientation around each singularity is not much influenced by the other singularities, and therefore, within a local window, the singularities' position and the surrounding orientations are bound by simple linear equations. This makes it possible to implement the voting scheme which is the basis of the Hough transform. An initial location of the singularities is performed with the Poincaré approach; each singularity is then checked and its location refined through the Hough transform.
- Fan et al. (2008) approach is also based on the zero–pole model and the Hough transform. The robustness of this method to identify a single dominant loop (or delta) for a given fingerprint area is demonstrated for real fingerprint images (including noisy samples). However, to refine the position of singularities and to deal with the presence of more loops/deltas, other heuristics based on the Poincaré index and ridge tracing are used in conjunction with the zero–pole model and the Hough transform. See Fig. 3.31 for an example.
- Dass (2004) used the simultaneous computation of orientation image and singularities that is briefly discussed in Sect. 3.3.5.
- Wang et al. (2007a) exploited their global model (FOMFE) to obtain an analytical expression of the local orientation topology: in particular, the classification of each point as {normal point, loop or delta} is determined by the sign of the determinant of

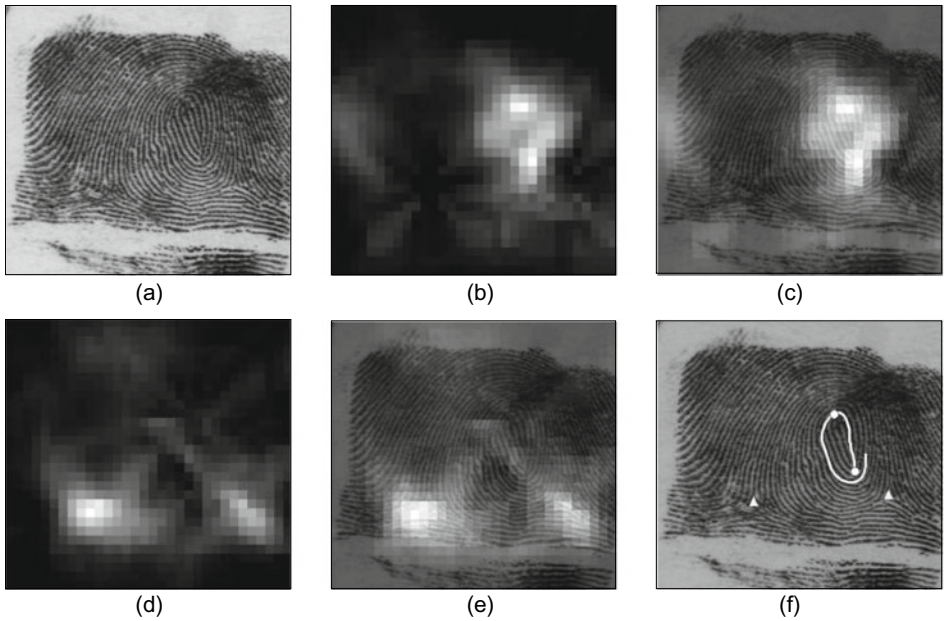

Fig. 3.31 An example of singularity detection with Fan et al. (2008) approach. **a** Original image; **b** Hough space (loop); **c** Hough space (loop) superimposed over the original image; **d** Hough space (delta); **e** Hough space (delta) superimposed over the original image; and **f** detected loops (circle) and deltas (triangle), ridge tracing is also shown. © IEEE. Reprinted, with permission, from Fan et al. (2008)

a 2×2 matrix, called the characteristic matrix. The computation of additional information such as the local curl and divergence assists in implementing effective rules for the removal of false singular points (Wang & Hu, 2008).

3.5.5 Fingerprint Pose Estimation

Fingerprint pose estimation in two dimensions consists in the determination of a (stable) reference point and a direction. The availability of such information allows to shift/rotate fingerprint images in order to achieve an *absolute registration* (or *absolute pre-alignment*). This can be very useful to extract pose independent (fixed-length) features which are needed by many fingerprint indexing methods (Chap. 5) and matching techniques in the encrypted domain (Chap. 9). Note that absolute registration discussed in this section differs from relative registration addressed in Chap. 4; in fact, relative registration is aimed

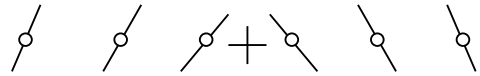

Fig. 3.32 The core point "+" located on the chosen sextet

at computing the displacement/rotation between a pair of fingerprints while absolute registration operates on a single fingerprint.

Reference Point Detection

The fingerprint core is a natural candidate reference point. Once the singularities have been extracted, the core position may be simply defined as the location of the northernmost loop. There is an ambiguity with the arch-type fingerprints that do not have singularities; for these fingerprints, the estimated flow field is flat and hence no core position can be located. When the core point is detected with the aim of registering fingerprint images (thus obtaining invariance with respect to (x, y) displacement), the core's location may be quite critical and an error at this stage often leads to a failure of subsequent processing steps (e.g., matching). On the other hand, if the core has to be used only for fingerprint registration, it is not important to find the northernmost loop exactly and any stable point in the fingerprint pattern is suitable.

One of the first automated methods for fingerprint registration was proposed by Wegstein (1982). This method, known as R92, searches for a core point independently of the other singularities. The core is searched by scanning (row by row) the orientation image to find *well-formed arches*; a well-formed arch is denoted by a sextet (set of six) of adjacent elements whose orientations comply with predefined rules. One sextet is chosen among the valid sextets by evaluating the orientation of the elements in adjacent rows. The exact core position is then located through interpolation (Fig. 3.32). Even though R92 is quite complicated and heuristic in nature, it usually gives good results and is able to localize the core point with sub-block accuracy. This algorithm was a fundamental component of the fingerprint identification systems used by the FBI.

Several other ideas for the location of stable registration points have been proposed. Novikov and Kot (1998) define the core as the crossing point of the lines normal to the ridges (Fig. 3.33) and used the Hough transform (Ballard, 1981) to determine its

Fig. 3.33 The straight lines normal to the ridges identify a valid registration point that corresponds to the center of curvature

coordinates. Similarly, Rerkrai and Areekul (2000) define the *focal point* as the point where pairs of straight lines normal to the ridges intersect. Because the ridges do not draw perfect concentric circumferences around the core, the normal lines (dashed lines in Fig. 3.33) do not exactly cross at a single point and a sort of average point has to be defined as the center of curvature. Novikov and Kot (1998) compute this average point in the least squares sense, whereas Rerkrai and Areekul (2000) compute the barycenter of the crossing between pairs of normals.

A robust voting schema based on the Hough transform was proposed by Yang et al. (2014). In this approach, votes are cast by individual 4×4 orientation patches. During an off-line training stage, orientation patches are clustered. Then, for each cluster, a statistical model is learnt to denote the relative displacement between the patch and the fingerprint center. During online center computation, each patch is mapped to the most similar cluster and the corresponding probability map is summed to the Hough image. At the end of the voting, the maximum in the Hough image provides a robust estimation of the center position. More details on this method, which was specifically introduced for latent fingerprints, are provided in Sect. 6.4.2. The pose estimation method by Gu et al. (2017) also relies on voting but in their Hough forest-based approach (Gall & Lempitsky, 2013), multiple trees are learnt and their votes are combined to improve robustness.

Although the focal point (or the center of curvature) does not necessarily correspond to the core point, it has been experimentally demonstrated to be quite stable with respect to fingerprint variation (displacement, rotation, distortion, etc.). Therefore, it can be reliably used for fingerprint registration. The main problem of these methods is in isolating a fingerprint region characterized by a single center of curvature. In fact, if the selected fingerprint region contains more than one singularity, the result may be unpredictable. To solve this problem, Areekul et al. (2006) proposed an algorithm that separately determines a focal point for each neighbor of a high curvature point and then tries to reach a consensus.

The focal point is also used in the method by Deerada et al. (2020) aimed at determining the pose of latent fingerprints: since orientation estimation in latent fingerprints is not reliable enough to be used for direct focal point computation, the authors proposed a closed-loop approach that progressively locates potential poses and enhances weak friction ridges that form and support these poses through an iterative feedback.

Jain et al. (2000) proposed a multi-resolution approach for locating the northmost loop-type singularities (core) based on the integration of sine components in two adjacent regions R_I and R_{II} (Fig. 3.34). The geometry of the two regions is designed to capture the maximum curvature in concave ridges. At each scale and for each candidate position $[x, y]$, the sine components of the orientation image are integrated over the two regions resulting in the values SR_I and SR_{II}. The points $[x, y]$ that maximize the quantity $(SR_I–SR_{II})$ are retained as candidate positions and analyzed at a finer resolution.

Another interesting multi-resolution approach, designed by Jiang et al. (2004), performs the core localization by means of a hierarchical analysis of orientation coherence:

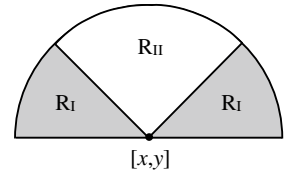

Fig. 3.34 Regions of integration of the sine components in the method proposed by Jain et al. (2000).

the core is chosen as the point having local minimum coherence at both large and fine scales. This approach was able to correctly locate the core position in about 95% of the FVC2000 DB2 fingerprint images (Maio et al., 2002). An improvement of the above technique, where the orientation coherence is combined with the direction of curvature, was proposed by Van and Le (2009a).

Inspired by the filtering approaches proposed by Nilsson and Bigun (2003) and Chikkerur and Ratha (2005) for singularities detection (see Sect. 3.5.2), Le and Van (2012) designed a new complex filter to detect the core point based on orientation symmetries. Compared to the parabolic symmetry filter of Nilsson and Bigun (2003), the new filter also works with arch fingerprints and is able to deal better with double loop prints. A second operator, based on vertical orientation variation, is used to filter out unwanted singularities (e.g., the southmost loop in a double loop) or spurious detections. The authors report a core detection accuracy of 96.6% on FVC2004 DB1.

Finally, in the approach by Zacharias et al. (2017), core detection starts with generating chain code contours from the thinned input fingerprint image. Chain-coded contours of the fingerprint ridges are then smoothened to improve robustness against noise and their curvature is computed using a rotation-invariant curvature estimation method. The core point is then localized on the ridge that bends the most.

Reference Direction Computation

Absolute pre-alignment with respect to rotation can be straightforward in full-size and/or high-quality fingerprints but critical for low-quality and partial fingerprint images. Some authors proposed to use (i) the shape of the external fingerprint silhouette (if available), (ii) the direction of the core delta segment (if a delta exists), (iii) the average orientation in neighborhoods around the core (Fig. 3.35), or (iv) the directions of the singularities (Bazen & Gerez, 2002). A specific geometric technique was proposed by Li et al. (2014) to measure the fingerprint direction of arch fingerprints using a set of isosceles triangles. Hotz (2009) showed that the vertical symmetry axis can be easily determined in arch fingerprints while the disturbance induced by the singularities can negatively affect the computation in non-arch fingerprints; therefore, he proposed a model-based cancelation of singularities to obtain an arch-like pattern from fingerprints of all the classes.

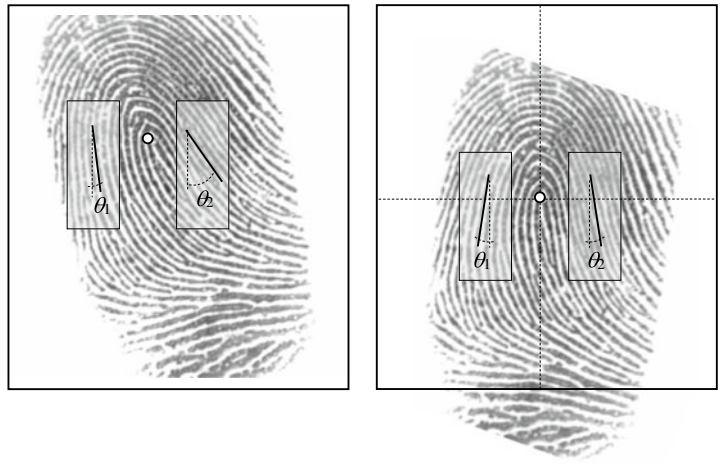

Fig. 3.35 The M82 method, developed for minutiae-based fingerprint matching in the AFIS community (Wegstein, 1982), performs a pre-alignment according to the core position and the average orientation of the two regions on the two sides of the core. The fingerprint on the right has been translated to move the core point to the image center and rotated to minimize the difference between the angles θ_1 and θ_2

Su et al. (2016) proposed to address fingerprint pose estimation as an object-detection problem. The finger detection method, which is based on the histogram of ridge orientation and support vector classifier, was used to compute both the center point and the direction of the fingerprint at the same time. This approach is shown to be effective on rolled fingerprints, but its application to plain (often partial) fingerprints remains challenging. Ouyang et al. (2017) proposed to use Faster R-CNN deep network model (Girshick, 2015) for fingerprint pose estimation and reported a better performance than Su et al. (2016). Another deep learning-based technique was proposed by Schuch et al. (2018a, b) where a Siamese CNN is trained to estimate the rotation between pairs of the sample. During training, a fingerprint is rotated by two random angles θ_1 and θ_2. The CNN estimates rotations θ'_1 and θ'_2, so that differences $(\theta_1 - \theta_2)$ and $(\theta'_1 - \theta'_2)$ are as similar as possible. While training takes place on fingerprints pairs, at inference time, the model receives a single input and provides its absolute direction.

Unlike singularities, fingerprint pose lacks a consistent definition, which makes it difficult to compare the methods of different research groups. A possible approach is to evaluate the pose consistency of different images of the same fingerprint (Su et al. 2016). Such evaluation is closely related to the application of pose in fingerprint indexing.

3.6 Enhancement

The performance of subsequent minutiae extraction algorithms and other fingerprint recognition techniques relies heavily on the quality of the input fingerprint images. In an ideal fingerprint image, ridges and valleys alternate and flow in a locally constant direction. In such situations, the ridges can be easily detected and minutiae can be precisely located in the image. Figure 3.36a shows an example of a good-quality fingerprint image. However, in practice, due to skin conditions (e.g., wet or dry, cuts, and bruises), sensor noise, finger pressure, and inherently low-quality fingertips (e.g., elderly people and manual workers), a significant percentage of fingerprint images (approximately 10%, according to our experience) is of poor quality like those in Figs. 3.36b, c. In many cases, a single fingerprint image contains regions of good, medium, and poor quality where the ridge pattern is very noisy and corrupted (Fig. 3.37). In general, there are several types of degradation associated with fingerprint images:

1. The ridges are not strictly continuous; that is, the ridges have small breaks (gaps).
2. Parallel ridges are not well separated. This is due to the presence of noise that links parallel ridges, resulting in their poor separation.
3. Cuts, creases, and bruises on the finger.

These three types of degradation make ridge extraction extremely difficult in the highly corrupted regions. This leads to the following problems in minutiae extraction: (i) a significant number of spurious minutiae are extracted, (ii) a large number of genuine minutiae are missed, and (iii) large errors in the location (position and orientation) of minutiae are introduced. In order to ensure good performance of the ridge and minutiae extraction

Fig. 3.36 **a** A good-quality fingerprint; **b** a medium-quality fingerprint characterized by scratches and ridge breaks; and **c** a poor-quality fingerprint containing a lot of noise

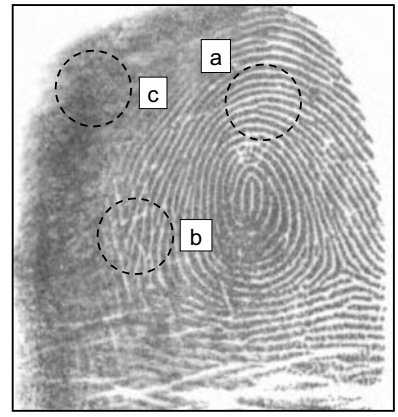

Fig. 3.37 A fingerprint image containing regions of different quality: **a** a well-defined region, **b** a recoverable region, and **c** an unrecoverable region

algorithms in poor-quality fingerprint images, an enhancement algorithm to improve the clarity of the ridge structure is necessary.

A human fingerprint expert is often able to correctly identify the minutiae by using various visual clues such as local ridge orientation, ridge continuity, ridge tendency, and so on. In theory, it is possible to develop an enhancement algorithm that exploits these visual clues to improve image quality. Generally, for a given fingerprint image, the fingerprint areas resulting from the segmentation step may be divided into three categories (Fig. 3.37):

- *Well-defined region*: ridges can be clearly differentiated from each other.
- *Recoverable region*: ridges are corrupted by a small amount of gaps, creases, smudges, links, and the like, but they are still visible and the neighboring regions provide sufficient information about their true structure.
- *Unrecoverable region*: ridges are corrupted by such a severe amount of noise and distortion that no ridges are visible and the neighboring regions do not allow them to be reconstructed.

Good-quality regions, recoverable regions, and unrecoverable regions may be identified according to several criteria; in general, image contrast, orientation consistency, ridge frequency, and other local features may be combined to define a quality index. Since the estimation of fingerprint quality is central for a number of algorithms and practical applications, a section devoted to quality computation is provided at the end of this chapter. The goal of an enhancement algorithm is to improve the clarity of the ridge structures in the recoverable regions and mark the unrecoverable regions as too noisy for further processing. Usually, the input of the enhancement algorithm is a gray-scale image. The output may either be a gray-scale or a binary image, depending on the algorithm and goal.

A particular case of enhancement is super-resolution which consists in increasing the native resolution of a fingerprint image. If a fingerprint is captured at low resolution

(e.g., with a low-cost mobile sensor) with a resolution of 250 dpi, then its ridges are difficult to separate and minutiae detection is prone to errors. While a simple upscaling by interpolation leads to a blurred representation, effective super-resolution approaches have been proposed based on sparse representation by patch dictionaries (Singh et al., 2015 and Bian et al., 2017b).

3.6.1 Pixel-Wise Enhancement

In a pixel-wise operation, the new value of each pixel only depends on the previous value of that pixel and some global parameters (but not on the value of the neighboring pixels). Pixel-wise techniques do not produce satisfying results for fingerprint image enhancement. However, contrast stretching, histogram manipulation, normalization (Hong et al., 1998), and Wiener filtering (Greenberg et al., 2000) have been shown to be effective as initial processing steps in a more sophisticated fingerprint enhancement algorithm.

The normalization approach used by Hong et al. (1998) determines the new intensity value of each pixel in an image as

$$\mathbf{I}'[x, y] = \begin{cases} m_0 + \sqrt{(\mathbf{I}[x, y] - m)^2 \cdot v_0 / v} & \text{if } \mathbf{I}[x, y] > m \\ m_0 - \sqrt{(\mathbf{I}[x, y] - m)^2 \cdot v_0 / v} & \text{otherwise,} \end{cases} \tag{3.9}$$

where m and v are the image mean and variance, respectively, and m_0 and v_0 are the corresponding mean and variance after the normalization. Figure 3.38 shows an example. Since the mean and variance can change in different regions of a fingerprint image, the above global technique can be implemented in a local fashion: Kim and Park (2002) introduced a block-wise implementation of Eq. (3.9) where m and v are the block mean

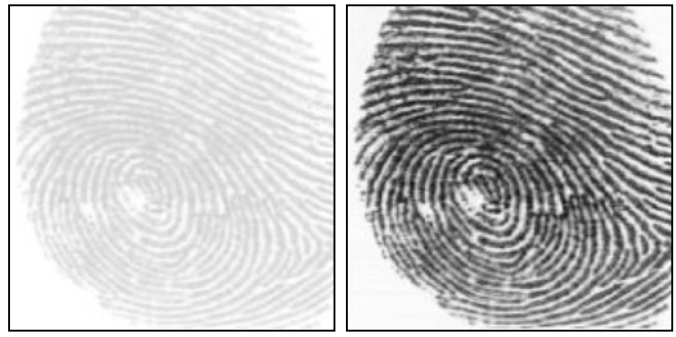

Fig. 3.38 An example of normalization with the method described in Hong et al. (1998) using ($m_0 = 100$, $v_0 = 100$). © IEEE. Reprinted, with permission, from Hong et al. (1998)

and variance, respectively, and m_0 and v_0 are adjusted for each block according to the block features. A similar adaptive normalization was proposed by Shi and Govindaraju (2006b). However, this kind of normalization involves pixel-wise operations and does not change the ridge and valley structures. In particular, it is not able to fill small ridge breaks, fill intra-ridge holes, or separate parallel touching ridges.

A contrast enhancement scheme was proposed by Hari et al. (2013) where a high-pass filtered and scaled version of an image is added with itself. A quadratic filter, optimized for fingerprint images, is shown to be more effective than Laplacian and Laplacian of Gaussian filters.

Sharma and Dey (2019) proposed to adapt the preprocessing stage to the fingerprint image quality. For this purpose, fingerprints are clustered into five classes {dry, wet, normal dry, normal wet, or good} and the parameters of the preprocessing algorithm (in particular, the image sharpening) are tuned accordingly.

3.6.2 Contextual Filtering

The most widely used technique for fingerprint image enhancement is based on *contextual filters*. In conventional image filtering, only a single filter is used for convolution throughout the image. In contextual filtering, the filter characteristics change according to the local context. Usually, a set of filters is pre-computed and one of them is selected for each image region. In fingerprint enhancement, the context is often defined by the local ridge orientation and local ridge frequency. In fact, the sinusoidal-shaped wave of ridges and valleys is mainly defined by a local orientation and frequency that varies slowly across the fingerprint area. An appropriate filter that is tuned to the local ridge frequency and orientation can efficiently remove the undesired noise and preserve the true ridge and valley structure.

Several types of contextual filters have been proposed in the literature for fingerprint enhancement. Although they have different definitions, the intended behavior is almost the same: (1) provide a low-pass (averaging) effect along the ridge direction with the aim of linking small gaps and filling impurities due to pores or noise and (2) perform a bandpass (differentiating) effect in the direction orthogonal to the ridges to increase the discrimination between ridges and valleys and to separate parallel linked ridges.

