Fingerprint Classification and Matching

Anil Jain
Dept. of Computer Science & Engg.
Michigan State University
East Lansing, MI 48824
jain@cse.msu.edu

Sharath Pankanti
Exploratory Computer Vision Grp.
IBM T. J. Watson Research Center
Yorktown Heights, NY10598
sharat@us.ibm.com

1 Introduction

The problem of resolving the identity of a person can be categorized into two fundamentally distinct types of problems with different inherent complexities [1]: (i) verification and (ii) recognition. Verification (authentication) refers to the problem of confirming or denying a person's claimed identity (Am I who I claim I am?). Recognition (Who am I?) refers to the problem of establishing a subject's identity[1]. A reliable personal identification is critical in many daily transactions. For example, access control to physical facilities and computer privileges are becoming increasingly important to prevent their abuse. There is an increasing interest in inexpensive and reliable personal identification in many emerging civilian, commercial, and financial applications.

Typically, a person could be identified based on (i) a person's possession ("something that you possess"), e.g., permit physical access to a building to all persons whose identity could be authenticated by possession of a key; (ii) person's knowledge of a piece of information ("something that you know"), e.g., permit login access to a system to a person who knows the user-id and a password associated with it. Another approach to positive identification is based on identifying physical characteristics of the person. The characteristics could be either a person's physiological traits, e.g., fingerprints, hand geometry, etc. or her behavioral characteristics, e.g., voice and

[1] Often, recognition is also referred to as identification.

signature. This method of identification of a person based on his/her physiological/behavioral characteristics is called *biometrics*. Since the biological characteristics can not be forgotten (like passwords) and can not be easily shared or misplaced (like keys), they are generally considered to be a more reliable approach to solving the personal identification problem.

2 Emerging Applications

Accurate identification of a person could deter crime and fraud, streamline business processes, and save critical resources. Here are a few mind boggling numbers: about one billion dollars in welfare benefits in the United States are annually claimed by "double dipping" welfare recipients with fraudulent multiple identities [33]. MasterCard estimates the credit card fraud at $450 million per annum which includes charges made on lost and stolen credit cards: unobtrusive positive personal identification of the legitimate ownership of a credit card at the point of sale would greatly reduce the credit card fraud; about 1 billion dollars worth of cellular telephone calls are made by the cellular bandwidth thieves – many of which are made from stolen PINS and/or cellular telephones. Again, an identification of the legitimate ownership of the cellular telephones would prevent cellular telephone thieves from stealing the bandwidth. A reliable method of authenticating legitimate owner of an ATM card would greatly reduce ATM related fraud worth approximately $3 billion annually [6]. A positive method of identifying the rightful check payee would also reduce billions of dollars that are misappropriated through fraudulent encashment of checks each year. A method of positive authentication of each system login would eliminate illegal break-ins into traditionally secure (even federal government) computers. The United States Immigration and Naturalization service stipulates that it could each day detect/deter about 3,000 illegal immigrants crossing the Mexican border without delaying legitimate persons entering the United States if it had a quick way of establishing positive personal identification.

High speed computer networks offer interesting opportunities for electronic commerce and electronic purse applications. Accurate authentication of identities over networks is expected to become one of the important application of biometric-based authentication.

Miniaturization and mass-scale production of relatively inexpensive biometric sensors (e.g., solid state fingerprint sensors) will facilitate the use of biometric-based authentication in asset protection.

3 Fingerprint as a Biometric

A smoothly flowing pattern formed by alternating crests (ridges) and troughs (valleys) on the palmar aspect of hand is called a palmprint. Formation of a palmprint depends on the initial conditions of the embryonic mesoderm from which they develop. The pattern on pulp of each terminal phalanx is considered as an individual pattern and is commonly referred to as a *fingerprint* (see, Figure 1). A fingerprint is believed to be unique to each person (and each finger)[2]. Fingerprints of even identical twins are different.

Fingerprints are one of the most mature biometric technologies and are considered legitimate proofs of evidence in courts of law all over the world. Fingerprints are, therefore, used in forensic divisions worldwide for criminal investigations. More recently, an increasing number of civilian and commercial applications are either using or actively considering to use fingerprint-based identification because of a better understanding of fingerprints as well as demonstrated matching performance than any other existing biometric technology.

4 History of Fingerprints

Humans have used fingerprints for personal identification for a very long time [23]. Modern fingerprint matching techniques were initiated in the late 16th century [7]. Henry Fauld, in 1880, first scientifically suggested the individuality and uniqueness of fingerprints. At the same time, Herschel asserted that he had practiced fingerprint identification for about 20 years [23]. This discovery established the foundation of modern fingerprint identification. In the late $19th$ century, Sir Francis Galton conducted an extensive study of fingerprints [23]. He introduced the minutiae features for single fingerprint classification in 1888. The discovery of uniqueness of fingerprints caused an immediate decline in the prevalent use of anthropometric methods of identification and led to the adoption of fingerprints as a more efficient method of identification [29]. An important advance in fingerprint identification was made in 1899 by Edward Henry, who (actually his two assistants from India) established the famous “Henry system” of fingerprint classification [23, 7]: an elaborate method of indexing fingerprints very much tuned to facilitating the human experts

[2] There is some anecdotal evidence that a fingerprint expert once found two (possibly latent) fingerprints belonging to two distinct individuals having 10 identical minutiae.

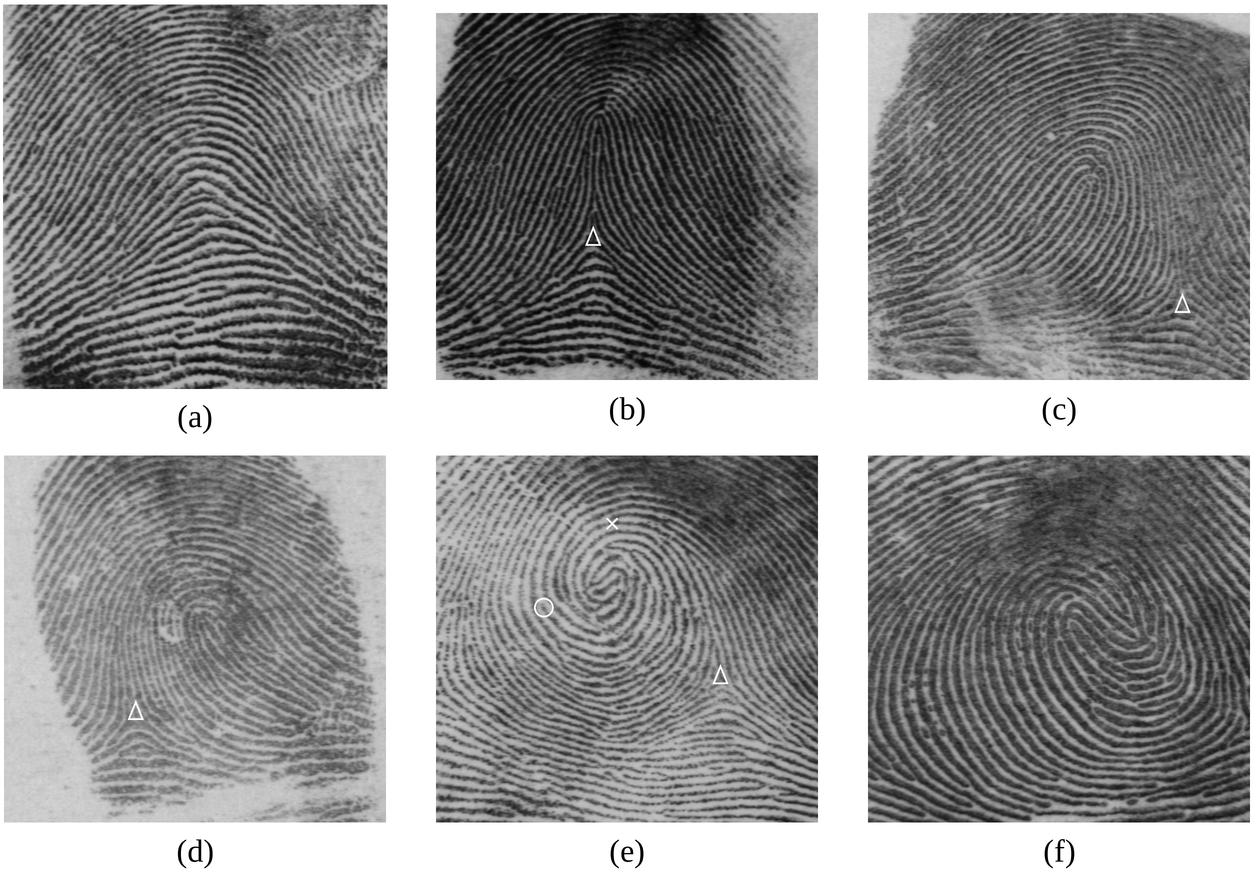


Figure 1: Fingerprints and a fingerprint classification schema involving six categories: (a) arch, (b) tented arch, (c) right loop, (d) left loop, (e) whorl, and (f) twin loop. Critical points in a fingerprint, called core and delta, are marked as squares and triangles. Note that an arch does not have a delta or a core. One of the two deltas in (e) and both the deltas in (f) are not imaged. A sample minutiae ridge ending (○) and ridge bifurcation (×) is illustrated in (e). Each image is 512 × 512 with 256 grey levels and is scanned at 512 *dpi* resolution. All features points were manually extracted by one of the authors.

performing (manual) fingerprint identification. In the early $20th$ century, fingerprint identification was formally accepted as a valid personal identification method by law enforcement agencies and became a standard procedure in forensics [23]. Fingerprint identification agencies were setup worldwide and criminal fingerprint databases were established [23]. With the advent of livescan fingerprinting and availability of cheap fingerprint sensors, fingerprints are increasingly used in government and commercial applications for positive person identification.