The output of a contextual fingerprint enhancement can be a gray-scale, near-binary, or binary image, depending on the filter parameters chosen. When selecting the appropriate set of filters and tuning their parameters, one should keep in mind that the goal is not to produce a good visual appearance of the image but to facilitate the robustness of the successive feature extraction steps. Since in poor-quality regions the estimation of the local context (orientation and frequency) may be erroneous, a too aggressive filtering is likely to produce spurious structures (Jiang, 2001). For example, an iterative application

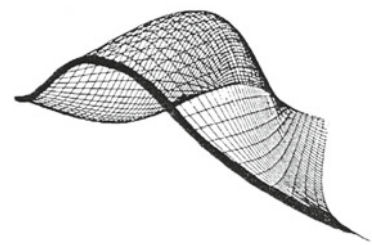

Fig. 3.39 The shape of the filter proposed by O'Gorman and Nickerson (1989). © Elsevier. Reprinted, with permission, from O'Gorman and Nickerson (1989)

of the Gabor filters has been used by Cappelli et al. (2000) (refer to Chap. 7) to generate a synthetic fingerprint pattern; in this case, the filters generate nonexistent ridge patterns.

Contextual Filtering in the Spatial Domain

The method proposed by O'Gorman and Nickerson (1988, 1989) was one of the first to use contextual filtering for fingerprint enhancement; the authors defined a "mother filter" based on four main parameters of fingerprint images at a given resolution: minimum and maximum ridge width, and minimum and maximum valley width. The filter is bell-shaped (see Fig. 3.39), elongated along the ridge direction, and cosine tapered in the direction normal to the ridges. The local ridge frequency is assumed constant, and therefore, the context is defined only by the local ridge orientation. Once the mother filter has been generated, a set of 16 rotated versions (in steps of 22.5°) is derived. The image enhancement is performed by convolving each point of the image with the filter in the set whose orientation best matches the local ridge orientation. Depending on some input parameters, the output image may be gray-scale or binary. Examples of image binarization using this technique are shown in Figs. 3.49b and 3.50a.

Hong et al. (1998) proposed an effective method based on the Gabor filters, which today is still one of the most popular techniques for fingerprint enhancement. The Gabor filters have both frequency-selective and orientation-selective properties and have the optimal joint resolution in both spatial and frequency domains (Daugman, 1985; Jain & Farrokhnia, 1991). As shown in Fig. 3.40, a Gabor filter is defined by a sinusoidal plane wave (the second term of Eq. (3.10)) tapered by a Gaussian (the first term in Eq. (3.10)). The even symmetric two-dimensional Gabor filter has the following form.

$$g(x, y{:}\theta, f) = \exp\left\{-\frac{1}{2}\left[\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right]\right\} \cdot \cos(2\pi f \cdot x_\theta), \tag{3.10}$$

where θ is the orientation of the filter, and $[x_\theta, y_\theta]$ are the coordinates of $[x, y]$ after a clockwise rotation of the Cartesian axes by an angle of (90°–θ).

$$\begin{bmatrix} x_\theta \\ y_\theta \end{bmatrix} = \begin{bmatrix} \cos(90^\circ - \theta) & \sin(90^\circ - \theta) \\ -\sin(90^\circ - \theta) & \cos(90^\circ - \theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \sin\theta & \cos\theta \\ -\cos\theta & \sin\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

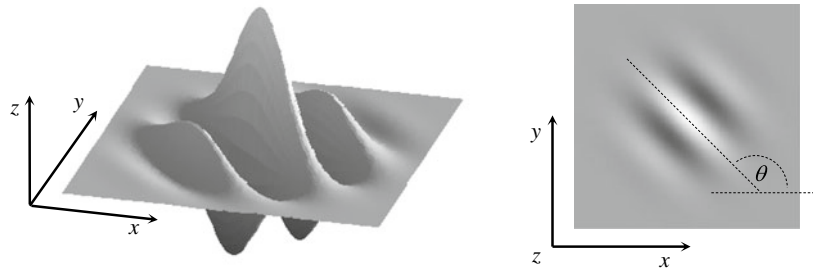

Fig. 3.40 Graphical representation (lateral view and top view) of the Gabor filter defined by the parameters $\theta = 135°, f = 1/5$, and $\sigma_x = \sigma_y = 3$

In the above expressions, f is the frequency of a sinusoidal plane wave, and σ_x and σ_y are the standard deviations of the Gaussian envelope along the x- and y-axes, respectively.

To apply the Gabor filters to an image, the four parameters (θ, f, σ_x, and σ_y) must be specified. Obviously, the frequency of the filter is completely determined by the local ridge frequency and the orientation is determined by the local ridge orientation. The selection of the values σ_x and σ_y involves a trade-off. The larger the values, the more robust the filters are to the noise in the fingerprint image, but they are also more likely to create spurious ridges and valleys. On the other hand, the smaller the values, the less likely the filters are to introduce spurious ridges and valleys but then they will be less effective in removing the noise. In fact, from the Modulation Transfer Function (MTF) of the Gabor filter, it can be shown that increasing σ_x and σ_y decreases the bandwidth of the filter and vice versa. Based on empirical data, Hong et al. (1998) set $\sigma_x = \sigma_y = 4$. To make the enhancement faster, instead of computing the best-suited contextual filter for each pixel "on the fly," a set $\{g_{ij}(x, y) | \, i = 1, \ldots, n_o, j = 1, \ldots, n_f\}$ of filters is a priori created and stored, where n_o is the number of discrete orientations $\{\theta_i \mid i = 1, \ldots, n_o\}$ and n_f denotes the number of discrete frequencies $\{ f_j \mid j = 1, \ldots, n_f \}$. Then each pixel $[x, y]$ of the image is convolved, in the spatial domain, with the filter $g_{ij}(x, y)$ such that θ_i is the discretized orientation closest to θ_{xy} and f_j is the discretized frequency closest to f_{xy}. Figure 3.41 shows an example of the filter set for $n_o = 8$ and $n_f = 3$. Figure 3.42 shows the application of the Gabor-based contextual filtering on medium- and poor-quality images.

Greenberg et al. (2000) noted that by reducing the value of σ_x with respect to σ_y, the filtering creates fewer spurious ridges and is more robust to noise. In practice, reducing σ_x results in increasing the frequency bandwidth, independently of the angular bandwidth which remains unchanged; this allows the filter to better tolerate errors in local frequency estimates. Analogously, one could decrease σ_y in order to increase the angular bandwidth as pointed out by Sherlock et al. (1994). Their method increases the angular bandwidth near the singularities where the ridges are characterized by higher curvatures and the orientation changes rapidly. The methods by Erol et al. (1999); Wu and Govindaraju (2006); Van and Le (2009b) relate the filter bandwidth to the local orientation coherence,

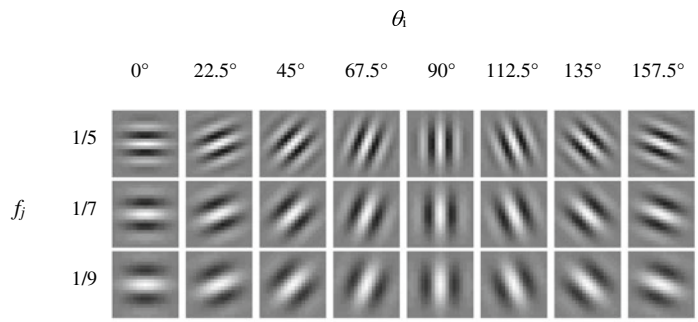

Fig. 3.41 A graphical representation of a bank of 24 ($n_o = 8$ and $n_f = 3$) Gabor filters where $\sigma_x = \sigma_y = 4$

whereas the Bernard et al. (2002) approach reduces the filter bandwidth if none of the responses to an initial set of filters exceeds a certain threshold. Yang et al. (2003) argue that the fingerprint ridge and valley pattern does not always resemble a pure sinusoidal pattern, mainly because of the different values of ridge and valley width in some regions (see Fig. 3.43). Therefore, they propose the Gabor-like filters whose positive and negative peaks can have different periods and contextually adjust the two periods based on the local ridge width and local valley width, respectively. Zhu et al. (2004) note that implementing the Gabor-based contextual filtering with a squared mask can lead to artifacts that can be removed if the mask support is circular.

Wang et al. (2008) suggest replacing the standard Gabor filter with the Log-Gabor filter to overcome the drawbacks that the maximum bandwidth of a Gabor filter is limited to approximately one octave and the Gabor filters are not optimal if one is seeking broad spectral information with maximal spatial localization. Curved Gabor filters have been introduced by Gottschlich (2012) to better enhance the fingerprint pattern in high curvature regions.

For low-cost and computation-limited fingerprint systems (e.g., embedded systems), the two-dimensional convolution of an image with a Gabor filter pre-computed over a discrete mask (e.g., 15×15) can be too time-consuming. The computational complexity can be reduced by using separable Gabor filters (Areekul et al., 2005) or masks with sparse coefficients (Jang et al., 2006).

Contextual Filtering in the Frequency Domain

Sherlock et al. (1992, 1994) performed contextual filtering in the Fourier domain; in fact, it is well known that a convolution in the spatial domain corresponds to a point-by-point complex multiplication in the Fourier domain (Gonzales & Woods, 2007). The filter is defined in the frequency domain by the function:

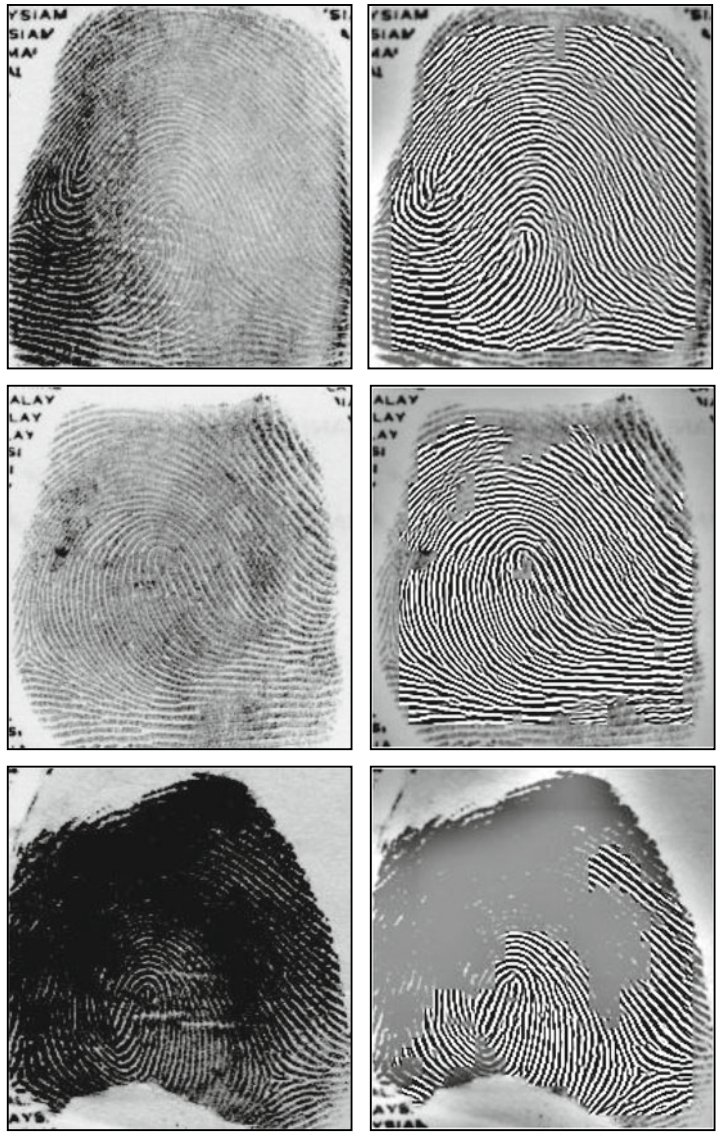

Fig. 3.42 Examples of fingerprint enhancement with the Gabor filtering as proposed by Hong et al. (1998). On the right, the enhanced recoverable regions are superimposed on the corresponding input images. © IEEE. Reprinted, with permission, from Hong et al. (1998)

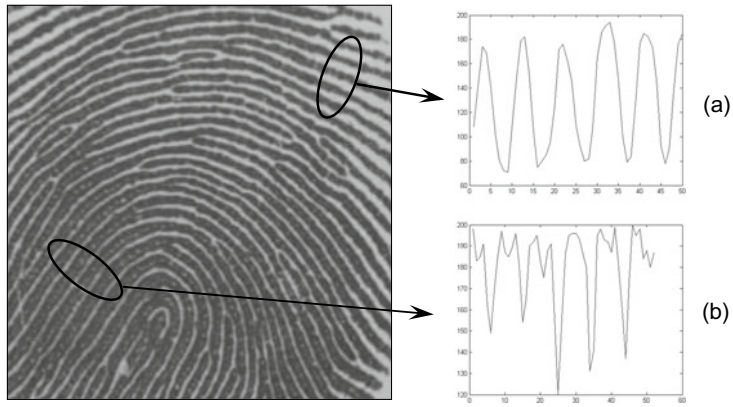

Fig. 3.43 Two examples of fingerprint regions where the local ridge–valley pattern conforms to **a** and deviates from **b** a sinusoidal pattern. © Elsevier. Reprinted, with permission, from Yang et al. (2003)

$$H(\rho, \theta) = H_{\text{radial}}(\rho) \cdot H_{\text{angle}}(\theta), \tag{3.11}$$

where H_{radial} depends only on the local ridge spacing $\rho = 1/f$ and H_{angle} depends only on the local ridge orientation θ. Both H_{radial} and H_{angle} are defined as bandpass filters and are characterized by a mean value and a bandwidth. A set of n discrete filters is derived by their analytical definition. To reduce the number of filters, only a single value is used for the local ridge frequency, and, therefore, the context is determined only by the orientation. The Fourier transform $\mathbf{P}_i$, $i = 1 \ldots n$ of the filters is pre-computed and stored. Filtering an input fingerprint image $\mathbf{I}$ is performed as follows (see Fig. 3.44).

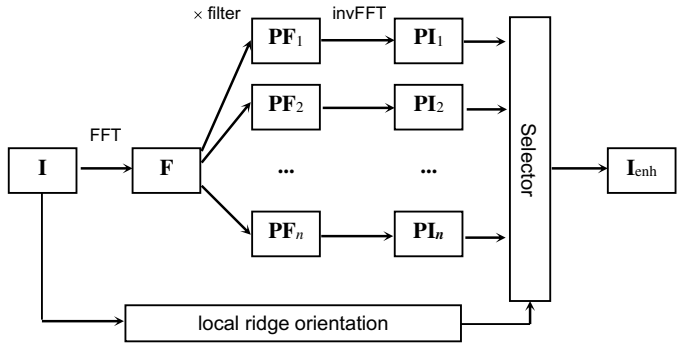

Fig. 3.44 Enhancement of the fingerprint image $\mathbf{I}$ according to the Sherlock et al. (1994) method

- The FFT (Fast Fourier Transform) $\mathbf{F}$ of $\mathbf{I}$ is computed.
- Each filter $\mathbf{P}_i$ is point-by-point multiplied by $\mathbf{F}$, thus obtaining n filtered image transforms $\mathbf{PF}_i$, $i = 1 \ldots n$ (in the frequency domain).
- Inverse FFT is computed for each $\mathbf{PF}_i$ resulting in n filtered images $\mathbf{PI}_i$, $i = 1 \ldots n$ (in the spatial domain).

The enhanced image $\mathbf{I}_{enh}$ is obtained by setting, for each pixel $[x,y]$, $\mathbf{I}_{enh}[x, y] = \mathbf{PI}_k[x, y]$, where k is the index of the filter whose orientation is the closest to θ_{xy}.

Chikkerur et al. (2007) proposed an efficient implementation of contextual filtering based on Short-Time Fourier transform (STFT) that requires partitioning the image into small overlapping blocks and performing the Fourier analysis separately on each block. The orientation and frequency of each block are probabilistically determined through Eqs. (3.6) and (3.8), and the orientation coherence is computed similar to Eq. (3.4). Each block is then filtered (by complex multiplication in the Fourier domain) with a filter equivalent to Eq. (3.11) except for the angular bandwidth which is adjusted according to the orientation coherence; in Sherlock et al. (1994), the angular bandwidth is related to the distance from the closest singular point. Since singular point estimation is less robust than coherence estimation, the Chikkerur et al. (2007) bandwidth adjustment seems to be more effective than the approach by Sherlock et al. (1994).

An approach similar to that of Chikkerur et al. (2007) was introduced by Jirachaweng and Areekul (2007), but their block-wise contextual information computation and filtering are performed in the DCT (Discrete Cosine Transform) domain instead of in the Fourier domain. Finally, in the method introduced by Bartunek et al. (2013), contextual filtering is performed in the spatial domain by using cosine functions aligned to the frequency and orientation of the dominant peaks extracted in the frequency domain.

Robust Filter Selection

The need for an effective enhancement is particularly important in poor-quality fingerprints where only the recoverable regions carry information necessary for matching. On the other hand, computing local information (context) with sufficient reliability in poor-quality fingerprint images is very challenging (see Fig. 3.45). To overcome this problem, Kamei and Mizoguchi (1995), Hong et al. (1996), Bernard et al. (2002), and Nakamura et al. (2004) proposed to apply all the filters of a given set at each point in the image (as in Fig. 3.44). A "selector" then chooses the best response from all the filter responses:

- In the method by Kamei and Mizoguchi (1995), the selection is performed by minimizing an energy function that includes terms that require orientation and frequency to be locally smooth.
- Hong et al. (1996) and Nakamura et al. (2004) base their selection on the analysis of local ridges extracted from the filtered images. In particular, Nakamura et al. (2004) enforce orientations that are consistent with a ridge parallelism model.

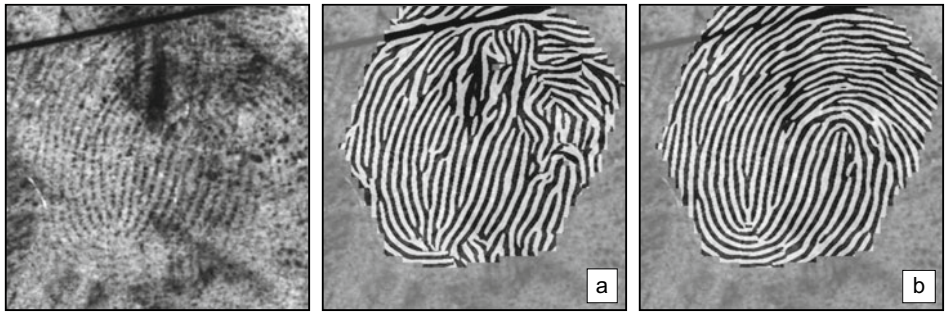

Fig. 3.45 Enhancement of a noisy latent fingerprint with Gabor filtering: **a** the contextual information (orientation and frequency) are automatically computed; **b** the contextual information are provided through manual markup. The much higher quality of the enhancement in (**b**) denotes the importance of using reliable contextual data. For details, refer to Cappelli et al. (2009)

- Bernard et al. (2002) make the selection according to the maximum response. However, unlike most of the Gabor-based methods, phase information coming from the real and the imaginary part of Gabor filters is also used for the final image enhancement.

As expected, approaches that require convolution of an image with a large number of filters are computationally expensive, and it is difficult to obtain efficient implementations. This problem can be partially alleviated by exploiting steerable filter (Freeman & Adelson, 1991) which filters the image with a reduced number of basis filters and derives the remaining filter responses by a linear combination.

Matched filtering is another interesting technique that is able to perform a sort of contextual filtering without explicitly computing local ridge orientation and frequency. In the context of fingerprint enhancement, it was first proposed by Watson et al. (1994) and Willis and Myers (2001). Each 32×32 image block is enhanced separately; the Fourier transform of the block is multiplied by its power spectrum raised to a power k:

$$\mathbf{I}_{enh}[x, y] = F^{-1}\Big\{F(\mathbf{I}[x, y]) \times |F(\mathbf{I}[x, y])|^k\Big\}. \tag{3.12}$$

The power spectrum contains information about the underlying dominant ridge orientation and frequency and the multiplication has the effect of enhancing the block accordingly. Watson et al. (1994) set $k = 0.6$ whereas Willis and Myers (2001) proposed a more aggressive value of $k = 1.4$. Unfortunately, to avoid discontinuities at the edges between adjacent blocks, a large amount of overlap between the neighboring blocks (e.g., 24 pixels) is necessary and this significantly increases the enhancement time.

3.6.3 Multi-Resolution and Iterative Enhancement

Multi-resolution analysis has been proposed to remove noise from fingerprint images. Decomposing the image into different frequency bands (or sub-images) allows us to compensate for different noise components at different scales: in particular, at higher noise levels (low and intermediate frequency bands), the rough ridge–valley flow is cleaned and gaps are closed, whereas at the lower levels (higher frequencies), the finer details are preserved. The enhanced image bands are then recombined to obtain the final image.

- The Almansa and Lindeberg (2000) technique performs shape-adapted smoothing based on second-moment descriptors and automatic scale selection (over a number of scales) based on normalized derivatives. The smoothing operation is adapted according to the local ridge structures, allowing interrupted ridges to be joined. The scale selection procedure estimates local ridge width and adapts the amount of smoothing to the local noise.
- In Hsieh et al. (2003), the multi-resolution representation is based on wavelet decomposition (Mallat, 1989). Each sub-image is processed through both textural and directional filtering to suppress spectral noise and to close the gaps produced by creases and scars.
- The Cheng and Tian (2004) method is based on dyadic space scale and the decomposition depth is determined according to the average ridge width in the image; the noise reduction relies on smoothing the differential images between successive scales (i.e., the details that are lost passing from one scale to the successive one).
- Fronthaler (2007, 2008a) use a Laplacian-like image-scale pyramid to decompose the original fingerprint into three smaller images corresponding to different frequency bands. Each image is then processed through contextual filtering.

The idea behind iterative enhancement is to first filter the fingerprint regions where contextual information is reliable enough to avoid introducing artifacts. Such early filtering determines an enlargement of these reliable regions thus enabling to properly process their neighbors. The entire process can be seen as a wildfire expansion which terminates once the filtered regions collapse and cover the entire image.

- Cappelli et al. (2012) proposed contextual iterative filtering that selectively applies a Gabor filter-bank (six orientations and three frequencies), starting from high-quality regions and then iteratively expanding to low-quality regions (see Fig. 3.46). This approach does not require any prior contextual information: all responses from the Gabor filter-bank are calculated and the enhanced pixels with strong response and high ridge flow homogeneity are selected for the enhanced fingerprint. The algorithm continues iterating until the maximum number of iterations is reached or no more low-quality regions exist in the image.

Fig. 3.46 Iterative enhancement of a fingerprint image as proposed by Cappelli et al. (2012). © IEEE. Reprinted, with permission, from Cappelli et al. (2012)

- The algorithm proposed by Sutthiwichaiporn and Areekul (2013) applies a Gaussian-matched filter (see Eq. 3.12) starting from high-quality regions and then iteratively propagating good spectra of enhanced ridges to lower quality regions. Spectra diffusion is also a fundamental component of the methods by Bian et al. (2018) where the Gabor filtering and Linear Contrast Stretching are performed in advance to improve robustness.
- Orientation diffusion, formerly adopted by some researchers to smooth local orientations (see Sect. 3.3.4), was applied to fingerprint enhancement by Zhao et al. (2009) and Gottschlich and Schönlieb (2012).
- Yang et al. (2013) proposed a two-stage schema: contextual filtering in the spatial domain is performed in the first stage and bandpass filtering in the frequency domain takes place in the second stage; information available at the end of the first stage are exploited to better tune the second stage filters (e.g., improvement of the contextual information).

3.6.4 Learning-Based Enhancement

One of the first fingerprint enhancement methods with a learning stage was proposed by Rama and Namboodiri (2011): this technique makes use of previously learned prior patterns from a set of clean fingerprints to restore a noisy one. A hierarchical interconnected Markov Random Field is used to process the information at multiple resolutions.

A CNN-based approach was proposed by Li et al. (2018) and successfully applied to latent fingerprints. The model consists of an encoder–decoder architecture composed of one common convolution part (encoder) shared by two different deconvolution branches (decoders), which are the enhancement and the orientation branch (see Fig. 3.47). While the enhancement branch is aimed at removing structured noise and providing an enhanced fingerprint image as output, the orientation branch is exploited to guide the enhancement

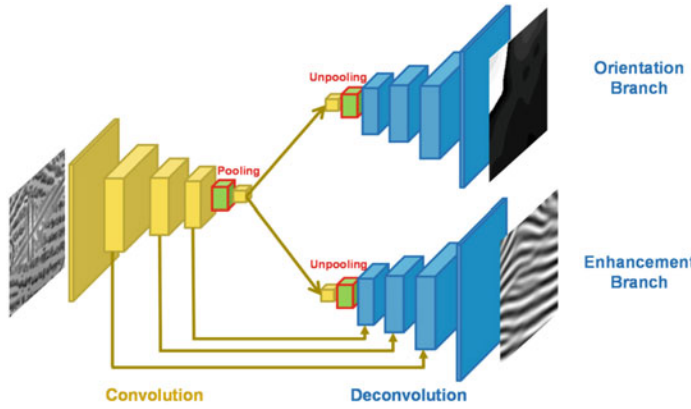

Fig. 3.47 The architecture proposed by Li et al. (2018). It is worth noting that the receptive field of a neuron after the last convolutional layer of the encoder (i.e., the fourth layer) is 53×53 pixels thus providing a large amount of contextual information. © Elsevier. Reprinted, with permission, from Li et al. (2018)

through a multi-task learning strategy. The network is trained end-to-end with a loss function working at the pixel level. The training examples are good-quality fingerprint patches corrupted by artificial structured noise; ground truth data for supervised training can be reliably obtained given the good quality of patches before alteration.

A CNN model similar to FingerNet was introduced by Wong and Lai (2020). Here too, a second parallel branch is used to reconstruct the local orientations and guide the enhancement process, but the reconstructed orientations are then fused with the pre-enhanced image for further improvement, resulting in a deeper architecture. Furthermore, training data is obtained by synthetic fingerprint generation, using the SFinGe method discussed in Chap. 7, which can output both the native good-quality fingerprints and the corresponding noise corrupted versions.

Latent fingerprint enhancement was modeled as an image-to-image translation problem (Isola et al., 2017) and solved with a conditional GAN approach by Joshi et al. (2019). The GAN model consists of two networks: an enhancer network and a discriminator network; the former has an encoder–decoder architecture and is trained to produce an enhanced version of the given noisy fingerprint image and the latter is trained to classify whether the input is a real enhanced image or a generated one. Since image-to-image translation requires paired training data (i.e., for each noisy latent fingerprint, it requires the corresponding high-quality ground truth), synthetic generation by SFinGe is used to prepare the training data.

Deep models based on autoencoders were also proposed for fingerprint enhancement by Schuch et al. (2016) and Svoboda et al. (2017).

3.6.5 Crease Detection and Removal

Some fingerprints (the incidence is higher in elderly people) are affected by the presence of a large number of creases (see Fig. 3.7c). The presence of creases adversely influences the computation of orientation image and can lead to the detection of false ridge-ending minutiae. Provided that contextual information (i.e., local orientation and frequency) has been correctly estimated, contextual filtering techniques are usually capable of filling the small ridge-gap produced by creases; however, if the image is very noisy or the creases are too wide, contextual filtering can fail. To explicitly detect (and optionally remove) creases, some ad hoc techniques have been proposed:

- Vernon (1993) argued that creases are characterized by collinear terminations on ridges and proposed a detection approach based on the analysis of the Hough transform space derived from the ridge-ending minutiae. The Hough transform (Ballard, 1981) is in fact a simple but powerful technique aimed to detect lines in noisy data.
- Wu et al. (2003) and Zhou et al. (2004) modeled a crease by using a parameterized rectangle, followed by a multi-channel filtering framework to detect creases at different orientations. Principal Component Analysis is applied to estimate the crease orientation, length, and width. Figure 3.48 shows some examples of crease detection.
- Oliveira and Leite (2008) identify crease points by looking at the discordance between local orientations computed at two different scales. In fact, when computed at fine scale (i.e., on a small neighborhood), the local orientation within a crease markedly deviates from the overall underlying ridge-line orientation that can be estimated at the coarse scale (i.e., on a large neighborhood). An approach based on Watershed transform is then proposed to remove the creases.
- Zhou et al. (2009b) designed an optimal crease detection filter shaped as a second-order Gaussian derivative. Besides using the detected creases to filter spurious minutiae in

Fig. 3.48 Two examples of crease detection (in black) by Wu et al. (2003) approach. © IEEE. Reprinted, with permission, from Wu et al. (2003)

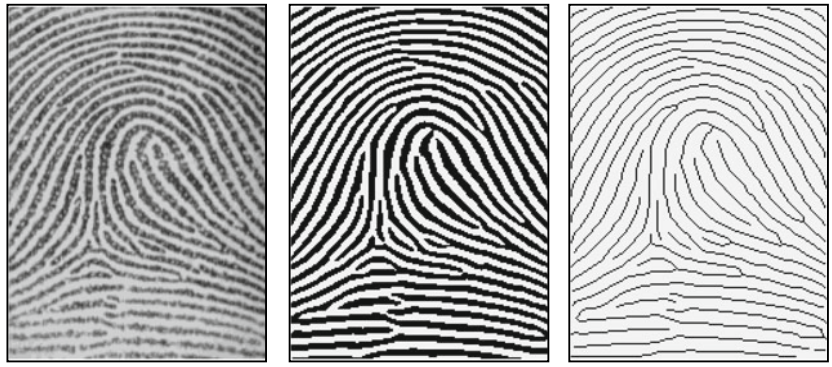

Fig. 3.49 **a** A fingerprint gray-scale image, **b** the image obtained after binarization of the image in (**a**), and **c** skeleton image obtained after a thinning of the image in (**b**). © IEEE. Reprinted, with permission, from Maio and Maltoni (1997)

their proximity, the authors showed that using the creases as features in combination with minutiae can improve fingerprint recognition in elderly people. A similar Gaussian derivative filter was used by Khan et al. (2016), but in this approach, the detected creases are then filled by an inpainting process guided by local orientations.

- Gottschlich et al. (2009) proposed a local-orientated detection approach that traces ridges and valleys on a binarized image. Groups of traced parallel lines are considered to determine local orientations. Traces can be also detected on creases but the number of parallel lines here is small compared to the genuine ridge and valley pattern, and therefore, creases can be quite easily discriminated.

3.7 Minutiae Detection

Most automated systems for fingerprint comparison are based on minutiae matching (see Chap. 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 Sect. 3.6); on the other hand, some enhancement algorithms directly produce a binary output, and therefore, the distinction between enhancement and binarization sometimes vanishes. The binary images are usually submitted to a thinning stage which reduces the ridge-line thickness to one pixel, resulting in a skeleton image (Fig. 3.49). A simple image scan then allows the detection of pixels that correspond to minutiae.

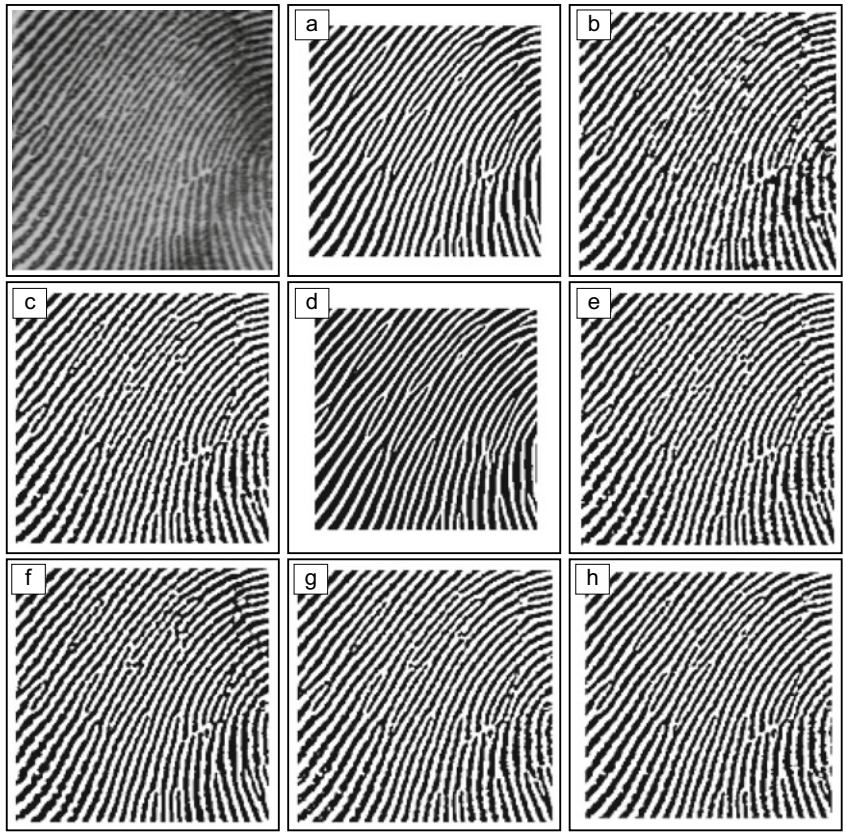

Fig. 3.50 A portion of a good-quality fingerprint image and its binarization through some of the early ad hoc binarization methods **a** O'Gorman and Nickerson (1989); **b** Verma et al. (1987); **c** local threshold approach; **d** Sherlock et al. (1994); **e** Xiao and Raafat (1991); **f** Moayer and Fu (1986); **g** Stock and Swonger (1969); and **h** Watson et al. (1994)

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

- A significant amount of information may be lost during the binarization process and thinning may introduce a large number of spurious minutiae.
- Binarization and thinning can be time-consuming. This is not a serious concern with today's fast processors but things were different some decades ago.
- 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.

Recently, learning-based techniques have been proposed where minutiae detection is approached holistically, that is all the prior steps required by a classical pipeline (computation of local orientation and frequency, enhancement, binarization, thinning, etc.) are performed by a single model trained end-to-end.

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 & 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.

In general, different portions of an image may be characterized by a 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. Therefore, a number of ad hoc binarization approaches have been introduced in the period 1969–1999. Since most of the modern fingerprint enhancement techniques (discussed in Sect. 3.6.2) produce a clear binary image for appropriately chosen parameters,[4] the former ad hoc binarization techniques are today of scarce practical relevance. Figure 3.50 shows a qualitative comparison of binarization results obtained by some of the early methods.

Minutiae detection from binary images is usually performed after an intermediate thinning step that reduces the width of the ridges to one pixel. Thinning algorithm is 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 Fig. 3.51), 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 et al. (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 a set of oriented templates. Luo and Tian (2000) implement a two-step method, where the skeleton extracted at the end of the first step is

[4] In any case, even when the output of the enhancement is a gray-scale image, a simple local thresholding technique often results in satisfactory binarization.

Fig. 3.51 The result of eliminating small holes from both the ridges and valleys of the binary image; the 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

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. Finally, in the method introduced by Bhowmick and Bhattacharya (2009), the binary skeleton is regularized through cubic B-splines fitting.

Mathematical morphology (Gonzales & 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 (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 et al. (1995) implement a morphological "open" operator whose structuring element is a small box oriented according to the local ridge orientation.
- Liang and Asano (2006) 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 & Woods, 2007) and the integral image (Viola & Jones, 2001).

As far as thinning techniques are concerned (Lam et al., 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 et al. (1995) adopted a technique included in the HIPS library (Landy et al., 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 et al. (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. Further fingerprint-specific thinning algorithms have been proposed by Xiang et al. (2009); Wang et al. (2011); and Saleh et al. (2009).

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 1-2011 (2015) coordinate models shown in Fig. 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 & Baja, 1984) as half the sum of the differences between pairs of adjacent pixels in the eight-neighborhood of $\mathbf{p}$:

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

where $\mathbf{p}_0$, $\mathbf{p}_1$, $\ldots\mathbf{p}_7$ 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 (Fig. 3.52) that a pixel $\mathbf{p}$ with $val(\mathbf{p}) = 1$ has the following properties:

- 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. (1991) method extracts the minutiae from

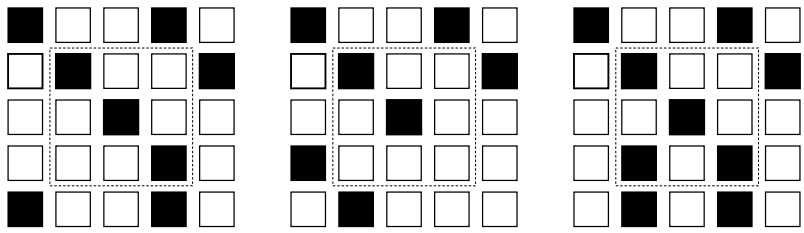

Fig. 3.52 **a** Intra-ridge pixel; **b** ridge-ending minutia; **c** bifurcation minutia

thinned binary images using a three-layer perceptron neural network. The algorithm by Gamassi et al. (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 et al. (2003); and Shi and Govindaraju (2006a) work on thick binary ridges and exploit special ridge tracking algorithms. Shin et al. (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 et al. (2007); Zhang et al. (2011b); and Ma and Zhu (2013) encode the skeleton by means of principal curves which are self-consistent smooth curves 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 angle or direction θ (see Fig. 3.5), in addition to the minutiae coordinates, is used by most of the matching algorithms, to enforce minutiae pairing or correspondence between two fingerprint images. 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).

3.7.2 Direct Gray-Scale Extraction

With the aim of overcoming some of the problems related to fingerprint image binarization and thinning (e.g., the presence of spurious minutiae in the case of irregular ridge edges), some authors have proposed direct gray-scale extraction methods.

Detection Based on Ridge Tracing

Maio and Maltoni (1997) proposed a direct gray-scale minutiae extraction technique, whose basic idea is to track the ridge lines in the gray-scale image, by "sailing" according to the local orientation of the ridge pattern. From a mathematical point of view, a ridge line is defined as a set of points that are local maxima along one direction. The ridge-line extraction algorithm attempts to locate, at each step, a local maximum relative to a section orthogonal to the ridge direction. By connecting the consecutive maxima, a polygonal approximation of the ridge line can be obtained. Given a starting point $[x_c, y_c]$ and a starting direction θ_c, the *ridge line following* algorithm (see Fig. 3.53) computes a new point $[x_t, y_t]$ at each step by moving μ pixels from the current point $[x_c, y_c]$ along the direction θ_c. Then it computes the *section set* Ω as the set of points belonging to the section segment lying on the xy-plane with a median point $[x_t, y_t]$, direction orthogonal to θ_c, and length $2\sigma + 1$. A new point $[x_n, y_n]$, belonging to the ridge line, is chosen among the local maxima of an enhanced version of the set Ω. The point $[x_n, y_n]$ becomes the current point $[x_c, y_c]$ and a new direction θ_c is computed (Fig. 3.53). The optimal

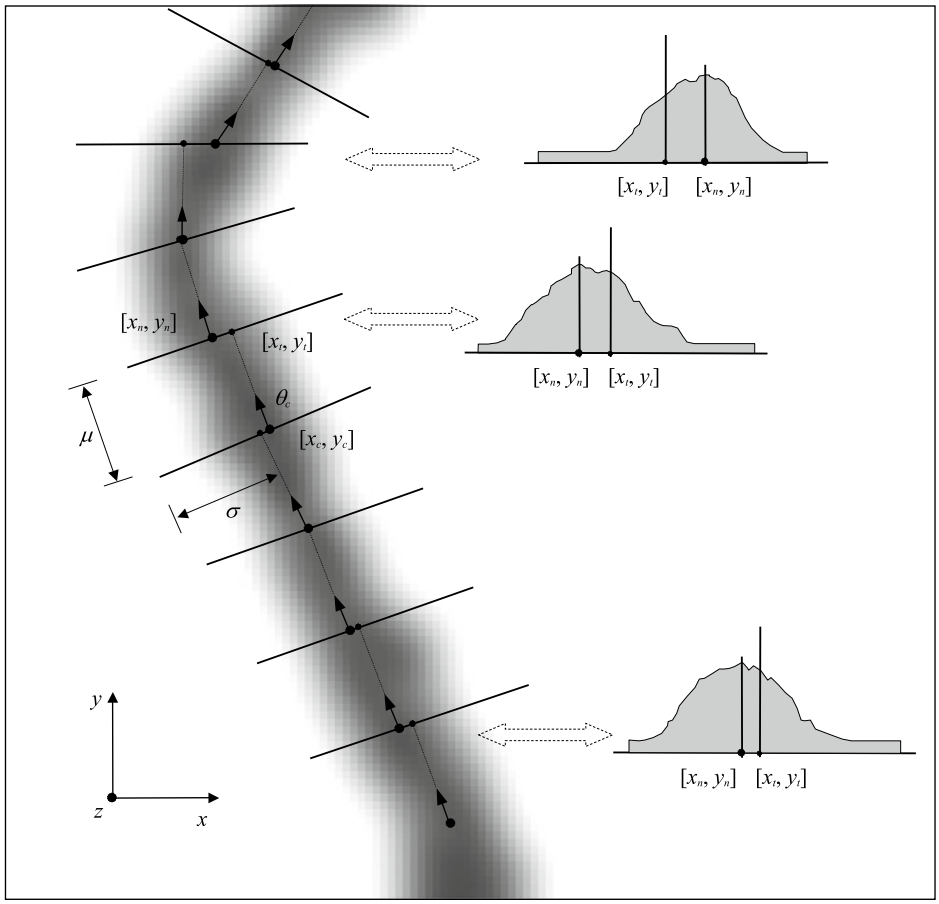

Fig. 3.53 Some ridge line following steps (Maio & Maltoni, 1997). On the right, some sections of the ridge line are shown. © IEEE. Reprinted, with permission, from Maio and Maltoni (1997)

value of the parameters μ and σ can be determined according to the average thickness of the ridge lines. Figure 3.54 shows the results obtained by applying the minutiae detection algorithm to a sample fingerprint.

Jiang et al. (1999, 2001) proposed a variant of the Maio and Maltoni (1997) method, where the ridge following step μ is dynamically adapted to the change of ridge contrast and bending level. Referring to Fig. 3.53, a large step value is selected if there is little variation in the gray-level intensity along the segment $[x_c, y_c]$ $[x_t, y_t]$, and the ridge bending is low. On the other hand, a high bending level of the ridge (possibly facing a ridge bifurcation) or large intensity variations (possibly facing a ridge ending) will result in a small step value. Using a dynamic step speeds up the tracing while maintaining good precision.

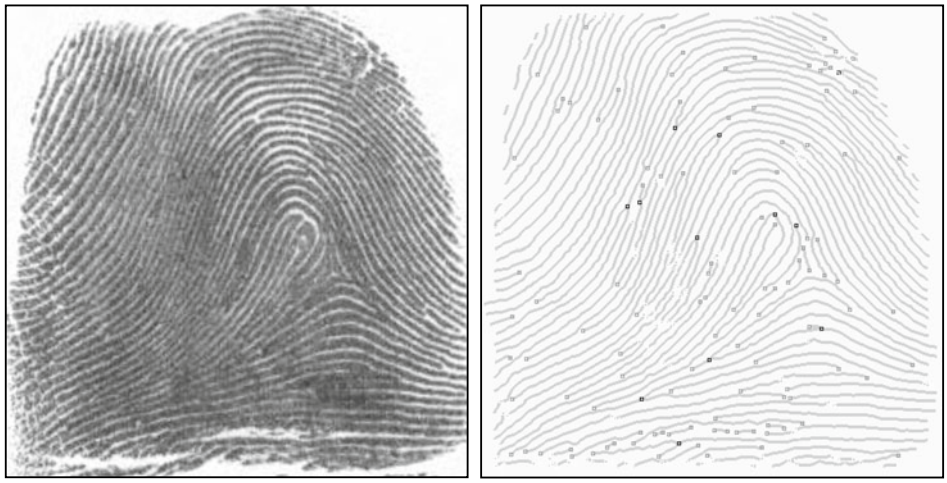

Fig. 3.54 Minutiae detection on a sample fingerprint by using the Maio and Maltoni (1997) method. Ridge-ending minutiae are denoted by gray boxes and bifurcation minutiae are denoted by black boxes

Liu et al. (2000) introduced another modification of the Maio and Maltoni (1997) method. Instead of tracking a single ridge, the algorithm simultaneously tracks a central ridge and the two surrounding valleys. For this purpose, they search a central maximum and two adjacent minima in each section Ω. Minutiae are detected when the shape of the section deviates from a configuration with a central maximum and two side minima. Here too, the ridge following step μ is dynamically adjusted according to the distances between lateral minima from the central maximum. This approach does not need an a priori setting of some parameters such as the maximum bending angle (which determines a stopping criterion in the original algorithm) and the step value μ.

The Chang and Fan (2001) approach is aimed at discriminating the true ridge maxima in the sections Ω obtained during ridge line following. Two thresholds are initially determined based on the gray-level histogram decomposition. The histogram is modeled as a sum of three Gaussian contributions associated with the background, valleys, and ridges, respectively. The mean, variance, and probability of each Gaussian are estimated and two thresholds are derived for successive characterization of maxima and minima of the section Ω. A set of rules is employed to discriminate real ridge points from background noise and intra-ridge variations.