5 System Architecture

The architecture of a fingerprint-based automatic identity authentication system is shown in Figure 2. It consists of four components: (i) user interface, (ii) system database, (iii) enrollment module, and (iv) authentication module. The user interface provides mechanisms for a user to indicate his/her identity and input his/her fingerprints into the system. The system database consists of a collection of records, each of which corresponds to an authorized person that has access to the system. Each record contains the following fields which are used for authentication purpose: (i) user name of the person, (ii) minutiae templates of the person's fingerprint, and (iii) other information (e.g., specific user privileges).

The task of enrollment module is to enroll persons and their fingerprints into the system database. When the fingerprint images and the user name of a person to be enrolled are fed to the enrollment module, a minutiae extraction algorithm is first applied to the fingerprint images and the minutiae patterns are extracted. A quality checking algorithm is used to ensure that the records in the system database only consist of fingerprints of good quality, in which a significant number (default value is 25) of genuine minutiae may be detected. If a fingerprint image is of poor quality, it is enhanced to improve the clarity of ridge/valley structures and mask out all the regions that cannot be reliably recovered. The enhanced fingerprint image is fed to the minutiae extractor again.

The task of authentication module is to authenticate the identity of the person who intends to access the system. The person to be authenticated indicates his/her identity and places his/her finger on the fingerprint scanner; a digital image of his/her fingerprint is captured; minutiae pattern is extracted from the captured fingerprint image and fed to a matching algorithm which matches it against the person's minutiae templates stored in the system database to establish the identity.

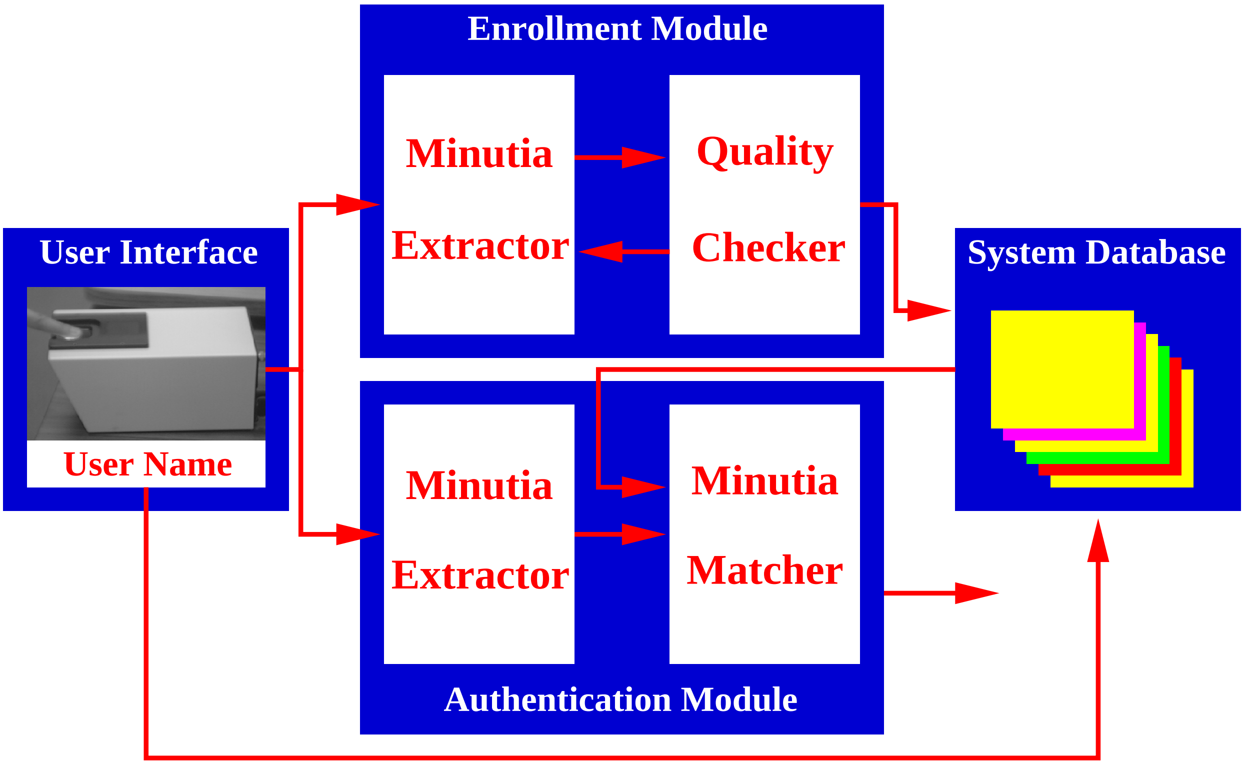
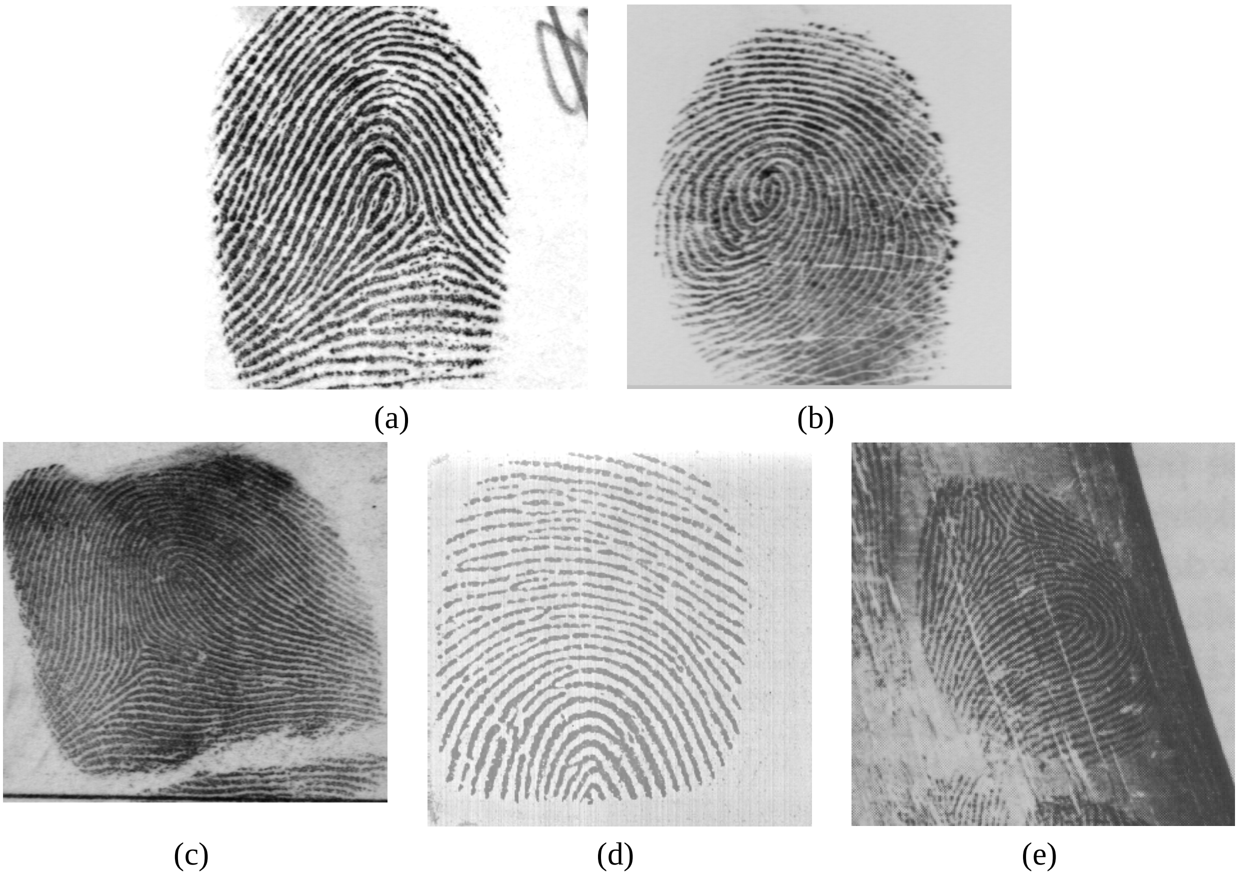


Figure 2: Architecture of an automatic identity authentication system.