Finally, an efficient integer-arithmetic version of Maio and Maltoni (1997) algorithm has been developed by Canyellas et al. (2005) whose main aim was to port it to low-cost hardware.

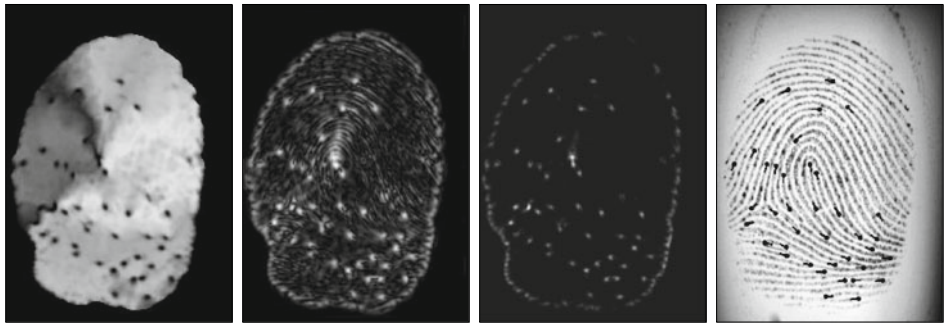

Fig. 3.55 Application of the minutiae detection method proposed by Fronthaler et al. (2008a) to the fingerprint image (**d**). **a** Linear symmetry (LS), **b** parabolic symmetry (PS), **c** PS $(1 - |\mathrm{LS}|)$, and **d** minutiae detected as local maxima of (**c**) superimposed to the original fingerprint image. Images courtesy of J. Bigun

Detection Based on Discontinuities of the Ridge–Valley Flow

Nilsson and Bigun (2001) proposed using Linear Symmetry (LS) properties computed by spatial filtering (Bigun & Granlund, 1987) via separable Gaussian filters and Gaussian derivative filters. Minutiae are identified in the gray-scale image as points characterized by the lack of symmetry. In fact, whereas a non-minutia point is characterized by a direction (i.e., the minutiae angle) along which an infinitesimal translation leaves the pattern least variant, minutiae are local discontinuities in the LS vector field. Fronthaler et al. (2008a) used both Linear Symmetry (LS) and Parabolic Symmetry (PS) to detect minutiae. Near a minutia point, the response to a parabolic symmetry filter is high while the response to a linear symmetry filter is low; hence, the expression PS $(1 - |\mathrm{LS}|)$ allows to detect minutiae more reliably than PS or LS alone. Figure 3.55 shows an example.

Liu and Cao (2016) demonstrated that minutiae can be extracted from Level-1 features, in particular from local orientations and frequencies computed with sufficient resolution. Their approach is based on the FM (Frequency Modulation) model introduced by Larkin and Fletcher (2007) which is discussed in more detail in Chap. 7. In the FM model, a fingerprint image can be represented as a function of a composite phase which is the sum of two components: a continuous phase which is smooth and changes slightly and a spiral phase containing singularities in correspondence of minutiae points. The computation of curve integrals along small circles[5] allows to detect singularities in the composite phase and therefore to extract minutiae points. While this method is very interesting from a theoretical point of view, its accuracy and efficiency are lower than techniques extracting minutiae from ridges.

[5] For analogy, see the Poincaré method in Sect. 3.5.1.

3.7.3 Learning-Based Approaches

Early minutiae detection approaches based on neural networks were proposed by Leung et al. (1990) and Wahab et al. (2004). In Leung et al. (1990), a multilayer perceptron analyzes the output of a rank of the Gabor filters applied to the gray-scale image. The image is first transformed into the frequency domain where the filtering takes place; the resultant magnitude and phase signals constitute the input to a neural network composed of six sub-networks, each of which is responsible for detecting minutiae at a specific orientation; a final classifier is employed to combine the intermediate responses.

One of the first techniques in the deep learning era was introduced by Sankaran et al. (2014) for latent fingerprints. Latent fingerprint minutiae extraction is posed as a binary classification problem to classify (64×64 pixels) patches as minutia or non-minutia. Minutia and non-minutia descriptors are learnt from a large number of tenprint fingerprint patches using stacked denoising sparse autoencoders. Jiang et al. (2016) also proposed a minutiae detection method working at the patch level. This method is based on two stages: in the former, a CNN-based classifier selects patches likely containing one or more minutia in their central part; in the latter, a second CNN locates in the pre-selected patches the internal subwindows containing minutiae.

A further patch-based approach was introduced by Darlow and Rosman (2017). Here a single CNN is used for classification and some post-processing steps are implemented to improve the location accuracy. The most relevant contribution of this work is the introduction of an automatic procedure to establish minutiae ground truth, based on a "smooth" fusion of the responses of five commercial minutiae extractors.

The main limitations of patch-based approaches are (i) the low efficiency due to the need of a sliding window scan over a large number of partially overlapped windows and (ii) the limited contextual information (i.e., the size of the patch). More complex models have been more recently proposed to overcome the above drawbacks.

- Tang et al. (2017a) approached minutia extraction as an object-detection problem (similarly to a Faster R-CNN model, Ren et al., 2015), where a Fully Convolutional Network is used to map raw fingerprints to a minutia-score map with 1 position every 16×16 pixels; the positions whose score exceeds a given threshold are candidate minutiae; the neighboring regions are further processed by a second CNN (sharing the same convolutional levels) that also regresses the minutia orientation (see Fig. 3.56).
- Tang et al. (2017b) proposed a unified deep model, denoted as FingerNet, where domain knowledge and end-to-end training are both exploited to improve accuracy. In particular, the model is initialized by converting traditional feature extraction steps (normalization, orientation extraction, enhancement, and minutiae detection) into convolutional layers with fixed weights. Then the above basic layers are expanded with further layers and all the model weights are further tuned during end-to-end training of the model. A composite loss function is used to exploit weak, strong, and ground

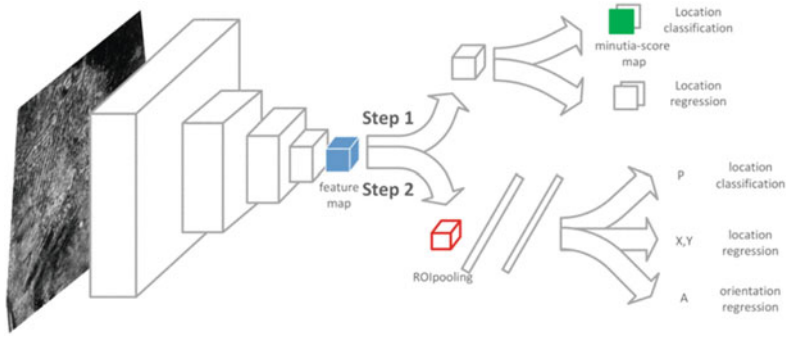

Fig. 3.56 The architecture proposed by Tang et al. (2017b). The proposals generated by the detection branch (Step 1) are classified by a second branch which also calculates their orientations (Step 2). © IEEE. Reprinted, with permission, from Tang et al. (2017b)

truth labels on orientation, foreground area, and minutiae. The approach by Tang et al. (2017b) is described in more detail and with illustrations in Sect. 6.4.6.

- Another two-step method was proposed by Nguyen et al. (2018b). In the first step, an improved FingerNet (Tang et al. 2017b) is used to produce candidates that are further checked at patch level by a second CNN. While the entire approach is quite complex, its minutiae detection accuracy exceeded previous state-of-the-art methods on FVC2004 and NIST SD27.
- Nguyen et al. (2020) introduced a simple semantic segmentation approach where a light version of the U-Net model (Ronneberger et al. 2015), is used to classify each pixel into 37 classes: 36 classes corresponding to a minutia with a given angle (i.e., with 10 degrees steps) and the last class for non-minutia. Good minutiae detection results are reported for many FVC databases with an inference time of only 130 ms for 640×480 images on GPU.

3.7.4 Minutiae Encoding Standards

Standards have been introduced to define the way minutiae information should be encoded. Most of these standards, including ISO/IEC 19794-2 (2011), ANSI/NIST-ITL 1-2011 (2015), ANSI/INCITS 378 (2004), and CDEFFS (2008), require to encode minutiae data with the same conventions in order to promote interoperability. For example (see Fig. 3.57), ISO/IEC 19794-2 (2011) requires to extract both the ridge and the valley skeletons and to place (i) bifurcation minutiae in correspondence with the ridge skeleton bifurcations (i.e., pixels $\mathbf{p}$ with $cn(\mathbf{p}) = 3$), and (ii) ridge-ending minutiae in correspondence with the valley skeleton bifurcations (this exploits the ridge-ending/bifurcation

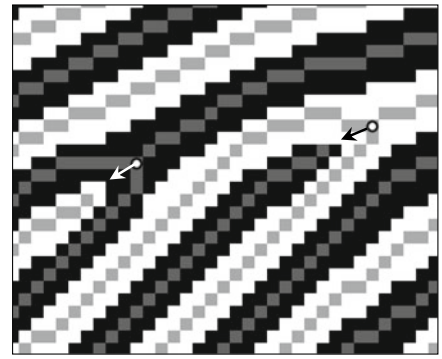

Fig. 3.57 Ridge-ending (black arrow) placement on valley skeleton and bifurcation (white arrow) placement on ridge skeleton according to ISO/IEC 19794-2 (2011). In the figure, ridges are black, valleys are white, and ridge and valley skeletons are gray

duality shown in Fig. 3.6); the minutia direction θ is computed in both the cases (ridge ending and bifurcation) as the bisector of the two angularly closest branches departing from **p**.

The main aim of the standards is to achieve interoperability among minutiae templates extracted by different approaches and so, for this purpose, clear and unambiguous rules should be given; unfortunately, none of the current standards is sufficiently clear and exhaustive and new efforts are being made to improve them. The interoperability evaluation conducted in the MINEX (Grother et al., 2006; Wu & Garris, 2007) and MTIT (Bazin & Mansfield, 2007) projects has clearly pointed out such limitations. The main problems are as follows:

- Minutiae position is often defined resorting to the skeleton of a binary fingerprint image, but the skeleton itself depends on enhancement, binarization, and thinning algorithms.
- While criteria are given to validate minutiae and filter out spurious minutiae (e.g., if one of the branches departing from a minutia cannot be tracked for more than a given length, then the minutia is invalid), an iterative unordered application of these rules can also lead to unintended deletion of valid minutiae.
- No specific rules are given to define how to deal with very noisy regions and with high curvature regions (i.e., singularities) where minutiae tend to cluster.

Bolle et al. (2002) addressed the problem of precisely and consistently locating the minutiae points in the gray-scale fingerprint pattern. In fact, different feature extraction algorithms tend to locate the minutiae at slightly different positions (depending on their operational definition of minutia and the intermediate processing steps) and this may lead to interoperability problems. Bolle et al. (2002) provided a formal definition of minutia based on the gray-scale image that allows the location and orientation of an existing minutia to be more precisely determined. However, this approach has been ignored by current standards that define the minutiae location on the basis of binary skeletons.

3.7.5 Benchmarking Minutiae Extraction

Since minutiae extraction is a fundamental fingerprint processing step, benchmarking it in isolation (instead of when it is embedded in a whole fingerprint recognition system) can be very useful to assess the performance of existing techniques and guide future developments. One of the main obstacles is the need for accurate ground truth information. In fact, manual labeling not only is boring and prone to errors, but it is very critical in low-quality fingerprints that, on the other hand, constitute the most interesting cases. Synthetic fingerprint generation is a valuable strategy to obtain reliable ground truth labels also for low-quality data (see Chap. 7).

Some studies have been published where minutiae extraction techniques (both commercial solutions and academic developments) have been tested on available datasets:

- Kayaoglu et al. (2013) used a GUI tool to manually mark a total of 116,000 minutiae over four FVC datasets (acquired with optical, capacitive, and thermal sensors). Commercial minutiae extractors were compared with manually extracted minutiae in terms of associated fingerprint recognition accuracy to conclude that manual extraction leads to significantly better performance especially on low-quality fingerprints.
- Chugh et al. (2017) used the four mentioned labeled FVC datasets in conjunction with NIST SD27 consisting of 257 rolled fingerprints whose minutiae were marked by forensic examiners. The test also includes a corpus of synthetically generated fingerprints. Three COTS algorithms and an open-source solution (*mindtct* by NIST) have been compared in terms of missing and spurious minutiae, also by using the overall Goodness Index metric introduced by Ratha et al. (1995). In the experiments, the fingerprints were divided into five bins according to their NFIQ2.0[6] quality: 0–20, 21–40, 41–60, 61–80, and 81–100; the best algorithm produced a percentage of spurious minutiae ranging from 44% (bin 0–20) to 10% (bin 81–100) and a percentage of missing minutiae ranging from 29% (bin 0–20) to 16% (bin 81–100). The open-source algorithm extracted a significantly higher number of spurious minutiae across all fingerprint quality. Positional and angular errors were also measured for paired minutiae: typical positional errors are 3, …, 5 pixels and angular errors are 5°, …, 8°.
- The impact of different types of minutiae detection errors constitutes the subject of the study by Grosz et al. (2020). Their evaluations revealed that the performance of fingerprint minutiae matchers is more susceptible to non-linear distortion and missing minutiae than spurious minutiae and small positional displacements of the minutiae locations.

[6] NFIQ2.0 is introduced in Sect. 3.11.3.

3.8 Minutiae Filtering

A post-processing stage is often useful in removing the spurious minutiae detected in highly corrupted fingerprint regions or introduced by preceding processing steps (e.g., thinning). However, some of the techniques introduced in Sect. 3.7 (e.g., especially learning-based methods working on large input neighborhoods) do not need a separate post-filtering stage.

Two main post-processing types have been proposed:

- Structural post-processing.
- Minutiae filtering in the gray-scale domain.

3.8.1 Structural Post-Processing

Simple structural rules may be used to detect many of the false minutiae that usually affect thinned binary fingerprint images. Xiao and Raafat (1991) identified the most common false-minutiae structures and introduced an ad hoc approach to remove them (Fig. 3.58). Their algorithm is rule-based and requires some numerical characteristics associated with the minutiae as input: the length of the associated ridge(s), the minutia angle, and the number of facing minutiae in a neighborhood. As shown in Fig. 3.58, the algorithm connects facing endpoints (a) and (b); removes bifurcations facing with endpoints (c) or with other bifurcations (d); and removes spurs (e), bridges (f), triangles (g), and ladder structures (h).

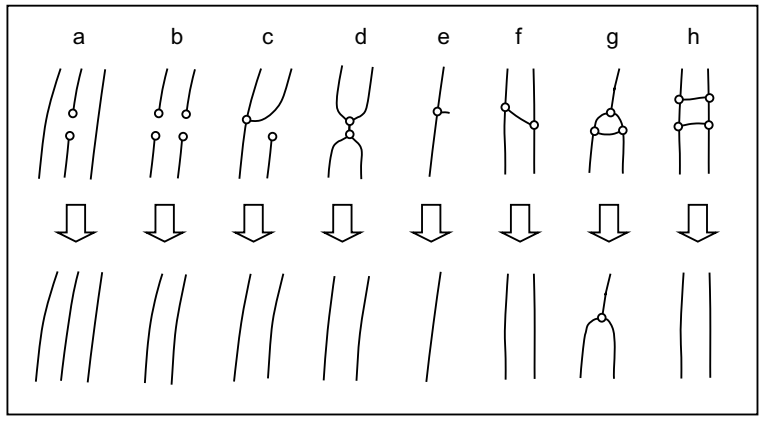

Fig. 3.58 The most common false-minutiae structures (top row) and the structural changes resulting from their removal (bottom row)

Hung (1993) and Zhao and Tang (2007) exploited the minutiae duality (Fig. 3.6) to purify false minutiae extracted from binary thinned images.

- In Hung (1993), both ridge and valley skeletons are extracted and only ridge minutiae having a counterpart (of complementary type) in the valley skeleton are retained. A graph is defined for both ridge and valley skeletons by assigning a vertex to each ridge ending and bifurcation and by assigning an edge to each ridge. Each edge is characterized by the length of the corresponding ridge, and the degree of a vertex is given by the number of converging edges. Spurs (i.e., very short edges) and holes (i.e., loops with a very small diameter) are first removed by considering some property of the ridge graph. Bridges between adjacent ridges are then removed by exploiting their relation with breaks in the dual space.
- Zhao and Tang (2007) argue that for most of the false minutiae there is at least a bridge structure; referring to Fig. 3.58, this is true not only for bridges (f), triangles (g), and ladders (h) in the ridge skeleton but also for breaks (a) in the valley skeleton. An H-point is defined as a bridge structure in one of the two (ridge or valley) skeletons and its corresponding break in the dual skeleton; a simple rule is then defined to detect and remove H-points thereby eliminating many false minutiae. However, to avoid cancelation of genuine structures, the H-point removal must be embedded within an ordered sequence of deletion steps: short breaks, spurs, H-points, close minutiae, and border minutiae.

In the approach by Farina et al. (1999), spurs and bridges are removed based on the observation that in a "spurious" bifurcation, only two branches are generally aligned whereas the third one is almost orthogonal to the other two. Short ridges are removed on the basis of the relationship between the ridge length and the average distance between ridges. Ridge endings and bifurcations are then topologically validated; they are (i) removed if topological requirements are not satisfied; (ii) classified as less reliable if the requirements are not fully satisfied; and (iii) considered as highly reliable minutiae, otherwise. An example is shown in Fig. 3.59.

A slightly different implementation of spurious minutiae removal was proposed by Kim e al. (2001). In their work, local orientation and flow of ridges are key factors for post-processing to avoid eliminating true minutiae. Bhowmick et al. (2002) assign a score to each minutia based on clarity of ridge and valley flow and the noise level in the locality of the minutia; Kim (2005) and Chen et al. (2007) also assign scores to the minutiae based on local ridge connectivity, interspacing, and symmetry. The scores can then be used either for immediate filtering or minutiae weighting during matching.

The filtering method of Bhanu et al. (2000) verifies each minutia, detected from the thinned binary image, through correlation with logical templates (i.e., template matching) adapted to the local ridge orientation. In Bhanu and Tan (2001), an evolution of the above

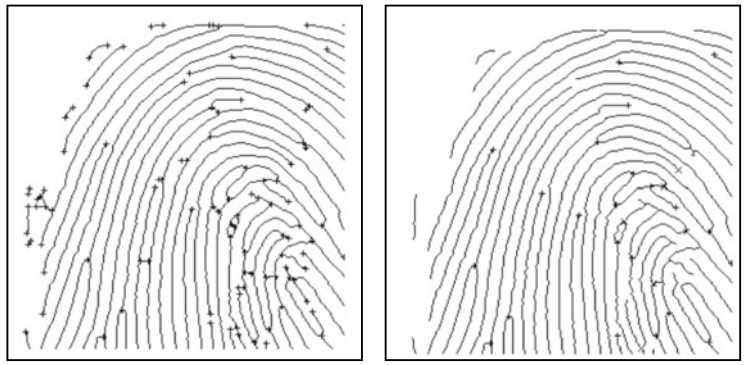

Fig. 3.59 Minutiae post-processing according to Farina et al. (1999). On the right, most of the false minutiae present in the thinned binary image (on the left) have been removed. © Elsevier. Reprinted, with permission, from Farina et al. (1999)

method is proposed where templates are not static, but are learned in a supervised manner from examples.

3.8.2 Minutiae Filtering in the Gray-Scale Domain

A direct gray-scale minutiae filtering technique reexamines the gray-scale image in the spatial neighborhood of a detected minutia with the aim of verifying the presence of a real minutia.

Maio and Maltoni (1998b) used a three-layer neural network where a partial weight sharing allows the ridge-ending/bifurcation duality to be exploited (Fig. 3.60). The minutiae neighborhoods in the original gray-scale image are normalized, with respect to their angle and the local ridge frequency, before passing them to a neural network classifier, which classifies them as ridge ending, bifurcation, and non-minutia. Figure 3.61b shows the same minutiae neighborhoods of Fig. 3.61a after the normalization. To take advantage of the ridge-ending/bifurcation duality, both the original neighborhood and its negative version constitute the input to the neural network classifier. Additionally, to avoid the problems related to training large networks, the dimensionality of the normalized neighborhoods is reduced through the Karhunen–Loeve transform (Jolliffe, 1986). Experimental results showed that this filtering method, in spite of a certain increase in missed minutiae, provides a significant reduction in false minutiae and misclassified minutiae (i.e., a ridge ending detected as bifurcation and vice versa) errors.

The minutiae verifier of Prabhakar et al. (2003) operates on the gray-scale neighborhoods extracted from the original image after enhancement through the Gabor filtering (Hong et al., 1998). Minutiae neighborhoods are normalized with respect to minutiae

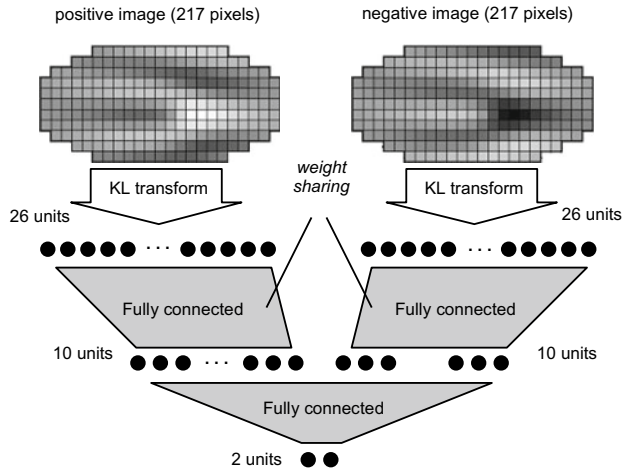

Fig. 3.60 The neural network architecture to classify gray-scale minutiae neighborhoods into ridge ending, bifurcation, and non-minutiae (Maio & Maltoni, 1998b). © IEEE. Reprinted, with permission, from Maio and Maltoni (1998b)

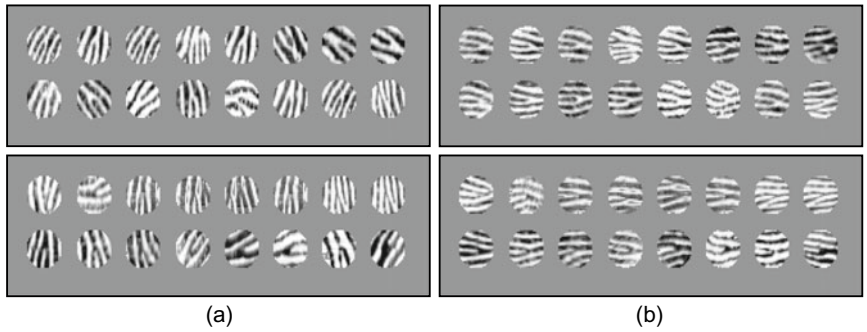

Fig. 3.61 **a** Minutiae neighborhoods (ridge-ending minutiae at the top, bifurcation minutiae at the bottom) as they appear in the original gray-scale images; **b** the same neighborhoods have been normalized with respect to minutiae angle and local ridge frequency (Maio & Maltoni, 1998b). © IEEE. Reprinted, with permission, from Maio and Maltoni (1998b)

angle and local ridge frequency. The resulting patterns are classified through a Learning Vector Quantizer (Kohonen et al., 1992) trained in a supervised fashion to discriminate between minutiae and non-minutiae. The authors obtained a classification accuracy of 87% and a reduction of about 4% fingerprint matching error when their minutiae verification algorithm was embedded into the minutiae-based fingerprint verification system described in Jain et al. (1997).

Chikkerur et al. (2005) proposed two minutiae verifiers:

- The first one is based on the response of the minutiae neighborhood to a bank of steerable wedge filters. The response is fed to a feedforward backpropagation network to classify the inputs as either minutiae or non-minutiae neighborhoods.
- The second (and more accurate) one encodes the minutiae neighborhoods as a linear sum of basis images made up of multi-resolution Gabor elementary functions. A parametric Bayesian classification is then applied. The authors report a minutiae verification accuracy of 98%.

Most of the deep learning techniques introduced in Sect. 3.7.3 can be used for minutiae filtering in the gray-scale domain. In particular,

- the patch-based classifiers (e.g., Sankaran et al. 2014; Jiang et al. 2016; Darlow and Rosman 2017), can be directly used for minutiae post-filtering without any sliding window scan;
- in the two-stage methods (e.g., Tang et al. 2017b; Nguyen et al., 2018b), the second stage can be conceived as a minutiae post-filtering.

3.9 Estimation of Ridge Count

Absolute position, direction, and type of minutiae (e.g., ridge ending or bifurcation) are not the only features that may be used for fingerprint recognition. In fact, forensic experts and latent fingerprint examiners have often used *ridge count* to increase the reliability of their analysis (Henry, 1900). Ridge count is an abstract measurement of the distances between any two points in a fingerprint image (Lin & Dubes, 1983). Let **a** and **b** be two points in a fingerprint; then the ridge count between **a** and **b** is the number of ridges intersected by segment **ab** (Fig. 3.62).

Ridge count has been typically used in forensic matching because of the difficulty of human experts to work in the Euclidean space. However, because the early automated

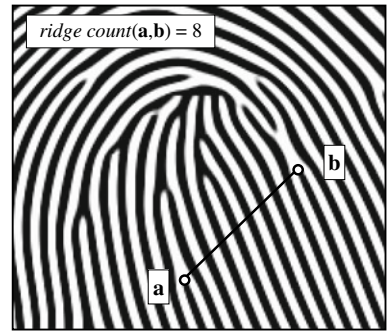

Fig. 3.62 In this example, the number of ridges intersected by segment **ab** (ridge count between **a** and **b**) is 8.

fingerprint identification systems (AFIS) were developed from an intuitive design geared toward duplicating the performance of human experts in matching fingerprints, ridge counts have been used in the AFIS systems. With an increased interest in improving the performance of fingerprint recognition systems in commercial applications, several authors have proposed ridge counts as features.

Although the general definition of ridge count includes measuring the number of ridges between any two points (**a** and **b** in Fig. 3.62) in the fingerprint images, typically, these points coincide with some well-defined points in the fingerprint pattern (e.g., position of the singularities or position of the minutiae). For example, in forensic applications of AFIS, it is common to count the ridges between core and delta.

There exist two main approaches for counting the number of ridges between points **a** and **b** in a fingerprint image:

- Determine the number of (0–1) transitions along the segment **ab** in a binarized image.
- Determine the number of local maxima in the section **ab** of a gray-scale image. Refer to the *x*-signature method (Fig. 3.21) for an example of a possible implementation.

In both cases, the estimation of ridge count may be problematic in noisy areas, near singularities, near minutiae, and when the segment orientation is close to the underlying ridge orientation. Sha et al. (2006) suggest using only a subset of highly reliable ridge counts and propose a labeling-based algorithm for their extraction. Kovacs-Vajna (2000) matched the ridge profiles between pairs of minutiae using the dynamic time warping technique. This method is conceptually similar to ridge counting, even if it cannot be directly used to estimate the ridge count but only to verify that the ridge–valley profiles between two pairs of corresponding points are similar.

3.10 Pore Detection

While pores are highly distinctive and extremely important for latent fingerprint examiners, their practical utility to improve automated fingerprint recognition algorithms is questionable (Zhang & Jain, 2010). More details are provided in Chap. 4. Noticing that a single ridge with a string of connected pores may be incorrectly extracted as two ridges, Hara (2011a) proposed an algorithm to extract and then eliminate pores to avoid such a problem. Given sufficiently high-resolution fingerprint images (e.g., 1,000 dpi), several techniques can be used for automatic pore extraction.

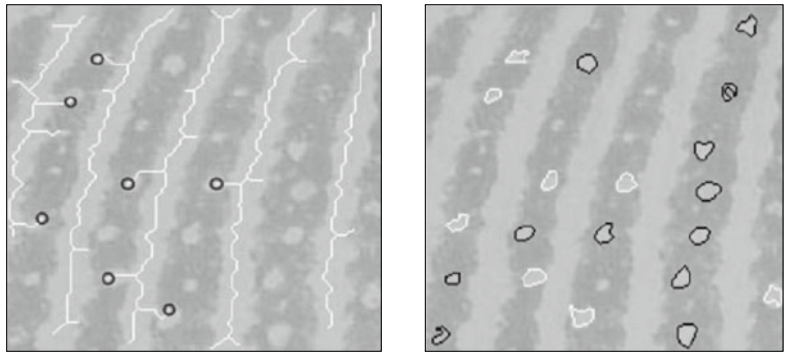

Fig. 3.63 Examples from Stosz and Alyea (1994): **a** detection of open pores from the skeleton; **b** extraction of open pores (white) and closed pores (black). Images courtesy of K. Kryszczuk

3.10.1 Skeletonization

One of the first methods was proposed by Stosz and Alyea (1994). Specifically, the locations of all endpoints (with at most one neighbor) and branch points (with exactly three neighbors) in the skeleton image are extracted and each endpoint is used as a starting location for tracking the skeleton. The tracking algorithm advances one pixel at a time until one of the following stopping criteria is encountered: (1) another endpoint is detected, (2) a branch point is detected, and (3) the path length exceeds a maximum allowed value. Condition (1) implies that the tracked segment is a *closed pore*, while condition (2) implies an *open pore*. Finally, skeleton artifacts resulting from scars and wrinkles are corrected and pores from reconnected skeletons are removed. An example of pore extraction is shown in Fig. 3.63. Another skeletonization-based pore detection algorithm with more anatomical constraints with respect to the Stosz and Alyea (1994) algorithm was proposed by Kryszczuk et al. (2004).

3.10.2 Filtering

Jain et al. (2007) noted that skeletonization is effective for pore extraction only when the image quality is very good and the image resolution is very high. Hence, unlike in previous studies, pores were extracted using the Gabor filters and Mexican hat wavelet transform; see Fig. 3.64 for some details of a pore extraction process. Other filtering-based approaches were introduced by Abhyankar and Schuckers (2010) and Zhao et al. (2010a); the formed is still is based on wavelet transform while the latter relies on DoG (Difference of Gaussians) computed at various scales.

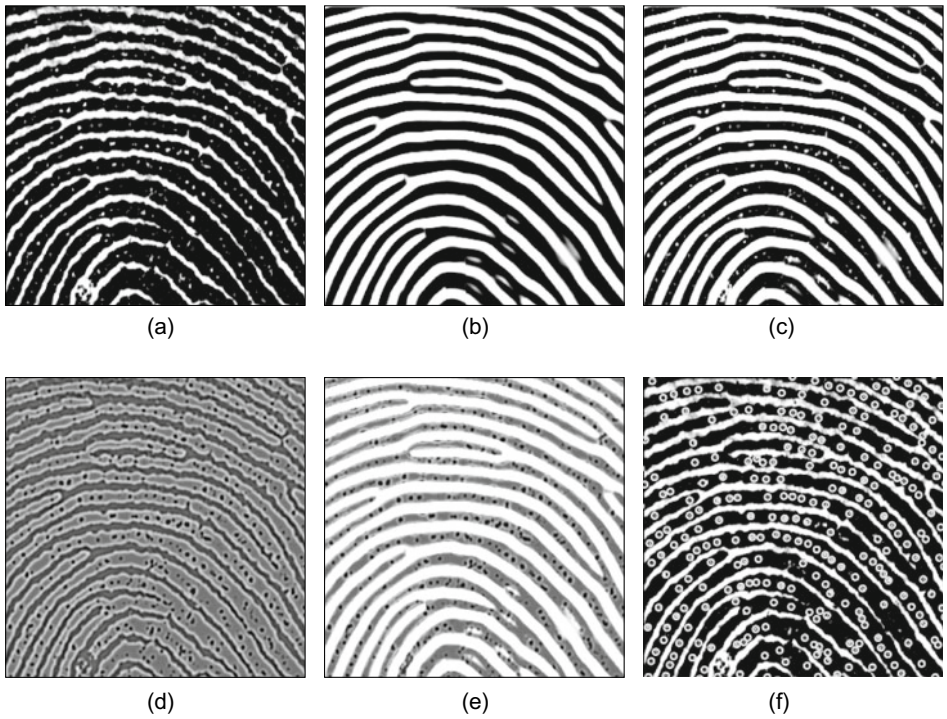

Fig. 3.64 Pore extraction in Jain et al. (2007): **a** a partial fingerprint image at 1,000 dpi; **b** enhancement of the image shown in (**a**) using a Gabor filter-based contextual technique; **c** a linear combination of (**a**) and (**b**); **d** wavelet bandpass filtering of the image in (**a**) that exhibits small dark blob in correspondence of the pores; **e** a linear combination of (**d**) and (**b**), and **f** extracted pores (white circles) after thresholding the image in (**e**). © IEEE. Reprinted, with permission, from Jain et al. (2007)

Zhao et al. (2010b) argued that an anisotropic filtering is preferable because open pores are not circularly shaped and their response to isotropic filtering can be weak. The proposed adaptive anisotropic filtering approach achieved a true detection rate of 84.8% and a false detection rate of 17.6% on a database of 24 fingerprint images acquired at 1,000 dpi.

3.10.3 Topological Approaches

In the method proposed by Malathi et al. (2010), pores detection is performed by employing a topological algorithm known as marker controlled watershed segmentation. The method is claimed to be able to extract pores also on 500 dpi images.

The Teixeira and Leite (2013) method relies on a multi-scale morphological operator used in conjunction with local orientation to extract both closed and open pores. The accuracy reported on the 30 pore-annotated fingerprints included in the PolyU HRF Database is higher than previous filtering-based methods. A post-processing technique was then proposed by Teixeira and Leite (2014) to reduce the number of false pore detection; the filtering is based on the observation that distance between adjacent pores in a ridge is proportional to the ridge width.

3.10.4 Deep Learning Methods

In the method introduced by Donida Labati et al. (2018), the fingerprint image is first processed by a five-layer CNN aimed at enhancing pores while discarding the ridge–valley pattern. This can be seen as a sort of image-to-image translation. Pore centroids are extracted from the obtained pore map by looking at connected components after simple binarization. A second CNN is then used to check each candidate pore by also taking into account the ridge map and some symmetry information. In their experiments, the authors also considered touchless reader-acquired fingerprints and latent fingerprints and, as expected, confirmed that in these cases pore extraction is very challenging. An improved version of this method was introduced by Jang et al. (2017), relying on a deeper CNN model and a more accurate selection of pores from the pore map. More complex and still deeper models (e.g., ResNet and DenseNet) were adopted by other researchers (Anand & Kanhangad, 2019; Shen et al., 2019) to gain minor improvements over the previous methods. In Ding et al. (2021), a modified U-Net is used to extract sweat glands and sweat pores from OCT volumes.

3.11 Estimation of Fingerprint Quality

Many different meanings could be associated with the term quality. However, most of the researchers have focused only on the *operational* definition of quality, that is, the estimate of quality must be predictive of the utility of features and/or "matchability" of the fingerprint images.

In general, fingerprint quality can be estimated at a *global* level (i.e., a single quality value is derived for the whole image) or at a *local* level (i.e., a distinct value is estimated for each block/pixel of the image). Of course, the availability of local estimates is preferable since it is more descriptive, and in any case, one could compute the global quality from the statistics of the local estimates. Estimating fingerprint quality is important since it has the following properties (Grother & Tabassi, 2007):

- Rejects very low-quality samples (images) during enrollment or/and selects the best sample(s).
- Isolates unrecoverable regions (see Sect. 3.6) where fingerprint enhancement is counterproductive as it leads to the detection of several spurious features.
- Adapts the matching strategy to the quality of fingerprints.
- Assigns weights to features (at the matching stage) according to the quality.

The most popular approach to estimate global fingerprint quality is NFIQ (NIST Fingerprint Image Quality). The first version of NFIQ dates back to 2004 (Tabassi et al., 2004) and the second version, known as NFIQ 2.0, was introduced one decade later (Tabassi et al., 2021). NFIQ defines the quality as a prediction of a matcher performance: good-quality fingerprints are likely to produce high match scores. An important advantage of this method is that it does not require a ground truth provided by a human expert; in fact, defining the ground truth by visual inspection is quite complicated, could lead to subjective evaluations, and is not necessarily the best approach when the focus is on automated matching algorithms.

NFIQ and NFIQ 2.0 were conceived for plain and rolled fingerprint images at 500 dpi captured using optical sensors or scanned from inked cards. Some adaptations (at least retraining and calibration) would be necessary for other categories of sensors. Some authors introduced specific quality metrics for latent fingerprints (Sankaran et al., 2013; Yoon et al., 2013) and for high-resolution fingerprints (e.g., 1000 dpi) where pore visibility and reliability are quality-related (Zhao et al., 2010c; Teixeira & Leite, 2017). Quality estimation of latent fingerprints is discussed in more detail in Chap. 6.

Some studies (Young & Elliott, 2007) have shown that, on average, (i) fingerprint images from the index and middle fingers exhibit better quality than images taken from the ring and little fingers, and (ii) Whorl is the fingerprint class containing the largest proportion of good-quality fingerprint images, whereas Arch is at the opposite side of the quality scale. This knowledge, when available, could be useful as a prior probability to quality estimation approaches.

In the following, a brief summary of the existing global and local approaches for estimating fingerprint image quality is reported. More comprehensive (and comparative) reviews can be found in Alonso-Fernandez et al. (2007) and Yao et al. (2018).

3.11.1 Local Quality Estimation

A number of methods have been proposed to estimate block-wise fingerprint quality. Most of them estimate the local quality according to the local orientation reliability (see Sect. 3.3.1): Shen et al. (2001), Lim et al. (2002), Chen et al. (2004a), Yao et al. (2004), Chen et al. (2005), Zhu et al. (2005), and Panetta et al. (2019). The way the reliability of a single orientation is obtained depends on the orientation computation method, but is

usually related to the coherence of a set of orientation estimations in a given neighborhood (see Eq. (3.4) and Fig. 3.14c). Although orientation coherence is a very powerful feature to measure quality, it fails near the singularities; in fact, singularities are characterized by high curvatures which result in low coherence (see Fig. 3.28). The use of linear and parabolic symmetry operators (see Fronthaler et al. 2006, 2008a, b) overcomes this problem. In fact, the orientations in good-quality regions highly correlate with (i) linear symmetry filters outside singularities and (ii) parabolic symmetry filters near the singularities, whereas low-quality regions weakly respond to both types of filters.

Other features can be used to characterize local quality:

- Statistics derived from pixel intensities (e.g., mean, variance, contrast, gradient magnitude, histogram properties, etc.) as proposed by Shi et al. (2004), Uchida (2004), Chen et al. (2004b), Lim et al. (2004), Hwang (2004), Qi et al. (2005a), and Zhu et al. (2005).
- Ridge frequency, ridge thickness, and ridge-to-valley thickness (Lim et al., 2002; Zhu et al., 2005); a substantial deviation of these values from their typical range is alarming and can be exploited to detect unreliable regions.
- The presence of dominant frequencies in the Fourier spectrum of a local region (Lim et al., 2004) and the local resemblance to an ideal sinusoidal wave pattern in the spatial domain (Lee et al., 2006).
- The regularity of the manifold determined by fingerprint patches in a local neighborhood (Tao et al., 2012).

Liu et al. (2020) performed a cluster analysis of a large number of fingerprint blocks to group them into four quality classes (further expanded with background and singular regions) and then trained a simple neural network classifier to compute the block class starting from a feature vector including some of the previously discussed local features.

It is difficult to compare the relative performance of these methods quantitatively because of a non-uniform quality definition. In ANSI/NIST-ITL 1-2011 (2015) standard, the local quality of fingerprint is defined according to the reliability of different level features (six levels). However, there is no public fingerprint database based on this standard for scientific research.

3.11.2 Global Quality Estimation

Qi et al. (2005b) combine local and global features already available in the literature, but among the global features, the authors suggest taking into account the size of the foreground area, the foreground centered with respect to the image center, and the presence of detectable singularities.

A good-quality fingerprint image exhibits a ring around the origin of the frequency coordinate in the Fourier spectrum, because the ridge–valley patterns are quasi-periodic structures and present a dominant frequency in most directions with an almost uniform modulus. The implementation of ad hoc ring detectors allows an estimate of the overall fingerprint image quality: see Uchida (2004), Chen et al. (2005), and Lee et al. (2005).

Some authors (Munir et al., 2012; Sharma & Dey, 2019) proposed hierarchical or fuzzy clustering to group fingerprints in quality classes and in particular to infer their appearance: *normal*, *wet,* or *dry.*

An interesting quality measure was introduced by Yao et al. (2015): this metric only relies on the minutiae points and therefore it can be computed starting from templates (e.g., in the ISO standard format) with no need of the original fingerprint image. To compute the quality, the set of minutiae is modeled with the convex-hull and Delaunay triangulation. The authors observed that bad-quality samples generate tiny and extremely narrow triangles (considered as unreasonable) due to spurious minutiae. Hence, geometrical characterization of minutiae triangles was used to determine the quality. Comparative evaluations in Yao et al. (2018) denote that, in spite of its simplicity, this approach is quite effective when used to select the best sample for enrollment.

The size of the overlapping area of two fingerprints has a large impact on matching performance. To ensure a large overlapping area, the central area of fingerprints (corresponding to the most distinctive region) is preferred to be captured in enrollment and recognition stages. Thus, a fingerprint with a large central area should be assigned a higher quality than a fingerprint, which has a similar image quality and overall size, but a smaller central area. To take this factor into account, Hara (2011b) proposed to estimate fingerprint quality based on completeness of central area (called pattern area) and ridge quality in this area.

To evaluate the effectiveness of global quality estimation approaches, Grother and Tabassi (2007) proposed to measure the improvement in matching accuracy when rejecting low-quality fingerprints. For this purpose, the FNMR can be plotted vs increasing amounts of rejections (see Fig. 3.65): curves closer to the origin denote more effective quality estimation algorithms. A mix of fingerprint feature extraction and matching techniques should be considered to avoid that results are biased by a specific algorithm.

3.11.3 NFIQ (NIST Fingerprint Image Quality)

NFIQ 1.0

The quality $q(x_i)$ of fingerprint image x_i is defined as the prediction of its normalized matching score $normscore(x_i)$. Let T be a training set containing n fingers and a pair of image samples (x_i, x_i') for each finger $i = 1, \ldots, n$. The normalized matching score of x_i is defined according to the separation of the x_i genuine matching score from the x_i impostor scores:

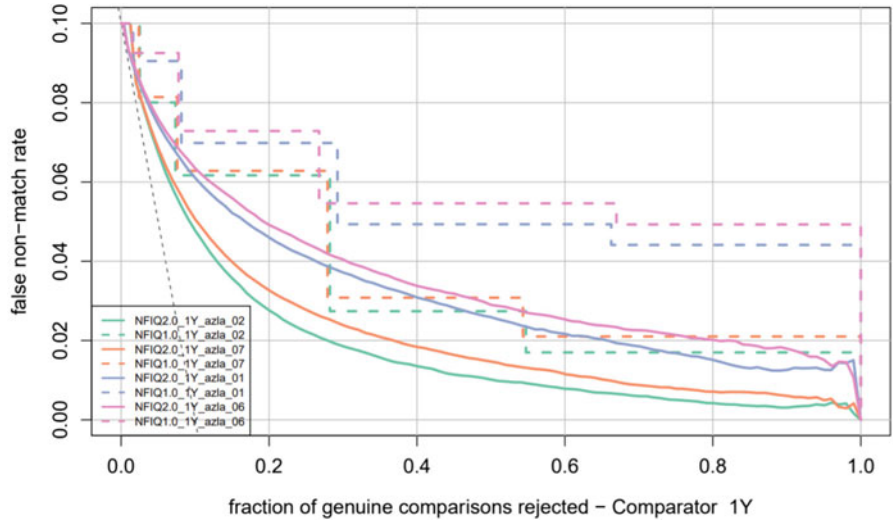

Fig. 3.65 In this graph, extracted from Tabassi et al. (2021), FNMR is plotted for different amounts of rejections. Fingerprints are sorted by quality so that the first samples rejected are the lowest quality ones. The solid green curve corresponding to an NFIQ 2.0 approach is here the most effective. Images courtesy of E. Tabassi

$$normscore(x_i) = \frac{score\left(x_i, x_i'\right) - \text{avg}_{j=1\ldots n, j\neq i}\left(score\left(x_i, x_j\right)\right)}{\text{stdev}_{j=1\ldots n, j\neq i}\left(score\left(x_i, x_j\right)\right)},$$

where *score* (*a*, *b*) returns the matching score between the two fingerprints *a* and *b* according to a given automatic matcher, avg() is the average value, and stdev() the standard deviation of the scores.