6 Fingerprint Sensing

There are two primary methods of capturing a fingerprint image: inked (off-line) and live scan (ink-less) (see Figure 3). An inked fingerprint image is typically acquired in the following way: a trained professional[3] obtains an impression of an inked finger on a paper and the impression is then scanned using a flat bed document scanner. The live scan fingerprint is a collective term for a fingerprint image directly obtained from the finger without the intermediate step of getting an impression on a paper. Acquisition of inked fingerprints is cumbersome; in the context of an identity authentication system, it is both infeasible and socially unacceptable. The most popular technology to obtain a live-scan fingerprint image is based on optical frustrated total internal reflection (FTIR) concept [22]. When a finger is placed on one side of a glass platen (prism), ridges of the finger are

[3] Possibly, for reasons of expediency, MasterCard sends fingerprint kits to their credit card customers. The kits are used by the customers themselves to create an inked fingerprint impression to be used for enrollment.



(a) (b)

(c) (d) (e)

Figure 3: Fingerprint sensing: (a) An inked fingerprint image could be captured from the inked impression of a finger; (b) a livescan fingerprint is directly imaged from a live finger based on optical total internal reflection principle: the light scatters where finger (e.g., ridges) touch the glass prism and light reflects where finger (e.g., valleys) does not touch the glass prism. (c) rolled fingerprints are images depicting nail-to-nail area of a finger (d) fingerprints captured using solid state sensors show a smaller area of finger than a typical fingerprint dab captured using optical scanners. (e) a latent fingerprint refers to partial print typically lifted from a scene of crime.

in contact with the platen, while the valleys of the finger are not in contact with the platen. The rest of the imaging system essentially consists of an assembly of an LED light source and a CCD placed on the other side of the glass platen. The laser light source illuminates the glass at a certain angle and the camera is placed such that it can capture the laser light reflected from the glass. The light incidenting on the platen at the glass surface touched by the ridges is randomly scattered while the light incidenting at the glass surface corresponding to valleys suffers total internal reflection. Consequently, portions of the image formed on the imaging plane of the CCD corresponding to ridges is dark and those corresponding to valleys is bright. More recently, capacitance-based solid state live-scan fingerprint sensors are gaining popularity since they are very small in size and hold promise of becoming inexpensive in the near future. A capacitance-based fingerprint sensor essentially consists of an array of electrodes. The fingerprint skin acts as the other electrode, thereby, forming a miniature capacitor. The capacitance due to the ridges is higher than those formed by valleys. This differential capacitance is the basis of operation of a capacitance-based solid state sensor [34].

7 Fingerprint Representation

Fingerprint representations are of two types: local and global. Major representations of the local information in fingerprints are based on the entire image, finger ridges, pores on the ridges, or salient features derived from the ridges. Representations predominantly based on ridge endings or bifurcations (collectively known as minutiae (see Figure 4)) are the most common, primarily due to the following reasons: (i) minutiae capture much of the individual information, (ii) minutiae-based representations are storage efficient, and (iii) minutiae detection is relatively robust to various sources of fingerprint degradation. Typically, minutiae-based representations rely on locations of the minutiae and the directions of ridges at the minutiae location. Fingerprint classification identifies the typical global representations of fingerprints and is the topic of Section 10. Some global representations include information about locations of critical points (e.g., core and delta) in a fingerprint.

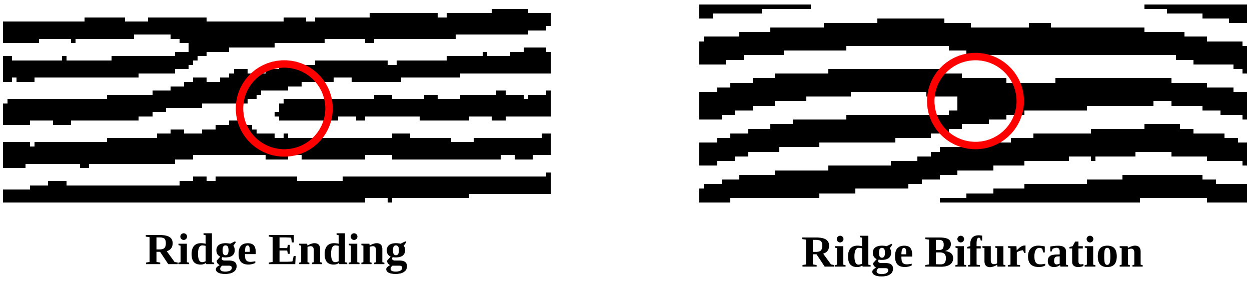


Figure 4: Ridge ending and ridge bifurcation.

8 Feature Extraction

A feature extractor finds the ridge endings and ridge bifurcations from the input fingerprint images. If ridges can be perfectly located in an input fingerprint image, then minutiae extraction is just a trivial task of extracting singular points in a thinned ridge map. However, in practice, it is not always possible to obtain a perfect ridge map. The performance of currently available minutiae extraction algorithms depends heavily on the quality of the input fingerprint images. Due to a number of factors (aberrant formations of epidermal ridges of fingerprints, postnatal marks, occupational marks, problems with acquisition devices, *etc.*), fingerprint images may not always have well-defined ridge structures.

A reliable minutiae extraction algorithm is critical to the performance of an automatic identity authentication system using fingerprints. The overall flowchart of a typical algorithm [28, 18] is depicted in Figure 6. It mainly consists of three components: (i) Orientation field estimation, (ii) ridge extraction, and (iii) minutiae extraction and postprocessing.

1. **Orientation Estimation** The orientation field of a fingerprint image represents the directionality of ridges in the fingerprint image. It plays a very important role in fingerprint image analysis. A number of methods have been proposed to estimate the orientation field of fingerprint images [22]. Fingerprint image is typically divided into a number of non-overlapping blocks (e.g., 32 $\times$ 32 pixels) and an orientation representative of the ridges in the block is assigned to the block based on an analysis of grayscale gradients in the block. The block orientation could be determined from the pixel gradient orientations based on, say, averaging [22], voting [25], or optimization [28]. We have summarized orientation estimation algorithm in Figure 5.

2. **Segmentation** It is important to localize the portions of fingerprint image depicting the fin-

(a) *Divide the input fingerprint image into blocks of size* $W \times W$.

(b) *Compute the gradients* G_x *and* G_y *at each pixel in each block [4].*

(c) *Estimate the local orientation at each pixel* (i, j) *using the following equations [28]:*

$$\begin{aligned} V_x(i,j) &= \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} 2G_x(u,v)G_y(u,v), && (1)\\ V_y(i,j) &= \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} (G_x^2(u,v) - G_y^2(u,v)), && (2)\\ \theta(i,j) &= \frac{1}{2}tan^{-1}(\frac{V_x(i,j)}{V_y(i,j)}), && (3) \end{aligned}$$

where W *is the size of the local window;* G_x *and* G_y *are the gradient magnitudes in* x *and* y *directions, respectively.*

(d) *Compute the* consistency level *of the orientation field in the local neighborhood of a block* (i, j) *with the following formula:*

$$\begin{aligned} C(i,j) &= \frac{1}{N}\sqrt{\sum_{(i',j')\in D} |\theta(i',j') - \theta(i,j)|^2}, && (4)\\ |\theta' - \theta| &= \begin{cases} d & \textit{if}\,(d = (\theta' - \theta + 360)\ \textit{mod}\ 360) < 180, \\ d - 180 & \textit{otherwise}, \end{cases} && (5) \end{aligned}$$

where D *represents the local neighborhood around the block* (i, j) *(in our system, the size of D is* 5×5*);* N *is the number of blocks within* D*;* $\theta(i', j')$ *and* $\theta(i, j)$ *are local ridge orientations at blocks* (i', j') *and* (i, j)*, respectively.*

(e) *If the* consistency level *(Eq.(5)) is above a certain threshold* T_c*, then the local orientations around this region are re-estimated at a lower resolution level until* $C(i, j)$ *is below a certain level.*

Figure 5: Hierarchical orientation field estimation algorithm.

ger (foreground). The simplest approaches segment the foreground by global or adaptive thresholding. A novel and reliable approach to segmentation by Ratha et al. [28] exploits the fact that there is significant difference in the magnitudes of variance in the graylevels along and across the flow of a fingerprint ridge. Typically, block size for variance computation spans 1-2 inter-ridge distance.