Given a feature vector $\mathbf{v}_i$ extracted from x_i, a mapping between $\mathbf{v}_i$ and $q(x_i)$ can be found by regression over T by considering the pairs $< \mathbf{v}_i$, *normscore* $(x_i) >$, $i = 1\ldots n$. Tabassi et al. (2004) preferred to formulate the problem as a classification problem (instead of as a regression problem) in order to quantize the fingerprint quality into just five values; for this purpose, a neural network classifier is trained to classify the feature vector into one of the five predefined quality classes (where class one means top quality and class five the worst quality). Each feature vector is constituted by 11 features. To extract these features, the method calculates a quality map of the foreground according to the consistency of local orientation, the local contrast, and the curvature. Minutiae detection is then performed and the reliability of each detected minutia point is computed according to simple pixel intensity statistics (mean and standard deviation) within the immediate neighborhood of the minutia point. The minutiae reliability is then combined with the local quality at the minutiae location (from the quality map) to produce a quality measure for each minutia.

NFIQ 2.0

NFIQ 2.0, the natural replacement of NFIQ, is described in Tabassi et al. (2021). The main desiderata considered during its development were: better efficiency, finer granularity (101 quality bins instead of 5), and modular design. A comprehensive literature survey was initially performed to select 155 candidate features: 14 features were finally chosen according to their predictive power, computation time, and correlation (see Table 3.2). Feature vectors built by concatenating the above features were then used to train a Random Forest classifier that predicts the fingerprint quality.

A lite version of NFIQ2.0 was proposed by Tabassi et al. (2013). This method is based on the Bag-of-Words approach and can be efficiently run on low-resource clients. A large training set of blocks (e.g., 24×24 pixels) extracted from fingerprint images are clustered by spatial similarity to form a codebook (or visual dictionary). For this purpose, the authors used a self-organizing map (SOM) instead of a classical K-means approach. One of the reasons is that the topological ordering resulting from SOM clustering allows to speed up the search of the Best Matching Unit (BMU) during the online phase. Given a codebook, a fixed-size feature vector can be extracted from a fingerprint image by (i) subdividing it into blocks; (ii) searching the BMU in the codebook for each block, and (iii) constructing an histogram of the BMU indices. A random forest classifier is then trained to map feature vectors to quality bins.

Table. 3.2 Features used in NFIQ2.0. To build fixed-length feature vectors, the local features (estimated block-wise) are aggregated by computing the mean, standard deviation, and histogram. NFIQ2.0 quality features are standardized as part of ISO/IEC 29794-4 (2017) and serve as the reference implementation of the standard

Number	Type	Feature
1	Local	FDA: Frequency-Domain Analysis
2	Local	LCS: Local Clarity Score
3	Local	OCS: Orientation Certainty Level
4	Local	OFL: Orientation Flow
5	Local	RVU: Ridge–Valley Uniformity
6	Global	MU: Mean Gray Level
7	Global	MMB: Mean of Blocks Gray-Level Mean
8	Global	Minutiae Count (detected by FingerJetFX)
9	Global	Minutiae Count in the Central Region (detected by FingerJetFX)
10	Global	The Mean of Minutiae Quality (based on gray-level stats)
11	Global	The Mean of Minutiae Quality (based on orientation coherence)
12	Global	ROI Area Mean
13	Global	ROI Orientation Map Coherence Sum
14	Global	ROI Relative Orientation Map Coherence Sum

3.12 Summary

Most of the early work in fingerprint analysis was based on general-purpose image processing techniques, and therefore, the resulting matching performance was often inadequate. Subsequently, several special-purpose (domain-specific) algorithms have been designed for processing fingerprint images and exploiting the peculiarity of fingerprint patterns; an example is a contextual enhancement. In the last decade, machine learning approaches have been applied to fingerprint processing and feature extraction, initially based on dictionaries and bag-of-words approaches and more recently relying on successful deep learning models such as CNN. This allowed to extract more reliable fingerprint representations especially in very poor-quality fingerprints. In particular, fully automated processing of latent fingerprints (i.e., including segmentation and minutiae extraction) has been made possible by designing techniques leveraging both learned models and domain knowledge.

Today, fingerprint image processing and feature extraction are undoubtedly a mature field, and effective algorithms are available for most of the problems: local orientation and local frequency estimation, foreground segmentation, image enhancement, singularity detection, minutiae, pore extraction, etc. Automatic processing of latent fingerprints is also at hand, even if here we can expect a further improvement in terms of accuracy and efficiency.

Deep learning approaches often achieve state-of-the-art accuracy on many of the above tasks, but usually, this comes at the cost of computing intensive algorithm running on a GPU. Many low-resource clients or embedded devices do not have such processing facilities, but fortunately, traditional approaches work well enough for most commercial applications. For example, a recent benchmarking on different enhancement techniques confirms that contextual filtering based on the Gabor filters is still one of the most effective algorithms (Schuch et al., 2018b).

References

Abhyankar, A., & Schuckers, S. (2010). Towards integrating level-3 features with perspiration pattern for robust fingerprint recognition. In *2010 IEEE International Conference on Image Processing* (pp. 3085–3088). https://doi.org/10.1109/ICIP.2010.5654261.

Almansa, A., & Lindeberg, T. (1997). Enhancement of fingerprint images using shape-adapted scale-space operators. In J. Sporring, M. Nielsen, L. Florack & P. Johansen (Eds.), *Gaussian Scale-space Theory* (pp. 21–30). Kluwer.

Almansa, A., & Lindeberg, T. (2000). Fingerprint enhancement by shape adaptation of scale-space operators with automatic scale selection. *IEEE Transactions on Image Processing, 9*(12), 2027–2042.

Alonso-Fernandez, F., Fierrez-Aguilar, J., & Ortega-Garcia, J. (2005). An enhanced Gabor filter-based segmentation algorithm for fingerprint recognition systems. In *Proceedings of International Symposium on Image and Signal Processing and Analysis.*

Alonso-Fernandez, F., Fierrez, J., Ortega-Garcia, J., Gonzalez-Rodriguez, J., Fronthaler, H., Kollreider, K., & Bigun, J. (2007). A comparative study of fingerprint image-quality estimation methods. *IEEE Transactions on Information Forensics and Security, 2*(4), 734–743.

Anand, V., & Kanhangad, V. (2019). Pore detection in high-resolution fingerprint images using deep residual network. *Journal of Electronic Imaging*, *28*(2), 020502.

ANSI/INCITS. (2004). INCITS 378-2004—Finger minutiae format for data interchange. ANSI/INCITS standard.

ANSI/NIST—CDEFFS group. (2008). Data format for the interchange of extended fingerprint and palmprint features—Addendum to ANSI/NIST–ITL 1–2007. ANSI/NIST, Working Draft 0.2. http://www.fingerprint.nist.gov/standard/cdeffs. Accessed 27 Nov 2008.

Araque, J. L., Baena, M., Chalela, B. E., Navarro, D., & Vizcaya P. R. (2002). Synthesis of fingerprint images. In *Proceedings of 16th International Conference on Pattern Recognition* (Vol. 2, pp. 442–445).

Arcelli, C., & Baja, G. S. D. (1984). A width independent fast thinning algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 4*(7), 463–474.

Areekul, V., Watchareeruetai, U., Suppasriwasuseth, K., & Tantaratana, S. (2005). Separable Gabor filter realization for fast fingerprint enhancement. In *Proceedings of International Conference on Image Processing* (Vol. 3, pp. 253–256).

Areekul, V., Suppasriwasuseth, K., & Jirachawang, S. (2006). The new focal point localization algorithm for fingerprint registration. In *Proceedings of 18th International Conference on Pattern Recognition* (Vol. 4, pp. 497–500).

Ashbaugh, D. R. (1999). *Quantitative–qualitative friction ridge analysis: An introduction to basic and advanced ridgeology*. CRC Press.

Aujol, J. F., Gilboa, G., Chan, T., & Osher, S. (2006). Structure-texture image decomposition-modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, *67*, 111–136.

Ballard, D. H. (1981). Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition, 3*(2), 110–122.

Bartunek, J. S., Nilsson, M., Sallberg, B., & Claesson, I. (2013). Adaptive fingerprint image enhancement with emphasis on preprocessing of data. *IEEE Transactions on Image Processing*, *22*(2), 644–656.

Baruch, O. (1988). Line thinning by line following. *Pattern Recognition Letters, 8*(4), 271–276.

Bazen, A. M., & Gerez, S. H. (2001). Segmentation of fingerprint images. In *Proceedings of Workshop on Circuits Systems and Signal Processing*.

Bazen, A. M., & Gerez, S. H. (2002). Systematic methods for the computation of the directional fields and singular points of fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(7), 905–919.

Bazin, A. I., & Mansfield, T. (2007). An investigation of minutiae template interoperability. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 13–18).

Bernard, S., Boujemaa, N., Vitale, D., & Bricot, C. (2002). Fingerprint segmentation using the phase of multiscale Gabor wavelets. In *Proceedings of Asian Conference Computer Vision.*

Bhanu, B., Boshra, M., & Tan, X. (2000). Logical templates for feature extraction in fingerprint images. In *Proceedings of 15th International Conference on Pattern Recognition* (Vol. 2, pp. 850–854).

Bhanu, B., & Tan, X. (2001). Learned templates for feature extraction in fingerprint images. In *Proceedings of Conference on Computer Vision and Pattern Recognition* (Vol. 2, pp. 591–596).

Bhowmick, P., Bishnu, A., Bhattacharya, B. B., Kundu, M. K., Murthy, C. A., & Acharya, T. (2002). Determination of minutiae scores for fingerprint image applications. In *Proceedings of Indian Conference on Computer Vision Graphics Image Processing* (pp. 463–468).

Bhowmick, P., & Bhattacharya, B. B. (2009). Removal of digitization errors in fingerprint ridgelines using B-splines. *Pattern Recognition*, *42*(3), 465–474.

Bian, W., Luo, Y., Xu, D., & Yu, Q. (2014). Fingerprint ridge orientation field reconstruction using the best quadratic approximation by orthogonal polynomials in two discrete variables. *Pattern Recognition*, *47*(10), 3304–3313.

Bian, W., Ding, S., & Xue, Y. (2017a). Combining weighted linear project analysis with orientation diffusion for fingerprint orientation field reconstruction. *Information Sciences*, *396*, 55–71.

Bian, W., Ding, S., & Xue, Y. (2017b). Fingerprint image super resolution using sparse representation with ridge pattern prior by classification coupled dictionaries. *IET Biometrics*, *6*(5), 342–350.

Bian, W., Ding, S., & Jia, W. (2018). Collaborative filtering model for enhancing fingerprint image. *IET Image Processing*, *12*(1), 149–157.

Bian, W., Xu, D., Li, Q., Cheng, Y., Jie, B., & Ding, X. (2019). A survey of the methods on fingerprint orientation field estimation. *IEEE Access, 7*, 32644–32663.

Bigun, J., & Granlund, G. H. (1987). Optimal orientation detection of linear symmetry. In *Proceedings of 1st International Conference on Computer Vision* (pp. 433–438).

Bolle, R., Serior, A. W., Ratha, N. K., & Pankanti, S. (2002). Fingerprint minutiae: A constructive definition. In *Proceedings of ECCV Workshop on Biometric Authentication* (pp. 58–66).

Branka, S., Marques, O., & Nešković, A. (2019). *Segmentation and separation of overlapped latent fingerprints: Algorithms, techniques, and datasets.* Briefs in Computer Science. Springer.

Buades, A., Le, T. M., Morel, J., & Vese, L. A. (2010). Fast cartoon + texture image filters. *IEEE Transactions on Image Processing*, *19*(8), 1978–1986.

Can, X., & Lin, Y. (2009). An adaptive algorithm for smoothing fingerprint orientation fields. In *Proceedings of International Conference on Computational Intelligence and Natural Computing,* Wuhan, China (pp. 70–72).

Canyellas, N., Cantó, E., Forte, G., &, López, M. (2005). Hardware–software codesign of a fingerprint identification algorithm. In *Proceedings of 5th International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 683–692).

Cao, K., & Jain, A. K. (2015). Latent orientation field estimation via convolutional neural network. In *Proceedings of International Conference on Biometrics (ICB)*, Phuket, Thailand.

Cao, K., Liang, J., & Tian, J. (2012). A div-curl regularization model for fingerprint orientation extraction. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems*, Arlington, VA (pp. 231–236).

Cao, K., Liu, E., & Jain, A. K. (2014). Segmentation and enhancement of latent fingerprints: A coarse to fine ridge structure dictionary. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(9), 1847–1859.

Cappelli, R., & Maltoni, D. (2009). On the spatial distribution of fingerprint singularities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(4), 742–748.

Cappelli, R., Maio, D., & Maltoni, D. (1999). Fingerprint classification based on multi-space KL. In *Proceedings of Workshop on Automatic Identification Advances Technologies* (pp. 117–120).

Cappelli, R., Maio, D., & Maltoni, D. (2000). Synthetic fingerprint-image generation. In *Proceedings of 15th International Conference on Pattern Recognition* (Vol. 3, pp. 475–478).

Cappelli, R., Maio, D., & Maltoni, D. (2009). Semi-automatic enhancement of very low quality fingerprints. In *Proceedings of International Symposium on Image and Signal Processing and Analysis*, Salzburg (pp. 678–683).

Cappelli, R., Maltoni, D., & Turroni, F. (2012). Fingerprint enhancement using contextual iterative filtering. In *Proceedings International Conference on Biometrics (ICB)*, New Delhi, India.

Champod, C., Lennard, C. J., Margot, P., & Stoilovic, M. (2016). *Fingerprints and Other Ridge Skin Impressions* (2nd ed.). CRC Press.

Chang, J. H., & Fan, K. C. (2001). Fingerprint ridge allocation in direct gray-scale domain. *Pattern Recognition, 34*(10), 1907–1925.

Chen, T., Jiang, X., & Yau, W. (2004a). Fingerprint image quality analysis. In *Proceedings of International Conference on Image Processing* (pp. 1253–1256).

Chen, X., Tian, J., Cheng, J., & Yang, X. (2004b). Segmentation of fingerprint images using linear classifier. *EURASIP Journal on Applied Signal Processing, 2004*(4), 480–494.

Chen, Y., Dass, S. C., & Jain, A. K. (2005). Fingerprint quality indices for predicting authentication performance. In *Proceedings of 5th International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 160–170).

Chen, J., Chan, F., & Moon, Y. S. (2007). Fingerprint matching with minutiae quality score. In *Proceedings of International Conference on Biometrics* (pp. 663–672).

Chen, F., Feng, J., Jain, A. K., Zhou, J., & Zhang, J. (2011a). Separating overlapped fingerprints. *IEEE Transactions on Information Forensics and Security*, *6*(2), 346–359.

Chen, H., Pang, L., Liang, J., Liu, E., & Tian, J. (2011b). Fingerprint singular point detection based on multiple-scale orientation entropy. *IEEE Signal Processing Letters*, *18*(11), 679–682.

Chen, C., Feng, J., & Zhou, J. (2016). Multi-scale dictionaries based fingerprint orientation field estimation. In *Proceedings of International Conference on Biometrics* (pp. 1–8).

Cheng, J., & Tian, J. (2004). Fingerprint enhancement with dyadic scale-space. *Pattern Recognition Letters, 25*(11), 1273–1284.

Chikkerur, S., & Ratha, N. (2005). Impact of singular point detection on fingerprint matching performance. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 207–212).

Chikkerur, S., Govindaraju, V., Pankanti, S., Bolle, R., & Ratha, N. (2005). Novel approaches for minutiae verification in fingerprint images. In *Proceedings of Workshops on Application of Computer Vision* (Vol. 1, pp. 111–116).

Chikkerur, S., Cartwright, A. N., & Govindaraju, V. (2007). Fingerprint enhancement using STFT analysis. *Pattern Recognition, 40*(1), 198–211.

Chugh, T., Arora, S. S., Jain, A. K., & Paulter, N. G. (2017). Benchmarking fingerprint minutiae extractors. In *Proceedings of International Conference of the Biometrics Special Interest Group.*

Coetzee, L., & Botha, E. C. (1993). Fingerprint recognition in low quality images. *Pattern Recognition, 26*(10), 1441–1460.

Da Costa, J. P., Le Pouliquen, F., Germain, C., & Baylou, P. (2001). New operators for optimized orientation estimation. In *Proceedings of International Conference on Image Processing.*

Darlow, L. N., & Rosman, B. (2017). Fingerprint minutiae extraction using deep learning. In *Proceedings of International Joint Conference on Biometrics*, Denver, CO (pp. 22–30).

Dass, S. C. (2004). Markov random field models for directional field and singularity extraction in fingerprint images. *IEEE Transactions on Image Processing, 13*(10), 1358–1367.

Daugman, J. G. (1985). Uncertainty relation for resolution in space, spatial-frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal Optical Society American, 2*, 1160–1169.

Deerada, C., Phromsuthirak, K., Rungchokanun, A., & Areekul, V. (2020). Progressive focusing algorithm for reliable pose estimation of latent fingerprints. *IEEE Transactions on Information Forensics and Security*, *15*, 1232–1247.

Ding, B., Wang, H., Chen, P., Zhang, Y., Guo, Z., Feng, J., & Liang, R. (2021). Surface and internal fingerprint reconstruction from optical coherence tomography through convolutional neural network. *IEEE Transactions on Information Forensics and Security*, *16*, 685–700.

Donahue, M. L., & Rokhlin, S. I. (1993). On the use of level curves in image analysis. *CVGIP: Image Understanding, 57*(2), 185–203.

Donida Labati, R., Genovese, A., Muñoz, E., Piuri, V., & Scotti, F. (2018). A novel pore extraction method for heterogeneous fingerprint images using convolutional neural networks. *Pattern Recognition Letters*, *113*, 58–66.

Duda, R. O., Hart, P. E., & Stork, D. G. (2000). *Pattern classification*, 2nd edn. Wiley.

Erol, A., Halici, U., & Ongun, G. (1999). Feature selective filtering for ridge extraction. In L.C. Jain, U. Halici, I. Hayashi & S.B. Lee (Eds.), *Intelligent biometric techniques in fingerprint & face recognition*. CRC Press.

Ezeobiejesi, J., & Bhanu, B. (2017). Latent fingerprint image segmentation using deep neural network. In B. Bhanu & A. Kumar (Eds.), *Deep learning for biometrics*. Springer.

Fan, L., Wang, S., Wang, H., & Guo, T. (2008). Singular points detection based on zero-pole model in fingerprint images. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 30*(6), 929–940.

Farina, A., Kovacs-Vajna, Z. M., Leone, A. (1999). Fingerprint minutiae extraction from skeletonized binary images. *Pattern Recognition, 32*(5), 877–889.

Feng, J., Shi, Y., & Zhou, J. (2012). Robust and efficient algorithms for separating latent overlapped fingerprints. *IEEE Transactions on Information Forensics and Security*, *7*(5), 1498–1510.

Feng, J., Zhou, J., & Jain, A. K. (2013). Orientation field estimation for latent fingerprint enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(4), 925–940.

Fitz, A. P., & Green, R. J. (1996). Fingerprint classification using hexagonal fast Fourier transform. *Pattern Recognition, 29*(10), 1587–1597.

Freeman, W. T., & Adelson, E. H. (1991). The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 13*(9), 891–906.

Fronthaler, H., Kollreider, K., & Bigun, J. (2006). Automatic image quality assessment with application in biometrics. In *Proceedings of CVPR Workshop on Biometrcis* (pp. 30–35).

Fronthaler, H., Kollreider, K., & Bigun, J. (2007). Pyramid-based image enhancement of fingerprints. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 45–50).

Fronthaler, H., Kollreider, K., & Bigun, J. (2008a). Local features for enhancement and minutiae extraction in fingerprints. *IEEE Transactions on Image Processing, 17*(3), 354–363.

Fronthaler, H., Kollreider, K., Bigun, J., Fierrez, J., Alonso-Fernandez, F., Ortega-Garcia, J., & Gonzalez-Rodriguez, J. (2008b). Fingerprint image-quality estimation and its application to multialgorithm verification. *IEEE Transactions on Information Forensics and Security, 3*(2), 331–338.

Gall, J., & Lempitsky, V. (2013). Class-specific Hough forests for object detection. In A. Criminisi & J. Shotton (Eds.), *Decision forests for computer vision and medical image analysis*. Springer.

Galton, F. (1892). *Finger prints*. Macmillan.

Gamassi, M., Piuri, V., & Scotti, F. (2005). Fingerprint local analysis for high-performance minutiae extraction. In *Proceedings of International Conference on Image Processing* (Vol. 3, pp. 265–268).

Girshick, R. (2015). Fast R-CNN. In *Proceedings of International Conference on Computer Vision* (pp. 1440–1448).

Gonzales, R. C., & Woods, R. E. (2007). *Digital image processing*, 3rd edn. Prentice-Hall.

Gottschlich, C. (2012). Curved-region-based ridge frequency estimation and curved gabor filters for fingerprint image enhancement. *IEEE Transactions on Image Processing*, *21*(4), 2220–2227.

Gottschlich, C., & Schönlieb, C. (2012). Oriented diffusion filtering for enhancing low-quality fingerprint images. *IET Biometrics*, *1*(2), 105–113.

Gottschlich, C., Mihailescu, P., & Munk, A. (2009). Robust orientation field estimation and extrapolation using semilocal line sensors. *IEEE Transactions on Information Forensics and Security*, *4*(4), 802–811.

Gottschlich, C., Tams, B., & Huckemann, S. (2017). Perfect fingerprint orientation fields by locally adaptive global models. *IET Biometrics*, *6*(3), 183–190.

Govindaraju, V., Shi, Z., & Schneider, J. (2003). Feature extraction using a chaincoded contour representation of fingerprint images. In *Proceedings of 4th International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 268–275.)

Grasselli, A. (1969). On the automatic classification of fingerprints. In S. Watanabe (Ed.), *Methodologies of pattern recognition.* Academic.

Greenberg, S., Aladjem, M., Kogan, D., & Dimitrov, I. (2000). Fingerprint image enhancement using filtering techniques. In *Proceedings of 15th International Conference on Pattern Recognition* (Vol. 3, pp. 326–329).

Grosz, S. A., Engelsma, J. J., Paulter, N. G., & Jain, A. K. (2020). White-box evaluation of fingerprint matchers: Robustness to minutiae perturbations. In *Proceedings of International Joint Conference on Biometrics* (pp. 1–10).

Grother, P., & Tabassi, E. (2007). Performance of biometric quality measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(4), 531–543.

Grother, P., McCabe, M., Watson, C., Indovina, M., Salamon, W., Flanagan, P., Tabassi, E., Newton, E., & Wilson, C. (2006). *Performance and Interoperability of the INCITS 378 Fingerprint Template.* NIST Research Report: NISTIR 7296.

Gu, J., Zhou, J., & Yang, C. (2006). Fingerprint recognition by combining global structure and local cues. *IEEE Transactions on Image Processing, 15*(7), 1952–1964.

Gu, S., Feng, J., Lu, J., & Zhou, J. (2018). Efficient rectification of distorted fingerprints. *IEEE Transactions on Information Forensics and Security*, *13*(1), 156–169.

Hara, M. (2011a). System for recognizing fingerprint image, method and program for the same. US Patent 8019132.

Hara, M. (2011b). Fingerprint collation apparatus, fingerprint pattern area extracting apparatus and quality judging apparatus, and method and program of the same. US Patent 7885437.

Hari, V. S., Jagathy Raj, V. P., & Gopikakumari, R. (2013). Unsharp masking using quadratic filter for the enhancement of fingerprints in noisy background. *Pattern Recognition*, *46*(12), 3198–3207.

He, Y., Tian, J., Luo, X., & Zhang, T. (2003). Image enhancement and minutiae matching in fingerprint verification. *Pattern Recognition Letters, 24*(9), 1349–1360.

Henry, E. (1900). *Classification and uses of finger prints.*

Hicklin, R. A. (2009). Anatomy of friction ridge skin. In S. Z. Li & A. K. Jain (Eds.), *Encyclopedia of biometrics.* Springer.

Hong, L., Jain, A. K., Pankanti, S., & Bolle, R. (1996). Fingerprint enhancement. In *Proceedings of Workshop on Applications of Computer Vision* (pp. 202–207).

Hong, L., Wan, Y., & Jain, A. K. (1998). Fingerprint image enhancement: Algorithms and performance evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20*(8), 777–789.

Hotz, T. (2009). Intrinsic coordinates for fingerprints based on their longitudinal axis. In *Proceedings of International Symposium on Image and Signal Processing and Analysis*, Salzburg (pp. 500–504).

Hou, Z., & Yau, W. (2010). A variational formulation for fingerprint orientation modelling. In *Proceedings of International Conference on Pattern Recognition*, Istanbul (pp. 1626–1629).

Hsieh, C. T., Lai, E., & Wang, Y. C. (2003). An effective algorithm for fingerprint image enhancement based on wavelet transform. *Pattern Recognition, 36*(2), 303–312.

Huang, C. Y., Liu, L. M., & Hung, D. C. D. (2007). Fingerprint analysis and singular point detection. *Pattern Recognition Letters, 28*(15), 1937–1945.

Huckemann, S., Hotz, T., & Munk, A. (2008). Global models for the orientation field of fingerprints: An approach based on quadratic differentials. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 30*(9), 1507–1519.

Hung, D. C. D. (1993). Enhancement and feature purification of fingerprint images. *Pattern Recognition, 26*(11), 1661–1671.

Hung, D. C. D., & Huang, C. (1996). A model for detecting singular points of a fingerprint. In *Proceedings of 9th Florida Artificial Intelligence Research Symposium* (pp. 444–448).

Hwang, K. (2004). Statistical quality assessment of a fingerprint. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification I*.

Ikeda, N., Nakanishi, M., Fujii, K., Hatano, T., Shigematsu, S., Adachi, T., Okazaki, Y., & Kyuragi, H. (2002). Fingerprint image enhancement by pixel-parallel processing. In *Proceedings of 16th International Conference on Pattern Recognition* (Vol. 3, pp. 752–755).

ISO/IEC 19794-2. (2011). ISO, "ISO/IEC 19794-2:2011 – Information technology – Biometric data interchange formats – Part 2: Finger minutiae data". Retrieved July, 2021, from https://www.iso.org/standard/50864.html.

ISO/IEC 29794-4. (2017). *Information technology—Biometric sample quality—Part 4: Finger image data.* ISO/IEC Standard.

Isola, P., Zhu, J., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of Conference on Computer Vision and Pattern Recognition* (pp. 5967–5976).

Jain, A. K., & Farrokhnia, F. (1991). Unsupervised texture segmentation using Gabor filters. *Pattern Recognition, 24*(12), 1167–1186.

Jain, A. K., Hong, L., Pankanti, S., & Bolle, R. (1997). An identity authentication system using fingerprints. *Proceedings of the IEEE, 85*(9), 1365–1388.

Jain, A. K., Prabhakar, S., Hong, L., & Pankanti, S. (2000). Filterbank-based fingerprint matching. *IEEE Transactions on Image Processing, 9*(5), 846–859.

Jain, A. K., Chen, Y., & Demirkus, M. (2007). Pores and ridges: High-resolution fingerprint matching using Level 3 features. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29*(1), 15–27.

Jang, W., Park, D., Lee, D., & Kim, S. J. (2006). Fingerprint image enhancement based on a half gabor filter. In *Proceedings of International Conference on Biometrics* (pp. 258–264).

Jang, H., Kim, D., Mun, S., Choi, S., & Lee, H. (2017). DeepPore: Fingerprint pore extraction using deep convolutional neural networks. *Signal Processing Letters*, *24*(12), 1808–1812.

Ji, L., & Yi, Z. (2008). Fingerprint orientation field estimation using ridge projection. *Pattern Recognition, 41*(5), 1508–1520.

Ji, L., Yi, Z., Shang, L., & Pu, X. (2007). Binary fingerprint image thinning using template-based PCNNs. *IEEE Transaction on Systems, Man, and Cybernetics, Part B, 37*(5), 1407–1413.

Jiang, X. (2000). Fingerprint image ridge frequency estimation by higher order spectrum. In *Proceedings of International Conference on Image Processing*.

Jiang, X. (2001). A study of fingerprint image filtering. In *Proceedings of International Conference on Image Processing*.

Jiang, X., Yau, W. Y., & Ser, W. (1999). Minutiae extraction by adaptive tracing the gray level ridge of the fingerprint image. In *Proceedings of International Conference on Image Processing*.

Jiang, X., Yau, W. Y., & Ser, W. (2001). Detecting the fingerprint minutiae by adaptive tracing the gray-level ridge. *Pattern Recognition, 34*(5), 999–1013.

Jiang, X., Liu, M., & Kot, A. C. (2004). Reference point detection for fingerprint recognition. In *Proceedings of 17th International Conference on Pattern Recognition* (Vol. 1, pp. 540–543).

Jiang, L., Zhao, T., Bai, C., Yong, A., & Wu, M. (2016). A direct fingerprint minutiae extraction approach based on convolutional neural networks. In *Proceedings of International Joint Conference on Neural Networks*, Vancouver, BC (pp. 571–578).

Jirachaweng, S., & Areekul, V. (2007). Fingerprint enhancement based on discrete cosine transform. In *Proceedings of International Conference on Biometrics* (pp. 96–105).

Jirachaweng, S., Hou, Z., Yau, W. Y., & Areekul, V. (2011). Residual orientation modeling for fingerprint enhancement and singular point detection. *Pattern Recognition*, *44*(2), 431–442.

Jolliffe, I. T. (1986). *Principle component analysis*. Springer.

Joshi, I., Anand, A., Vatsa, M., Singh, R., Roy, S. D., & Kalra, P. (2019). Latent fingerprint enhancement using generative adversarial networks. In *Proceedings Winter Conference on Applications of Computer Vision*, Waikoloa Village, HI, USA (pp. 895–903).

Kamei, T. (2004). Image filter design for fingerprint enhancement. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems* (pp. 113–126). Springer.

Kamei, T., & Mizoguchi, M. (1995). Image filter design for fingerprint enhancement. In *Proceedings of International Symposium on Computer Vision* (pp. 109–114).

Karu, K., & Jain, A. K. (1996). Fingerprint classification. *Pattern Recognition, 29*(3), 389–404.

Kass, M., & Witkin, A. (1987). Analyzing oriented patterns. *Computer Vision Graphics and Image Processing, 37*(3), 362–385.

Kawagoe, M., & Tojo, A. (1984). Fingerprint pattern classification. *Pattern Recognition, 17*, 295–303.

Kayaoglu, M., Topcu, B., & Uludag, U. (2013). Standard fingerprint databases: Manual minutiae labeling and matcher performance analyses. arXiv:1305.1443.

Khan, M. A. U., Khan, T. M., Bailey, D. G., & Kong, Y. (2016). A spatial domain scar removal strategy for fingerprint image enhancement. *Pattern Recognition*, *60*, 258–274.

Kim, D. H. (2005). Minutiae quality scoring and filtering using a neighboring ridge structural analysis on a thinned fingerprint image. In *Proceedings of 5th International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 674–682).

Kim, B. G., & Park, D. J. (2002). Adaptive image normalisation based on block processing for enhancement of fingerprint image. *Electronics Letters, 38*(14), 696–698.

Kim, S., Lee, D., & Kim, J. (2001). Algorithm for detection and elimination of false minutiae in fingerprint images. In *Proceedings of 3rd International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 235–240).

Kohonen, T., Kangas, J., Laaksonen, J., & Torkkola, K. (1992). LVQ_PAQ: A program package for the correct application of learning vector quantization algorithms. In *Proceedings of International Joint Conference On Neural Network* (pp. 1725–1730).

Koo, W. M., & Kot, A. (2001). Curvature-based singular points detection. In *Proceedings of 3rd International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 229–234).

Kovacs-Vajna, Z. M. (2000). A fingerprint verification system based on triangular matching and dynamic time warping. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22*, 1266–1276.

Kovacs-Vajna, Z. M., Rovatti, R., & Frazzoni, M. (2000). Fingerprint ridge distance computation methodologies. *Pattern Recognition, 33*(1), 69–80.

Kryszczuk, K., & Drygajlo, A. (2006). Singular point detection in finger-prints using quadrant change information. In *Proceedings of 18th International Conference on Pattern Recognition* (Vol. 4, pp. 594–597).

Kryszczuk, K. M., Morier, P., & Drygajlo, A. (2004). Study of the distinctiveness of level 2 and level 3 features in fragmentary fingerprint comparison. In *Proceedings of ECCV Workshop on Biometric Authentication* (pp. 124–133).

Lam, L., Lee, S. W., & Suen, C. Y. (1992). Thinning methodologies: A comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 14*(9), 869–885.

Landy, M. S., Cohen, Y., & Sperling, G. (1984). Hips: A Unix-based image processing system. *Computer Vision, Graphics and Image Processing, 25*(3), 331–347.

Larkin, K. G. (2005). Uniform estimation of orientation using local and nonlocal 2-D energy operators. *Optics Express, 13*(20), 8097–8121.

Larkin, K. G., & Fletcher, P. A. (2007). A coherent framework for fingerprint analysis: Are fingerprints holograms? *Optics Express*, *15*(14), 8667–8677.

Le, T. H., & Van, H. T. (2012). Fingerprint reference point detection for image retrieval based on symmetry and variation. *Pattern Recognition*, *45*(9), 3360–3372.

Lee, K., & Prabhakar, S. (2008). Probabilistic orientation field estimation for fingerprint enhancement and verification. In *Proceedings on Biometric Symposium.*

Lee, B., Moon, J., & Kim, H. (2005). A novel measure of fingerprint image quality using the Fourier spectrum. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification II*.

Lee, C., Lee, S., Kim, J., & Kim, S. J. (2006). Preprocessing of a fingerprint image captured with a mobile camera. In *Proceedings of International Conference on Biometrics* (pp. 348–355).

Lehtihet, R., El Oraiby, W., & Benmohammed, M. (2014). Ridge frequency estimation for low-quality fingerprint images enhancement using Delaunay triangulation. *International Journal of Pattern Recognition and Artificial Intelligence*, *28*(1), 1456002.

Leung, M., Engeler, W., & Frank, P. (1990). Fingerprint image processing using neural network. In *Proceedings of IEEE Region 10 Conference on Computer and Communications Systems.*

Leung, W. F., Leung, S. H., Lau, W. H., & Luk, A. (1991). Fingerprint recognition using neural network. In *Proceedings of Workshop Neural Network for Signal Processing.*

Li, J., Yau, W. Y., & Wang, H. (2006). Constrained nonlinear models of fingerprint orientations with prediction. *Pattern Recognition, 39*(1), 102–114.

Li, G., Busch, C., & Yang, B. (2014). A novel approach used for measuring fingerprint orientation of arch fingerprint. In *Proceedings of International Convention on Information and Communication Technology, Electronics and Microelectronics*, Opatija (pp. 1309–1314).

Li, J., Feng, J., & Kuo, C. C. J. (2018). Deep convolutional neural network for latent fingerprint enhancement. *Signal Processing: Image Communication*, *60*, 52–63.

Liang, X., & Asano, T. (2006). A linear time algorithm for binary fingerprint image denoising using distance transform. *IEICE Transactions on Information and Systems, 89*(4), 1534–1542.

Lim, E., Jiang, X., & Yau, W. (2002). Fingerprint quality and validity analysis. In *Proceedings of International Conference on Image Processing* (Vol. 1, pp. 469–472).

Lim, E., Toh, K. A., Suganthan, P. N., Jiang, X., & Yau, W. Y. (2004). Fingerprint image quality analysis. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 1241–1244).

Lin, W., & Dubes, R. (1983). A review of ridge counting in dermatoglyphics. *Pattern Recognition, 16*(1), 1–8.

Liu, J., Huang, Z., & Chan, K. (2000). Direct minutiae extraction from gray-level fingerprint image by relationship examination. In *Proceedings of International Conference on Image Processing.*

Liu, M., Jiang, X., & Kot, A. C. (2004). Fingerprint reference point detection. In *Proceedings of International Conference on Biometric Authentication* (pp. 272–279).

Liu, T., Zhu, G., Zhang, C., & Hao, P. (2005). Fingerprint indexing based on singular point correlation. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 293–296).

Liu, T., Zhang, C., & Hao, P. (2006). Fingerprint reference point detection based on local axial symmetry. In *Proceedings of 18th International Conference on Pattern Recognition* (Vol. 1, pp. 1050–1053).

Liu, S., & Liu, M. (2012). Fingerprint orientation modeling by sparse coding. In *Proceedings of International Conference on Biometrics*, New Delhi (pp. 176–181).
Liu, M., Liu, S., & Zhao, Q. (2014). Fingerprint orientation field reconstruction by weighted discrete cosine transform. *Information Sciences*, *268*, 65–77.
Liu, E., & Cao, K. (2016). Minutiae extraction from level 1 features of fingerprint. *IEEE Transactions on Information Forensics and Security*, *11*(9), 1893–1902.
Liu, S., Liu, M., & Yang, Z. (2017). Sparse coding based orientation estimation for latent fingerprints. *Pattern Recognition*, *67*, 164–176.
Liu, J., Yan, J., Deng, D., & Zhang, R. (2020). Fingerprint image quality assessment based on BP neural network with hierarchical clustering. *IET Information Security*, *14*(2), 185–195.
Luo, X., & Tian, J. (2000). Knowledge based fingerprint image enhancement. In *Proceedings of 15th International Conference on Pattern Recognition* (Vol. 4, pp. 783–786).
Ma, C., & Zhu, Y. (2013). Analysis and extraction of fingerprint features based on principal curves. *Journal of Computational Information Systems*, *9*(21), 8591–8601.
Maio, D., & Maltoni, D. (1996). A structural approach to fingerprint classification. In *Proceedings of 13th International Conference on Pattern Recognition.*
Maio, D., & Maltoni, D. (1997). Direct gray-scale minutiae detection in fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 19*(1).
Maio, D., & Maltoni, D. (1998a). Ridge-line density estimation in digital images. In *Proceedings of 14th International Conference on Pattern Recognition* (pp. 1654–1658).
Maio, D., & Maltoni, D. (1998b). Neural network based minutiae filtering in fingerprints. In *Proceedings of 14th International Conference on Pattern Recognition* (pp. 1654–1658).
Maio, D., Maltoni, D., Cappelli, R., Wayman, J. L., & Jain, A. K. (2002). FVC2000: Fingerprint verification competition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(3), 402–412.
Malathi, S., Uma Maheswari, S., & Meena, C. (2010). Fingerprint pore extraction based on marker controlled watershed segmentation. In *Proceedings of International Conference on Computer and Automation Engineering*, Singapore (pp. 337–340).
Mallat, S. G. (1989). A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 11*(7), 674–693.
Mehtre, B. M. (1993). Fingerprint image analysis for automatic identification. *Machine Vision and Applications, 6*, 124–139.
Mehtre, B. M., Murthy, N. N., Kapoor, S., & Chatterjee, B. (1987). Segmentation of fingerprint images using the directional image. *Pattern Recognition, 20*(4), 429–435.
Mei, Y., Sun, H., & Xia, D. (2009). A gradient-based combined method for the computation of fingerprints' orientation field. *Image and Vision Computing*, *27*(8), 1169–1177.
Miao, D., Tang, Q., & Fu, W. (2007). Fingerprint minutiae extraction based on principal curves. *Pattern Recognition Letters, 28*(16), 2184–2189.
Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., & Terzopoulos, D. (2021). Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* https://doi.org/10.1109/TPAMI.2021.3059968.
Moayer, B., & Fu, K. (1986). A tree system approach for fingerprint pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 8*(3), 376–388.
Munir, M. U., Javed, M. Y., & Khan, S. A. (2012). A hierarchical k-means clustering based fingerprint quality classification. *Neurocomputing*, *85*, 62–67.
Nakamura, T., Hirooka, M., Fujiwara, H., & Sumi, K. (2004). Fingerprint image enhancement using a parallel ridge filter. In *Proceedings of 17th International Conference on Pattern Recognition* (Vol. 1, pp. 536–539).

Nguyen, D., Cao, K., & Jain, A. K. (2018a). Automatic latent fingerprint segmentation. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems.*

Nguyen, D., Cao, K., & Jain, A. K. (2018b). Robust minutiae extractor: Integrating deep networks and fingerprint domain knowledge. In *Proceedings of International Conference on Biometrics.*

Nguyen, V. H., Liu, J., Nguyen, T. H. B., & Kim, H. (2020). Universal fingerprint minutiae extractor using convolutional neural networks. *IET Biometrics*, *9*(2), 47–57.

Nilsson, K., & Bigun, J. (2001). Using linear symmetry features as a pre-processing step for fingerprint images. In *Proceedings of International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 247–252).

Nilsson, K., & Bigun, J. (2002a). Complex filters applied to fingerprint images detecting prominent points used alignment. In *Proceedings of ECCV Workshop on Biometric Authentication* (pp. 39–47). Springer.

Nilsson, K., & Bigun, J. (2002b). Prominent symmetry points as landmarks in fingerprint images for alignment. In *Proceedings of 16th International Conference on Pattern Recognition* (Vol. 3, pp. 395–398).

Nilsson, K., & Bigun, J. (2003). Localization of corresponding points in fingerprints by complex filtering. *Pattern Recognition Letters, 24*(13), 2135–2144.

NIST. (2015). Data Format for the Interchange of Fingerprint, Facial & Other Biometric Information. Update 2015 of NIST Special Publication 500-290e3.

Novikov, S. O., & Kot, V. S. (1998). Singular feature detection and classification of fingerprints using hough transform. In *Proceedings of SPIE (6th International Workshop on Digital Image Processing and Computer Graphics: Applications in Humanities and Natural Sciences)* (Vol. 3346, pp. 259–269).

O'Gorman, L., & Nickerson, J. (1988). Matched filter design for fingerprint image enhancement. In *Proceedings of International Conference on Acoustic Speech and Signal Processing* (pp. 916–919).

O'Gorman, L., & Nickerson, J. V. (1989). An approach to fingerprint filter design. *Pattern Recognition, 22*(1), 29–38.

Ohtsuka, T., & Kondo, A. (2005). A new approach to detect core and delta of the fingerprint using extended relational graph. In *Proceedings of International Conference on Image Processing* (Vol. 3, pp. 249–252).

Ohtsuka, T., & Takahashi T. (2005). A new detection approach for the fingerprint core location using extended relation graph. *IEICE Transactions on Information and Systems, 88*(10), 2308–2312.

Ohtsuka, T., & Watanabe, D. (2010). Singular candidate method: Improvement of extended relational graph method for reliable detection of fingerprint singularity. *IEICE Transactions on Information and Systems*, *E93-D*(7), 1788–1797.

Oliveira, M. A., & Leite, N. J. (2008). A multiscale directional operator and morphological tools for reconnecting broken ridges in fingerprint images. *Pattern Recognition, 41*(1), 367–377.

Orczyk, T., & Wieclaw, L. (2011). Fingerprint ridges frequency. In *Proceedings World Congress on Nature and Biologically Inspired Computing*, Salamanca (pp. 558–561).

Ouyang, J., Feng, J., Lu, J., Guo, Z., & Zhou, J. (2017). Fingerprint pose estimation based on faster R-CNN. In *Proceedings of International Joint Conference on Biometrics*, Denver, CO (pp. 268–276).

Pais Barreto Marques, A. C., & Gay Thome, A. C. (2005). A neural network fingerprint segmentation method. In *Proceedings of International Conference on Hybrid Intelligent Systems.*

Paiva, A. R. C., & Tasdizen, T. (2012). Fingerprint image segmentation using data manifold characteristic features. *International Journal of Pattern Recognition and Artificial Intelligence*, *26*(4), 1256010.

Panetta, K., Kamath, K. M. S., Rajeev, S., & Agaian, S. S. (2019). LQM: localized quality measure for fingerprint image enhancement. *IEEE Access*, *7*, 104567–104576.

Perona, P. (1998). Orientation diffusions. *IEEE Transactions on Image Processing, 7*(3), 457–467.

Prabhakar, S., Jain, A. K., & Pankanti, S. (2003). Learning fingerprint minutiae location and type. *Pattern Recognition, 36*(8), 1847–1857.

Qi, J., Shi, Z., Zhao, X., & Wang, Y. (2005a). Measuring fingerprint image quality using gradient. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification II*.

Qi, J., Abdurrachim, D., Li, D., & Kunieda, H. (2005b). A hybrid method for fingerprint image quality calculation. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 124–129).

Ram, S., Bischof, H., & Birchbauer, J. (2010). Modelling fingerprint ridge orientation using Legendre polynomials. *Pattern Recognition*, *43*(1), 342–357.

Rama, R. K. N. V., & Namboodiri, A. M. (2011). Fingerprint enhancement using hierarchical Markov random fields. In *Proceedings of International Joint Conference on Biometrics*, Washington, DC (pp. 1–8).