3. **Ridge Detection** The approaches to ridge detection use either simple or adaptive thresholding. These approaches may not work for noisy and low contrast portions of the image. An important property of the ridges in a fingerprint image is that the gray level values on ridges attain their local maxima along a direction normal to the local ridge orientation [28, 18]. Pixels can be identified to be ridge pixels based on this property. The extracted ridges may be thinned/cleaned using standard thinning [26] and connected component algorithms [27].

4. **Minutiae Detection** Once the thinned ridge map is available, the ridge pixels with three ridge pixel neighbors are identified as ridge bifurcations and those with one ridge pixel neighbor identified as ridge endings. However, all the minutia thus detected are not genuine due to image processing artifacts and the noise in the fingerprint image.

5. **Postprocessing** In this stage, typically, genuine minutiae are gleaned from the extracted minutiae using a number of heuristics. For instance, too many minutiae in a small neighborhood may indicate noise and they could be discarded. Very close ridge endings oriented anti-parallel to each other may indicate spurious minutia generated by a break in the ridge due either to poor contrast or a cut in the finger. Two very closely located bifurcations sharing a common short ridge often suggest extraneous minutia generated by bridging of adjacent ridges as a result of dirt or image processing artifacts.

9 Fingerprint Enhancement

The performance of a fingerprint image matching algorithm relies critically on the quality of the input fingerprint images. In practice, a significant percentage of acquired fingerprint images (approximately 10% according to our experience) is of poor quality. The ridge structures in poor-quality

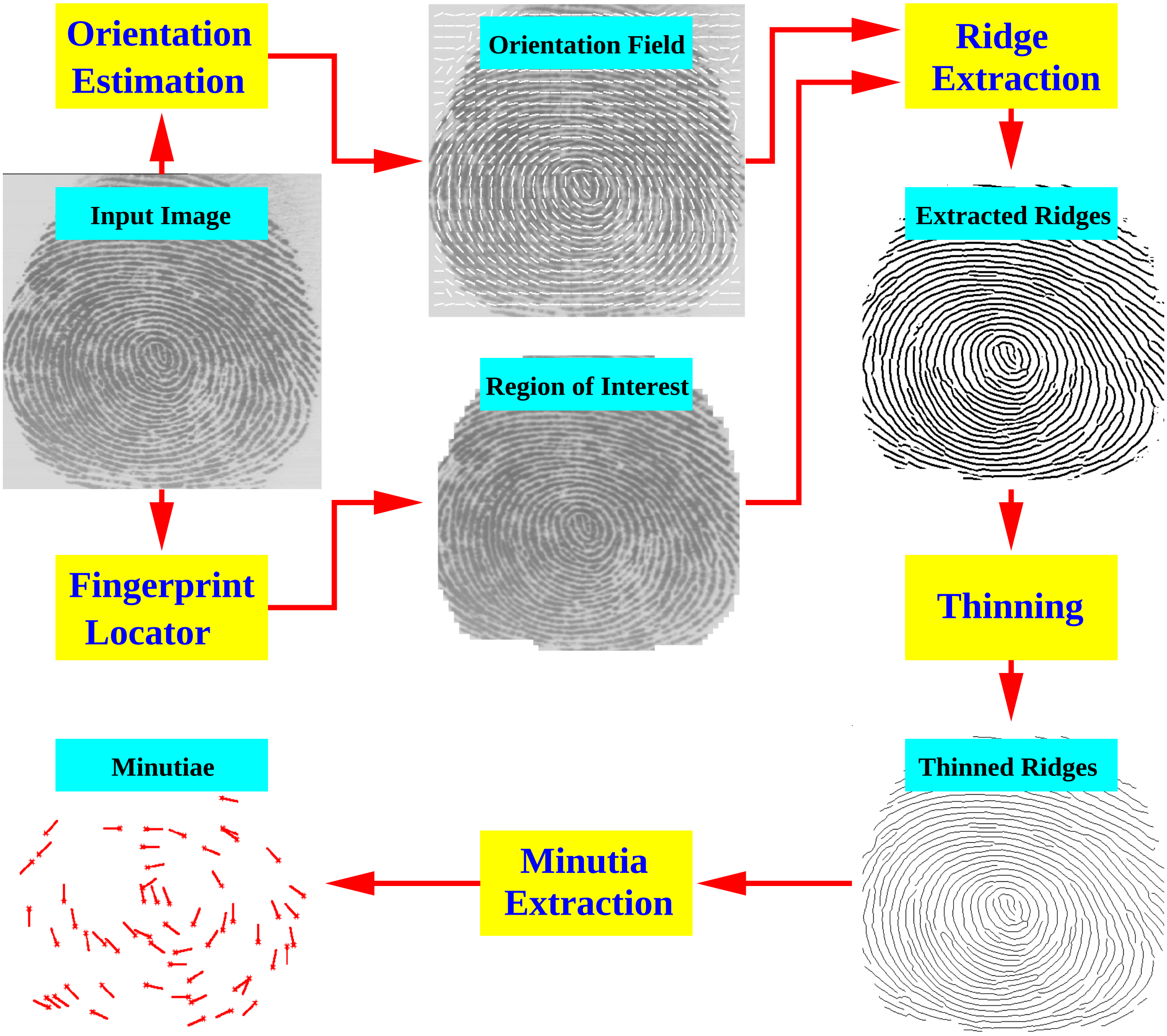


Figure 6: Flowchart of the minutiae extraction algorithm [18].

fingerprint images are not always well-defined and hence they can not be correctly detected. This leads to the following problems: (i) a significant number of spurious minutiae may be created, (ii) a large percentage of genuine minutiae may be ignored, and (iii) large errors in minutiae localization (position and orientation) may be introduced. In order to ensure that the performance of the minutiae extraction algorithm will be robust with respect to the quality of fingerprint images, an enhancement algorithm which can improve the clarity of the ridge structures is necessary.

Typically, fingerprint enhancement approaches [9, 20, 14, 5] employ frequency domain techniques [20, 10, 9] and are computationally demanding. In a small local neighborhood, the ridges and furrows approximately form a two-dimensional sinusoidal wave along the direction orthogonal to local ridge orientation. Thus, the ridges and furrows in a small local neighborhood have well-defined local frequency and local orientation properties. The common approaches employ bandpass filters which models the frequency domain characteristics of a good quality fingerprint image. The poor quality fingerprint image is processed using the filter to block the extraneous *noise* and pass the fingerprint *signal*. Some methods may estimate the orientation and/or frequency of ridge in each block in the fingerprint image and adaptively tune the filter characteristics to match the ridge characteristics.

One typical variation of this theme segments the image into non-overlapping square blocks of widths larger than the average inter-ridge distance. Using a bank of directional bandpass filters, each filter is matched to a predetermined model of generic fingerprint ridges flowing in a certain direction; the filter generating a strong response indicates the dominant direction of the ridge flow in the finger in the given block. The resulting orientation information is more accurate, leading to more reliable features. A single block direction can never truly represent the directions of the ridges in the block and may consequently introduce filter artifacts.