Rämö, P., Tico, M., Onnia, V., & Saarinen, J. (2001). Optimized singular point detection algorithm for fingerprint images. In *Proceedings of International Conference on Image Processing*.

Rao, A. R. (1990). *A taxonomy for texture description and identification.* Springer.

Ratha, N. K., Chen, S. Y., & Jain, A. K. (1995). Adaptive flow orientation-based feature extraction in fingerprint images. *Pattern Recognition, 28*(11), 1657–1672.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In *Proceedings of Advances in Neural Information Processing Systems* (pp. 91–99).

Rerkrai, K., & Areekul, V. (2000). A new reference point for fingerprint recognition. In *Proceedings of International Conference on Image Processing*.

Roddy, A., & Stosz, J. (1997). Fingerprint features: Statistical-analysis and system performance estimates. *Proceedings of the IEEE, 85*(9), 1390–1421.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of International Conference on Medical Image Computing and Computer Assisted Intervention* (pp. 234–241).

Saleh, A. M., Bahaa Eldin, A. M., & Wahdan, A. A. (2009). A modified thinning algorithm for fingerprint identification systems. In *Proceedings of International Conference on Computer Engineering & Systems*, Cairo (pp. 371–376).

Sankaran, A., Vatsa, M., & Singh, R. (2013). Automated clarity and quality assessment for latent fingerprints. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems*, Arlington, VA (pp. 1–6).

Sankaran, A., Pandey, P., Vatsa, M., & Singh, R. (2014). On latent fingerprint minutiae extraction using stacked denoising sparse AutoEncoders. In *Proceedings International Joint Conference on Biometrics*, Clearwater, FL (pp. 1–7).

Schuch, P., Schulz, S., & Busch, C. (2016). De-convolutional auto-encoder for enhancement of fingerprint samples. In *Proceedings of International Conference on Image Processing Theory, Tools and Applications*, Oulu (pp. 1–7).

Schuch, P., Schulz, S., & Busch, C. (2017a). Deep expectation for estimation of fingerprint orientation fields. In *Proceedings of International Joint Conference on Biometrics*, Denver, CO (pp. 185–190).

Schuch, P., Schulz, S., & Busch, C. (2017b). Intrinsic limitations of fingerprint orientation estimation. In *Proceedings of International Conference of the Biometrics Special Interest Group*, Darmstadt (pp. 1–5).

Schuch, P., May, J. M., & Busch, C. (2018a). Unsupervised learning of fingerprint rotations. In *Proceedings of International Conference of the Biometrics Special Interest Group*, Darmstadt (pp. 1–6).

Schuch, P., Schulz, S., & Busch, C. (2018b). Survey on the impact of fingerprint image enhancement. *IET Biometrics*, *7*(2), 102–115.

Sha, L., Zhao, F., & Tang, X. (2006). Minutiae-based fingerprint matching using subset combination. In *Proceedings of International Conference on Pattern Recognition* (Vol. 4, pp. 566–569).

Sharma, R. P., & Dey, S. (2019). Two-stage quality adaptive fingerprint image enhancement using fuzzy C-means clustering based fingerprint quality analysis. *Image and Vision Computing*, *83–84*, 1–16.

Shen, L., Kot, A., & Koo, W. M. (2001). Quality measures of fingerprint images. In *Proceedings of 3rd International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 266–271).

Shen, Z., Xu, Y., Li, J., & Lu, G. (2019). Stable pore detection for high-resolution fingerprint based on a CNN detector. In *Proceedings of International Conference on Image Processing*, Taipei, Taiwan (pp. 2581–2585).

Sherlock, B. G. (2004). Computer enhancement and modeling of fingerprint images. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems* (pp. 87–112). Springer.

Sherlock, B. G., & Monro, D. M. (1993). A model for interpreting fingerprint topology. *Pattern Recognition, 26*(7), 1047–1055.

Sherlock, B. G., Monro, D. M., & Millard, K. (1992). Algorithm for enhancing fingerprint images. *Electronics Letters, 28*(18), 1720.

Sherlock, B. G., Monro, D. M., & Millard, K. (1994). Fingerprint enhancement by directional Fourier filtering. *IEE Proceedings Vision Image and Signal Processing, 141*(2), 87–94.

Shi, Z., & Govindaraju, V. (2006a). A chaincode based scheme for fingerprint feature extraction. *Pattern Recognition Letters, 27*(5), 462–468.

Shi, Z., & Govindaraju, V. (2006b). Fingerprint image enhancement based on skin profile approximation. In *Proceedings of 18th International Conference on Pattern Recognition* (Vol. 3, pp. 714–717).

Shi, Z., Wang, Y., Qi, J., & Xu, K. (2004). A new segmentation algorithm for low quality fingerprint image. In *Proceedings of International Conference on Image and Graphics* (pp. 314–317).

Shin, J. H., Hwang, H. Y., & Chien, I. L. (2006). Detecting fingerprint minutiae by run length encoding scheme. *Pattern Recognition, 39*(6), 1140–1154.

Singh, K., Gupta, A., & Kapoor, R. (2015). Fingerprint image super-resolution via ridge orientation-based clustered coupled sparse dictionaries. *Journal of Electronic Imaging*, *24*(4), 043015.

Srinivasan, V. S., & Murthy, N. N. (1992). Detection of singular points in fingerprint images. *Pattern Recognition, 25*(2), 139–153.

Stock, R. M., & Swonger, C. W. (1969). *Development and Evaluation of a Reader of Fingerprint Minutiae*. Technical Report: XM–2478–X–1:13–17, Cornell Aeronautical Laboratory.

Stoney, D. A., & Thornton, J. I. (1987). A systematic study of epidermal ridge minutiae. *Journal of Forensic Sciences, 32*(5), 1182–1203.

Stosz, J. D., & Alyea, L. A. (1994). Automated system for fingerprint authentication using pores and ridge structure. In *Proceedings of SPIE (Automatic Systems for the Identification and Inspection of Humans)* (Vol. 2277, pp. 210–223).

Su, Y., Feng, J., & Zhou, J. (2016). Fingerprint indexing with pose constraint. *Pattern Recognition*, *54*, 1–13.

Sudiro, S. A., Paindavoine, M., & Kusuma, T. M. (2007). Simple fingerprint minutiae extraction algorithm using crossing number on valley structure. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 41–44).

Sutthiwichaiporn, P., & Areekul, V. (2013). Adaptive boosted spectral filtering for progressive fingerprint enhancement. *Pattern Recognition*, *46*(9), 2465–2486.

Svoboda, J., Monti, F., & Bronstein, M. M. (2017). Generative convolutional networks for latent fingerprint reconstruction. In *Proceedings of International Joint Conference on Biometrics* (pp. 429–436).

Székely, E., & Székely, V. (1993). Image recognition problems of fingerprint identification. *Microprocessors and Microsystems, 17*(4), 215–218.

Tabassi, E., Wilson, C., & Watson, C. (2004). *Fingerprint Image Quality*. NIST Research Report: NISTIR 7151.

Tabassi, E., Olsen, M. A., Makarov, A., & Busch, C. (2013). *Towards NFIQ II lite—Self-organizing Maps for Fingerprint Image Quality assessment*. NIST Interagency Report 79.

Tabassi, E., Olsen, M., Bausinger, O., Busch, C., Figlarz, A., Fiumara, G., Henniger, O., Merkle, J., Ruhland, T., Schiel, C., & Schwaiger, M. (2021). NFIQ 2.0—NIST Fingerprint Image Quality. NIST-IR 8382. https://doi.org/10.6028/NIST.IR.8382. Accessed July 2021.

Tamura, H. (1978). A comparison of line thinning algorithms from digital topology viewpoint. In *Proceedings of 4th International Conference on Pattern Recognition* (pp. 715–719).

Tang, Y., Gao, F., & Feng, J. (2017a). Latent fingerprint minutia extraction using fully convolutional network. In *Proceedings of International Joint Conference on Biometrics*, Denver, CO (pp. 117–123).

Tang, Y., Gao, F., Feng, J., & Liu, Y. (2017b). FingerNet: An unified deep network for fingerprint minutiae extraction. In *Proceedings of International Joint Conference on Biometrics*, Denver, CO (pp. 108–116).

Tao, X., Yang, X., Cao, K., Wang, R., Li, P., & Tian, J. (2010). Estimation of fingerprint orientation field by weighted 2D Fourier expansion model. In *Proceedings of International Conference on Pattern Recognition*, Istanbul (pp. 1253–1256).

Tao, X., Yang, X., Zang, Y., Jia, X., & Tian, J. (2012). A novel measure of fingerprint image quality using Principal Component Analysis (PCA). In *Proceedings of International Conference on Biometrics*, New Delhi (pp. 170–175).

Tashk, A., Helfroush, M. S., & Muhammadpour, M. (2009). Improvement of fingerprint orientation estimation by a modification of fingerprint orientation model based on 2D Fourier expansion (M-FOMFE). In *Proceedings International Conference on Computer, Control and Communication*, Karachi (pp. 1–6).

Teixeira, R. F. S., & Leite, N. J. (2013). On adaptive fingerprint pore extraction. In *Proceedings of International Conference on Image Analysis and Recognition* (pp. 72–79).

Teixeira, R. F. S., & Leite, N. J. (2014). Improving pore extraction in high resolution fingerprint images using spatial analysis. In *Proceedings of International Conference on Image Processing*, Paris (pp. 4962–4966).

Teixeira, R. F. S., & Leite, N. J. (2017). A new framework for quality assessment of high-resolution fingerprint images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(10), 1905–1917.

Thai, D. H., & Gottschlich, C. (2016). Global variational method for fingerprint segmentation by three-part decomposition. *IET Biometrics*, *5*(2), 120–130.

Trier, O., & Jain, A. K. (1995). Goal-directed evaluation of binarization methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 17*(12), 1191–1201.

Turroni, F., Maltoni, D., Cappelli, R., & Maio, D. (2011). Improving fingerprint orientation extraction. *IEEE Transactions on Information Forensics and Security*, *6*(3), 1002–1013.

Uchida, K. (2004). Image-based approach to fingerprint acceptability assessment. In *Proceedings of International Conference on Biometric Authentication* (pp. 294–300).

Van, T. H., & Le, H. T. (2009a). An efficient algorithm for fingerprint reference-point detection. In *Proceedings of International Conference on Computing and Communication Technologies*, Da Nang (pp. 1–7).

Van, T. H., & Le, H. T. (2009b). Adaptive noisy fingerprint enhancement based on orientation consistency. In *Proceedings of International Conference on Knowledge and Systems Engineering*, Hanoi (pp. 67–72).

Verma, M. R., Majumdar, A. K., & Chatterjee, B. (1987). Edge detection in fingerprints. *Pattern Recognition, 20*, 513–523.

Vernon, D. S. G. (1993). Automatic detection of secondary creases in fingerprints. *Optical Engineering, 32*(10), 2616–2623.

Viola, P., & Jones, M. J. (2001). Rapid object detection using a boosted cascade of simple features. In *Proceedings of International Conference on Computer Vision and Pattern Recognition* (pp. 511–518).

Vizcaya, P. R., & Gerhardt, L. A. (1996). A nonlinear orientation model for global description of fingerprints. *Pattern Recognition, 29*(7), 1221–1231.

Wahab, A., Chin, S. H., & Tan, E. C. (1998). Novel approach to automated fingerprint recognition. *IEE Proceedings Vision Image and Signal Processing, 145*(3), 160–166.

Wahab, A., Tan, E. C., & Jonatan, A. (2004). Direct gray-scale minutiae extraction. In *Proceedings of International Conference on Biometric Authentication* (pp. 280–286).

Wang, Y., & Hu, J. (2008). estimate singular point rotation by analytical models. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification V*.

Wang, Y., & Hu, J. (2011). Global ridge orientation modeling for partial fingerprint identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(1), 72–87.

Wang, L., Suo, H., & Dai, M. (2005). Fingerprint image segmentation based on Gaussian–Hermite moments. In *Proceedings of International Conference on Advanced Data Mining and Applications.*

Wang, Y., Hu, J., & Phillips, D. (2007a). A fingerprint orientation model based on 2D Fourier expansion (FOMFE) and its application to singular-point detection and fingerprint indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29*(4), 573–585.

Wang, X., Li, J., & Niu, Y. (2007b). Definition and extraction of stable points from fingerprint images. *Pattern Recognition, 40*(6), 1804–1815.

Wang, W., Li, J., Huang, F., & Feng, H. (2008). Design and implementation of Log-Gabor filter in fingerprint image enhancement. *Pattern Recognition Letters, 29*(3), 301–308.

Wang, J., Li, J., & Cao, L. (2011). An improved fast thinning algorithm for fingerprint image and its application. *Journal of Computational Information Systems*, *7*(7), 2285–2292.

Watson, C. I. (1993). NIST special database 14, fingerprint database. U.S. National Institute of Standards and Technology.

Watson, C. I., & Wilson, C. L. (1992). NIST special database 4, fingerprint database. U.S. National Institute of Standards and Technology.

Watson, C. I., Candela, G. I., & Grother, P. J. (1994). *Comparison of FFT fingerprint filtering methods for neural network classification.* Technical Report: NIST TR 5493, September 1994.

Weber, D. M. (1992). A cost effective fingerprint verification algorithm for commercial applications. In *Proceedings of South African Symposium on Communication and Signal Processing.*

Wegstein, J. H. (1982). An automated fingerprint identification system. U.S. Government Publication, U.S. Department of Commerce, National Bureau of Standards, Washington, DC.

Weng, D., Yin, Y., & Yang, D. (2011). Singular points detection based on multi-resolution in fingerprint images. *Neurocomputing*, *74*(17), 3376–3388.

Willis, A. J., & Myers, L. (2001). A cost-effective fingerprint recognition system for use with low-quality prints and damaged fingertips. *Pattern Recognition, 34*(2), 255–270.

Wong, W. J., & Lai, S. H. (2020). Multi-task CNN for restoring corrupted fingerprint images. *Pattern Recognition*, *101*, 107203.

Wu, J. C., & Garris, M. D. (2007). Nonparametric statistical data analysis of fingerprint minutiae exchange with two-finger fusion. In *Proceedings of SPIE Conference on Biometric Technology for Human Identification IV*.

Wu, C., & Govindaraju, V. (2006). Singularity preserving fingerprint image adaptive filtering. In *Proceedings of International Conference on Image Processing* (pp. 313–316).

Wu, N., & Zhou, J. (2004). Model based algorithm for singular point detection from fingerprint images. In *Proceedings of International Conference on Image Processing* (Vol. 2, pp. 885–888).

Wu, C., Zhou, J., Bian, Z., & Rong, G. (2003). Robust crease detection in fingerprint images. In *Proceedings of Conference on Computer Vision and Pattern Recognition* (Vol. II, pp. 505–510).

Xiang, M., Wu, X., & Hua, Q. (2009). A fast thinning algorithm for fingerprint image. In *Proceedings of International Conference on Information Science and Engineering*, Nanjing (pp. 1039–1042).

Xiao, Q., & Raafat, H. (1991). Fingerprint image post-processing: A combined statistical and structural approach. *Pattern Recognition, 24*(10), 985–992.

Yang, J., Liu, L., Jiang, T., & Fan, Y. (2003). A modified Gabor filter design method for fingerprint image enhancement. *Pattern Recognition Letters, 24*(12), 1805–1817.

Yang, J., Xiong, N., & Vasilakos, A. V. (2013). Two-stage enhancement scheme for low-quality fingerprint images by learning from the images. *IEEE Transactions on Human-Machine Systems*, *43*(2), 235–248.

Yang, X., Feng, J., & Zhou, J. (2014). Localized dictionaries based orientation field estimation for latent fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(5), 955–969.

Yao, M. Y. S., Pankanti, S., & Hass, N. (2004). Fingerprint quality assessment. In N. Ratha & R. Bolle (Eds.), *Automatic fingerprint recognition systems* (pp. 55–66). Springer.

Yao, Z., Le bars, J., Charrier, C., & Rosenberger, C. (2015). Quality assessment of fingerprints with minutiae delaunay triangulation. In *Proceedings of International Conference Information Systems Security and Privacy* (pp. 315–321).

Yao, Z., Le Bars, J., Charrier, C., & Rosenberger, C. (2018). Comparative study of digital fingerprint quality assessment metrics. In *Proceedings of International Conference on Biometrics*, Gold Coast, QLD (pp. 17–22).

Yin, Y., Wang, Y., & Yang, X. (2005). Fingerprint image segmentation based on quadric surface model. In *Proceedings of 5th International Conference on Audio- and Video-Based Biometric Person Authentication* (pp. 647–655).

Yoon, S., Cao, K., Liu, E., & Jain, A. K. (2013). LFIQ: latent fingerprint image quality. In *Proceedings of International Conference on Biometrics: Theory, Applications and Systems*, Washington, D.C.

Young, M. R., & Elliott, S. J. (2007). Image quality and performance based on henry classification and finger location. In *Proceedings of Workshop on Automatic Identification Advanced Technologies* (pp. 51–56).

Zacharias, G. C., Nair, M. S., & Lal, P. S. (2017). Fingerprint reference point identification based on chain encoded discrete curvature and bending energy. *Pattern Analysis and Applications*, *20*(1), 253–267.

Zhan, X., Sun, Z., Yin, Y., & Chu, Y. (2006). Fingerprint ridge distance estimation: Algorithms and the performance. In *Proceedings of International Conference on Biometrics* (pp. 294–301).

Zhang, Q., & Yan, H. (2007). Fingerprint orientation field interpolation based on the constrained Delaunay triangulation. *International Journal of Information and Systems Sciences, 3*(3), 438–452.

Zhang, D., Liu, F., Zhao, Q., Lu, G., & Luo, N. (2011a). Selecting a reference high resolution for fingerprint recognition using minutiae and pores. *IEEE Transactions on Instrumentation and Measurement*, *60*(3), 863–871.

Zhang, H., Miao, D., & Zhong, C. (2011b). Modified principal curves based fingerprint minutiae extraction and pseudo minutiae detection. *International Journal of Pattern Recognition and Artificial Intelligence*, *25*(8), 1243–1260.

Zhang, J., Lai, R., & Kuo, C. J. (2012a). Latent fingerprint segmentation with adaptive total variation model. In *Proceedings of International Conference on Biometrics*, New Delhi (pp. 189–195).

Zhang, J., Lai, R., & Kuo, C. J. (2012b). Latent fingerprint detection and segmentation with a directional total variation model. In *Proceedings of International Conference on Image Processing* (pp. 1145–1148).

Zhang, J., Lai, R., & Kuo, C. J. (2013). Adaptive directional total-variation model for latent fingerprint segmentation. *IEEE Transactions on Information Forensics and Security*, *8*(8), 1261–1273.

Zhang, N., Zang, Y., Yang, X., Jia, X., & Tian, J. (2014). Adaptive orientation model fitting for latent overlapped fingerprints separation. *IEEE Transactions on Information Forensics and Security*, *9*(10), 1547–1556.

Zhao, Q., & Jain, A. K. (2010). On the utility of extended fingerprint features: A study on pores. In *Proceedings of CVPR Workshop on Biometrics*, San Francisco.

Zhao, Q., & Jain, A. K. (2012). Model based separation of overlapping latent fingerprints. *IEEE Transactions on Information Forensics and Security*, *7*(3), 904–918.

Zhao, F., & Tang, X. (2007). Preprocessing and postprocessing for skeleton-based fingerprint minutiae extraction. *Pattern Recognition, 40*(4), 1270–1281.

Zhao, Q., Zhang, L., Zhang, D., Huang, W., & Bai, J. (2009). Curvature and singularity driven diffusion for oriented pattern enhancement with singular points. In *Proceedings of Conference on Computer Vision and Pattern Recognition*, Miami, FL (pp. 2129–2135).

Zhao, Q., Zhang, D., Zhang, L., & Luo, N. (2010a). High resolution partial fingerprint alignment using pore-valley descriptors. *Pattern Recognition*, *43*(3), 1050–1061.

Zhao, Q., Zhang, D., Zhang, L., & Luo, N. (2010b). Adaptive fingerprint pore modeling and extraction. *Pattern Recognition*, *43*(8), 2833–2844.

Zhao, Q., Liu, F., Zhang, L., & Zhang, D. (2010c). A comparative study on quality assessment of high resolution fingerprint images. In *Proceedings of International Conference on Image Processing*, Hong Kong (pp. 3089–3092).

Zhou, J., & Gu, J. (2004a). A model-based method for the computation of fingerprints' orientation field. *IEEE Transactions on Image Processing, 13*(6), 821–835.

Zhou, J., & Gu, J. (2004b). Modeling orientation fields of fingerprints with rational complex functions. *Pattern Recognition, 37*(2), 389–391.

Zhou, J., Wu, C., Bian, Z., & Zhang, D. (2004). Improving fingerprint recognition based on crease detection. In *Proceedings of International Conference on Biometric Authentication* (pp. 287–293).

Zhou, J., Gu, J., & Zhang, D. (2007). Singular points analysis in fingerprints based on topological structure and orientation field. In *Proceedings of International Conference on Biometrics* (pp. 261–270).

Zhou, J., Chen, F., Gu, J. (2009a). A novel algorithm for detecting singular points from fingerprint images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(7), 1239–1250.

Zhou, J., Chen, F., Wu, N., & Wu, C. (2009b). Crease detection from fingerprint images and its applications in elderly people. *Pattern Recognition*, *42*(5), 896–906.

Zhu, E., Yin, J., & Zhang, G. (2004). Fingerprint enhancement using circular Gabor filter. In *Proceedings of International Conference on Image on Analysis and Recognition* (pp. 750–758).

Zhu, E., Yin, J., Hu, C., & Zhang, G. (2005). Quality estimation of fingerprint image based on neural network. In *Proceedings of International Conference on Natural Computation* (pp. 65–70).

Zhu, E., Yin, J., Hu, C., & Zhang, G. (2006). A systematic method for fingerprint ridge orientation estimation and image segmentation. *Pattern Recognition, 39*(8), 1452–1472.

Zhu, E., Guo, X., & Yin, J. (2016). Walking to singular points of fingerprints. *Pattern Recognition*, *56*, 116–128.

Zhu, Y., Yin, X., Jia, X., & Hu, J. (2017). Latent fingerprint segmentation based on convolutional neural networks. In *Proceedings of Workshop on Information Forensics and Security*, Rennes (pp. 1–6).