For instance, one common directional filter used for fingerprint enhancement is a Gabor filter [17]. Gabor filters have both frequency-selective and orientation-selective properties and have optimal joint resolution in both spatial and frequency domains. The even-symmetric Gabor filter has the general form [17]

$$h(x,y) = exp\left\{-\frac{1}{2}\left[\frac{x^2}{\delta_x^2}+\frac{y^2}{\delta_y^2}\right]\right\}\cos(2\pi u_0 x), \tag{6}$$

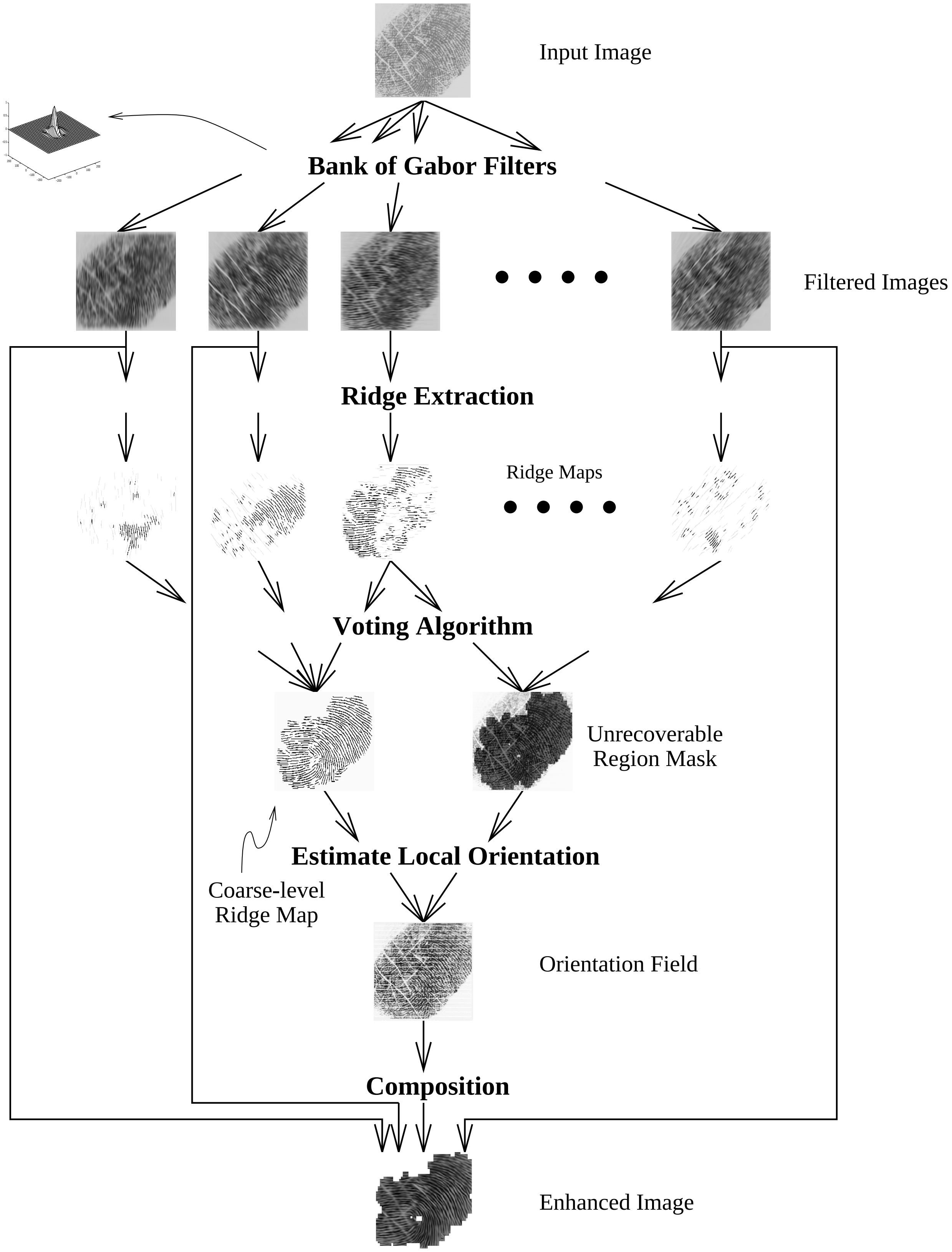
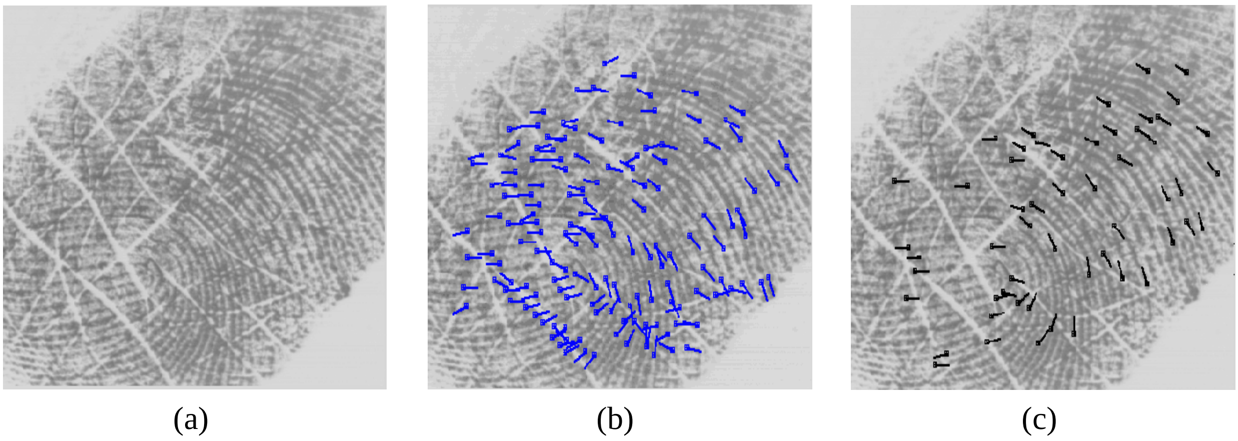
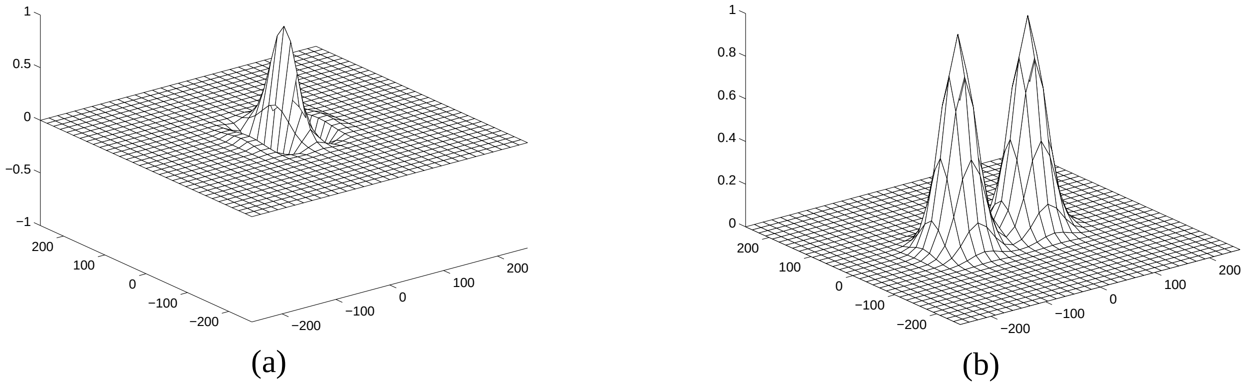


Figure 7: Fingerprint Enhancement Algorithm [11].



(a) (b) (c)

Figure 8: Fingerprint Enhancement Results: (a) a poor quality fingerprint; (b) minutia extracted without image enhancement; and (c) minutiae extracted after image enhancement [11].



(a) (b)

Figure 9: An even-symmetric Gabor filter: (a) Gabor filter tuned to 60 cycles/width and 0^o orientation; (b) corresponding MTF.

where u_0 is the frequency of a sinusoidal plane wave along the x-axis, and δ_x and δ_y are the space constants of the Gaussian envelope along x and y axes, respectively. Gabor filters with arbitrary orientation can be obtained via a rotation of the $x - y$ coordinate system. The modulation transfer function (MTF) of Gabor filter can be represented as

$$H(u,v) = 2\pi\delta_x\delta_y \left(exp\left\{-\frac{1}{2}\left[\frac{(u-u_0)^2}{\delta_u^2} + \frac{v^2}{\delta_v^2}\right]\right\} + exp\left\{-\frac{1}{2}\left[\frac{(u-u_0)^2}{\delta_u^2} + \frac{v^2}{\delta_v^2}\right]\right\}\right), \quad (7)$$

where δ_u = $1/2\pi\delta_x$ and δ_v = $1/2\pi\delta_y$. Figure 9 shows an even-symmetric Gabor filter and its MTF. Typically, in a 500 dpi, 512 × 512 fingerprint image, a Gabor filter with u_0 = 60 cycles per image width (height), the radial bandwidth of 2.5 octaves, and orientation θ models the fingerprint ridges flowing in the direction $\theta + \pi/2$.

We summarize a novel approach to fingerprint enhancement proposed by Hong et al. [11] (see Figure 7). It decomposes the given fingerprint image into several component images using a bank of directional Gabor bandpass filters and extracts ridges from each of the filtered bandpass images using a typical feature extraction algorithm [18]. By integrating information from the sets of ridges extracted from filtered images, the enhancement algorithm infers the region of fingerprint where there is sufficient information to be considered for enhancement (recoverable region) and estimates a coarse-level ridge map for the recoverable region. The information integration is based on the observation that genuine ridges in a region evoke a strong response in the feature images extracted from the filters oriented in the direction parallel to the ridge direction in that region and at most a weak response in feature images extracted from the filters oriented in the direction orthogonal to the ridge direction in that region. The coarse ridge map thus generated consists of the ridges extracted from each filtered image which are mutually consistent and portions of the image where the ridge information is consistent across the filtered images constitute *recoverable* region. The orientation field estimated from the coarse ridge map (see Section 1) is more reliable than the orientation estimation from the input fingerprint image.

After the orientation field is obtained, the fingerprint image can then be adaptively enhanced by using the local orientation information. Let $f_i(x, y)$ (i = 0, 1, 2, 3, 4, 5, 6, 7) denote the grey level value at pixel (x, y) of the filtered image corresponding to the orientation θ_i, $\theta_i = i * 22.5^o$. The grey level value at pixel (x, y) of the enhanced image can be interpolated according to the

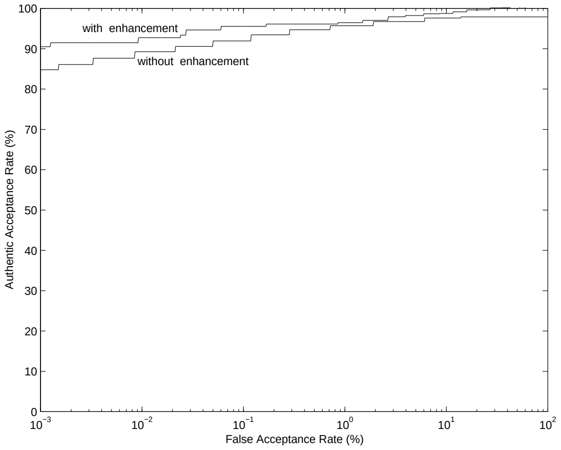


Figure 10: Performance of Fingerprint Enhancement Algorithm.

following formula:

$$f_{enh}(x,y) \quad = \quad a(x,y)f_{p(x,y)}(x,y) + (1-a(x,y))f_{q(x,y)}(x,y), \tag{8}$$

where $p(x,y) = \lfloor\frac{\theta(x,y)}{22.5}\rfloor$, $q(x,y) = \lceil\frac{\theta(x,y)}{22.5}\rceil \bmod 8$, $a(x,y) = \frac{\theta(x,y)-p(x,y)}{22.5}$, and $\theta(x,y)$ represents the value of local orientation field at pixel (x,y). The major reason that we interpolate the enhanced image directly from the limited number of filtered images is that the filtered images are already available and the above interpolation is computationally efficient.

An example illustrating the results of minutiae extraction algorithm on a noisy input image and its enhanced counterpart is shown in Figure 8. The improvement in performance due to image enhancement was evaluated using fingerprint matcher described in Section 11. Figure 10 shows improvement in accuracy of the matcher with and without image enhancement on the MSU database consisting of 700 fingerprint images of 70 individuals (10 fingerprints per finger per individual).

10 Fingerprint Classification

The fingerprints have been traditionally classified into categories based on information in the global patterns of ridges. In large scale fingerprint identification systems, elaborate methods of manual fingerprint classification systems were developed to index individuals into bins based on classification of their fingerprints; these methods of binning eliminate the need to match an input fingerprint(s) to the entire fingerprint database in identification applications and significantly reduce the computing requirements [8, 19].

Efforts in automatic fingerprint classification have been exclusively directed at replicating the manual fingerprint classification system. Figure 1 shows one prevalent manual fingerprint classification scheme that has been the focus of many automatic fingerprint classification efforts. It is important to note that the distribution of fingers into the six classes (shown in Figure 1) is highly skewed. A fingerprint classification system should be invariant to rotation, translation, and elastic distortion of the frictional skin. In addition, often a significant part of the finger may not be imaged (e.g., dabs frequently miss deltas) and the classification methods requiring information from the entire fingerprint may be too restrictive for many applications.

A number of approaches to fingerprint classification have been developed. Some of the earliest approaches did not make use of the rich information in the ridge structures and exclusively depended on the orientation field information. Although fingerprint landmarks provide very effective fingerprint class clues, methods relying on the fingerprint landmarks alone may not be very successful due to lack of availability of such information in many fingerprint images and due to the difficulty in extracting the landmark information from the noisy fingerprint images. As a result, the most successful approaches need to (i) supplement the orientation field information with ridge information; (ii) use fingerprint landmark information when available but devise alternative schemes when such information cannot be extracted from the input fingerprint images; and (iii) use reliable structural/syntactic pattern recognition methods in addition to statistical methods.

We summarize a method of classification [12] which takes into consideration the above mentioned design criteria that has been tested on a large database of realistic fingerprints to classify fingers into five major categories: right loop, left loop, arch, tented arch, and whorl[4].

The orientation field determined from the input image may not be very accurate and the extracted ridges may contain many artifacts and, therefore, cannot be directly used for fingerprint classification. A ridge verification stage assesses the reliability of the extracted ridges based upon

[4] Other types of prints, e.g., twin-loop, are not considered here but, in principle, could be lumped into “other” or “reject” category.

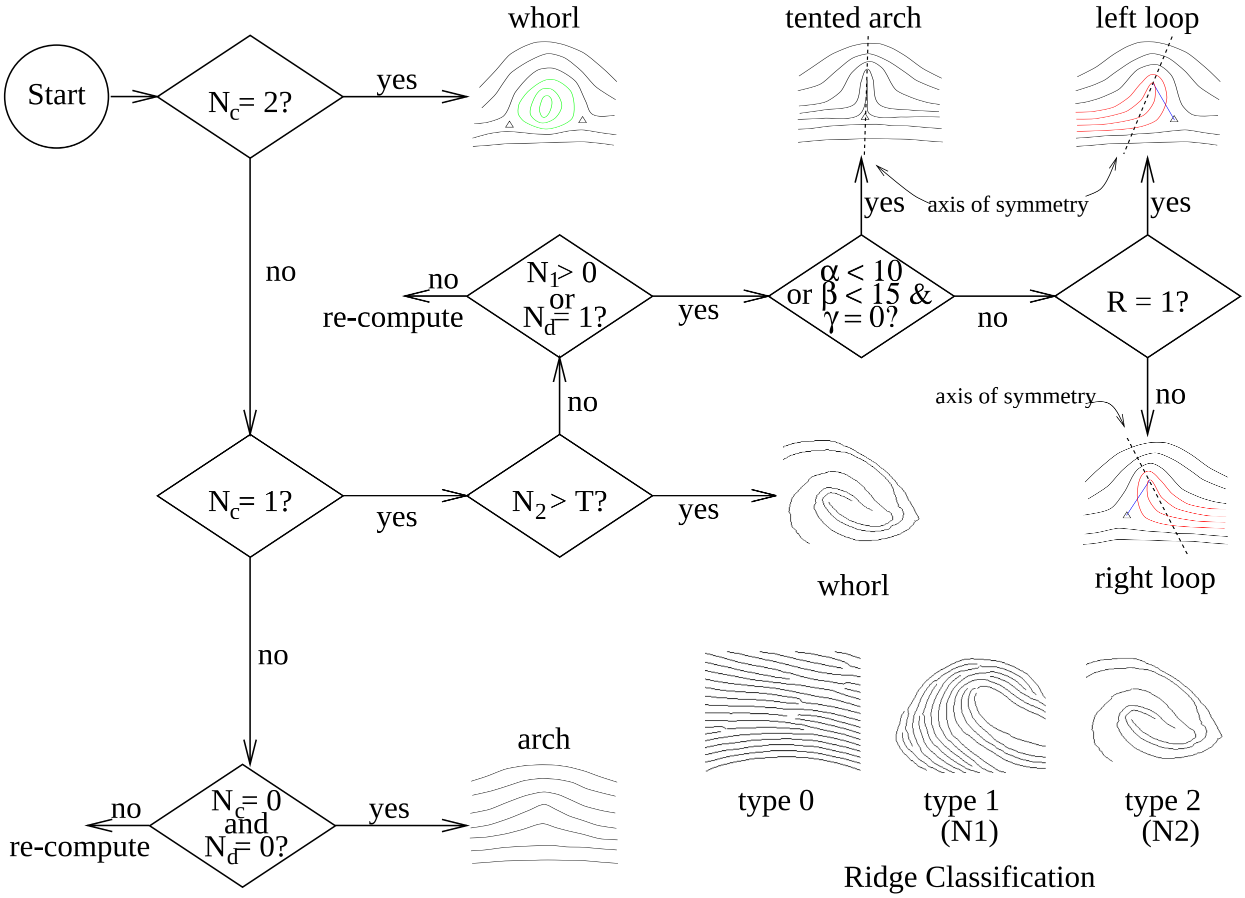


Figure 11: Flowchart of fingerprint classification algorithm. Inset also illustrates ridge classification [12]. The “re-compute” option involves starting the classification algorithm with a different preprocessing (e.g., smoothing) of the image.

the length of each connected ridge segment and its alignment with other adjacent ridges. Parallel adjacent subsegments typically indicate a good quality fingerprint region; the ridge/orientation estimates in these regions are used to refine the estimates in the orientation field/ridge map.

1. Singular Points: The Poincare index [22] on the orientation field is used to determine the number of delta (N_D) and core (N_C) points in the fingerprint. A digital closed curve, Ψ, about 25 pixels long, around each pixel is used to compute the Poincare index as defined below:

   $$Poincare(i,j) = \frac{1}{2\pi}\sum_{k=0}^{N_\Psi}\Delta(k),$$

   where

   $$\begin{aligned}\Delta(k) &= \begin{cases}\delta(k), & \text{if } |\delta(k)| < \pi/2,\\ \pi+\delta(k), & \text{if } \delta(k) \leq -\pi/2,\\ \pi-\delta(k), & otherwise,\end{cases}\\ \delta(k) &= \mathcal{O}'(\Psi_x(i'),\Psi_y(i')) - \mathcal{O}'(\Psi_x(i),\Psi_y(i)),\\ i' &= (i+1)modN_\Psi,\end{aligned}$$

   $\mathcal{O}$ is the orientation field, and $\Psi_x(i)$ and $\Psi_y(i)$ denote coordinates of the i^{th} point on the arc length parameterized closed curve Ψ.

2. Symmetry: The feature extraction stage also estimates an axis locally symmetric to the ridge structures at the core and computes (i) α, angle between the symmetry axis and the line segment joining core and delta, (i) β, average angle difference between the ridge orientation and the orientation of the line segment joining the core and delta, and (iii) γ, the number of ridges crossing the line segment joining core and delta. The relative position, R, of delta with respect to symmetry axis is determined as follows: R = 1 if the delta is on the right side of symmetry axis, R = 0, otherwise.

3. Ridge Structure: The classifier not only uses the orientation information but also utilizes the structural information in the extracted ridges. This feature summarizes the overall nature of

True Class	Assigned Class				
	A	T	L	R	W
A	**885**	13	10	11	0
T	179	**384**	54	14	5
L	31	27	**755**	3	20
R	30	47	3	**717**	16
W	6	1	15	15	**759**

Table 1: Five-class classification results on the NIST-4 database; A-Arch, T-Tented Arch, L-Left Loop, R-Right Loop, W-Whorl.

the ridge flow in the fingerprint. In particular, it classifies each ridge of the fingerprint into three categories:

- Non-recurring ridges: the ridges which do not curve very much.
- Type-1 Recurring ridges: ridges which curve approximately π.
- Type-2 Fully Recurring ridges: ridge which curve by more than π.

The classification algorithm summarized here (see Figure 11) essentially devises a sequence of tests for determining the class of a fingerprint and conducts simpler tests earlier in the decision tree. For instance, two core points are typically detected for a whorl (see Figure 11) which is an easier condition to verify than detecting the number of Type-2 recurring ridges. Another highlight of the algorithm is that if does not detect the salient characteristics of any category from features detected in a fingerprint; it recomputes the features with a different pre-processing method. For instance, in the current implementation, the differential pre-processing consists of a different method/scale of smoothing. As can be observed from the flowchart that the algorithm detects (i) whorls based upon detection of either two core points or a sufficient number of Type-2 recurring ridges; (ii) arch based upon the inability to detect either delta or core points; (iii) left (right) loops based on the characteristic tilt of the symmetric axis, detection of a core point, and detection of either a delta point or a sufficient number of Type-1 recurring curves; and (iv) tented arch based on relatively upright symmetric axis, detection of a core point, and detection of either a delta point or a sufficient number of Type-1 recurring curves.

Table 1 shows the results of the fingerprint classification algorithm on the NIST-4 database which contains 4,000 images (image size is 512×480) taken from 2,000 different fingers, 2 images per finger. Five fingerprint classes are defined: (i) Arch, (ii) Tented arch, (iii) Left Loop, (iv)

Right Loop, and (v) Whorl. Fingerprints in this database are uniformly distributed among these five classes (800 per class). The five-class error rate in classifying these 4,000 fingerprints is 12.5%. The confusion matrix is given in Table 1; numbers shown in bold font are correct classifications. Since a number of fingerprints in the NIST-4 database are labeled as belonging to possibly two different classes, each row of the confusion matrix in Table 1 does not sum up to 800. For the five-class problem, most of the classification errors are due to misclassifying a tented arch as an arch. By combining these two arch categories into a single class, the error rate drops from 12.5% to 7.7%. Besides the tented arch-arch errors, the other errors mainly come from misclassifications between arch/tented arch and loops and due to poor image quality.

11 Fingerprint Matching

Given two (input and template) sets of features originating from two fingerprints, the objective of the feature matching system is to determine whether or not the prints represent the same finger. Fingerprint matching has been approached from several different strategies, like image-based [2], ridge pattern-based, and point (minutiae) pattern-based fingerprint representations. There also exist graph-based schemes [16, 15, 30] for fingerprint matching. Image-based matching may not tolerate large amounts of non-linear distortion in the fingerprint ridge structures. Matchers critically relying on extraction of ridges or their connectivity information may display drastic performance degradation with a deterioration in the quality of the input fingerprints. We, therefore, believe that point pattern matching (minutiae matching) approach facilitates the design of a robust, simple, and fast verification algorithm while maintaining a small template size.

The matching phase typically defines the similarity (distance) metric between two fingerprint representations and determines whether a given pair of representations is captured from the same finger (mated pair) based on whether this quantified (dis)similarity is greater (less) than a certain (predetermined) threshold. The similarity metric is based on the concept of correspondence in minutiae-based matching. A minutiae in the input fingerprint and a minutiae in the template fingerprint are said to be corresponding if they represent the identical minutiae scanned from the same finger.

Before the fingerprint representations could be matched, most minutia-based matchers first transform (*register*) the input and template fingerprint features into a common frame of reference.

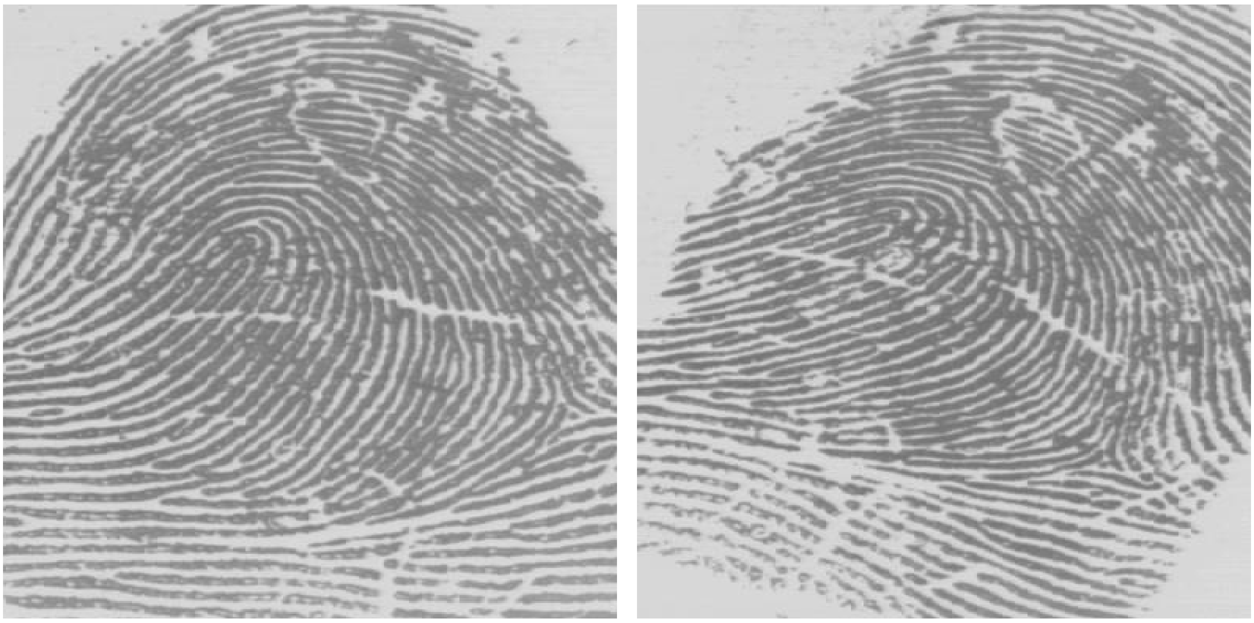


Figure 12: Two different fingerprint impressions of the same finger. In order to know the correspondence between the minutiae of these two fingerprint images, all the minutiae must be precisely localized and the deformation must be recovered.

The registration essentially involves alignment based on rotation/translation and may optionally include scaling. The parameters of alignment are typically estimated either from (i) singular points in the fingerprints, e.g., core and delta locations; (ii) pose clustering based on minutia distribution [28]; or (iii) any other landmark features. For example, Jain et al. [18] use a rotation/translation estimation method based on properties of ridge segment associated with ridge ending minutiae[5].

There are two major challenges involved in determinating the correspondence between two aligned fingerprint representations (see Figure 12): (i) dirt/leftover smudges on the sensing device and the presence of scratches/cuts on the finger either introduce spurious minutiae or obliterate the genuine minutiae; (ii) variations in the area of finger being imaged and its pressure on the sensing device affect the number of genuine minutiae captured and introduce displacements of the minutiae from their "true" locations due to elastic distortion of the fingerprint skin. Consequently, a fingerprint matcher should not only assume that the input fingerprint is a transformed template fingerprint by a similarity transformation (rotation, translation, and scale), but it should also tolerate both spurious minutiae as well as missing genuine minutiae and accommodate perturbations of

[5] The input and template minutiae used for the alignment will be referred to as reference minutiae below.

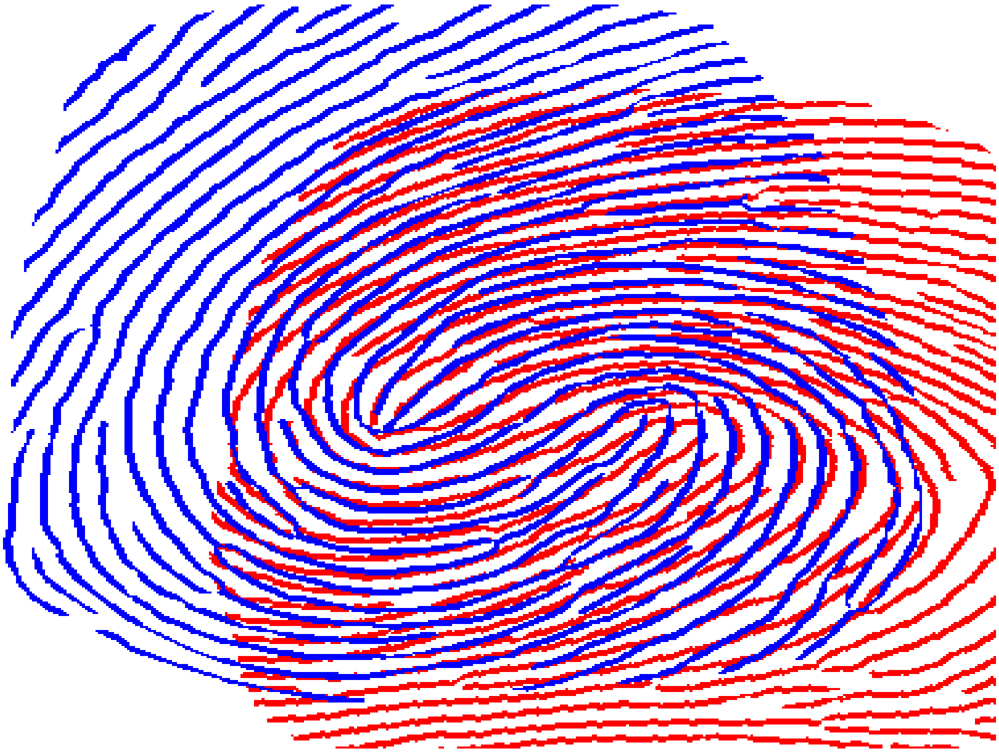


Figure 13: Aligned ridge structures of mated pairs. Note that the best alignment in one part (mid-left) of the image results in a large displacements between the corresponding minutiae in the other regions (bottom right).

minutiae from their true locations. Figure 13 illustrates a typical situation of aligned ridge structures of mated pairs. Note that the best alignment in one part (top left) of the image may result in a large amount of displacements between the corresponding minutiae in other regions (bottom right). In addition, observe that the distortion is non-linear: given the amount of distortions at two arbitrary locations on the finger, it is not possible to predict the distortions at all the intervening points on the line joining the two points.

The adaptive elastic string matching algorithm [18] summarized in this chapter uses three attributes of the aligned minutiae for matching: its distance from the reference minutiae (*radius*), angle subtended to the reference minutiae (*radial angle*), and local direction of the associated ridge (*minutiae direction*). The algorithm initiates the matching by first representing the aligned input (template) minutiae as an input (template) minutiae string. The string representation is obtained by imposing a linear ordering based on radial angles and radii. The resulting input and

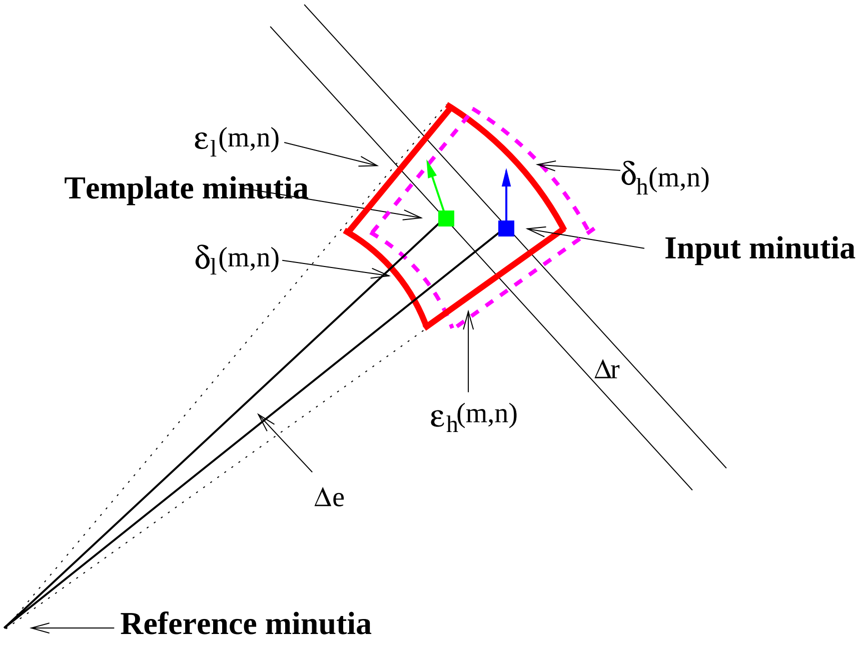


Figure 14: Bounding box and its adjustment.

template minutiae strings are matched using an inexact string matching algorithm to establish the correspondence.

The inexact string matching algorithm essentially transforms (*edits*) the input string to template string and the number of edit operations is considered as a metric of the (dis)similarity between the strings. While permitted edit operators model the impression variations in a representation of a finger (deletion of the genuine minutiae, insertion of spurious minutiae, and perturbation of the minutiae), the penalty associated with each edit operator models the likelihood of that edit. The sum of penalties of all the edits (*edit distance*) defines the similarity between the input and template minutiae strings. Among several possible sets of edits that permit the transformation of the input minutiae string into the reference minutiae string, the string matching algorithm chooses the transform associated with the minimum cost based on dynamic programming.

The algorithm tentatively considers a candidate (aligned) input and a candidate template minutiae in the input and template minutiae string to be a mismatch if their attributes are not within a tolerance window (see Figure 14) and penalizes them for deletion/insertion edit. If the attributes are within the tolerance window, the amount of penalty associated with the tentative match is proportional to the disparity in the values of the attributes in the minutiae. The algorithm accommodates

Threshold Value	False Acceptance Rate (MSU)	False Reject Rate (MSU)	False Acceptance Rate (NIST 9)	False Reject Rate (NIST 9)
7	0.07%	7.1%	0.073%	12.4%
8	0.02%	9.4%	0.023%	14.6%
9	0.01%	12.5%	0.012%	16.9%
10	0	14.3%	0.003%	19.5%

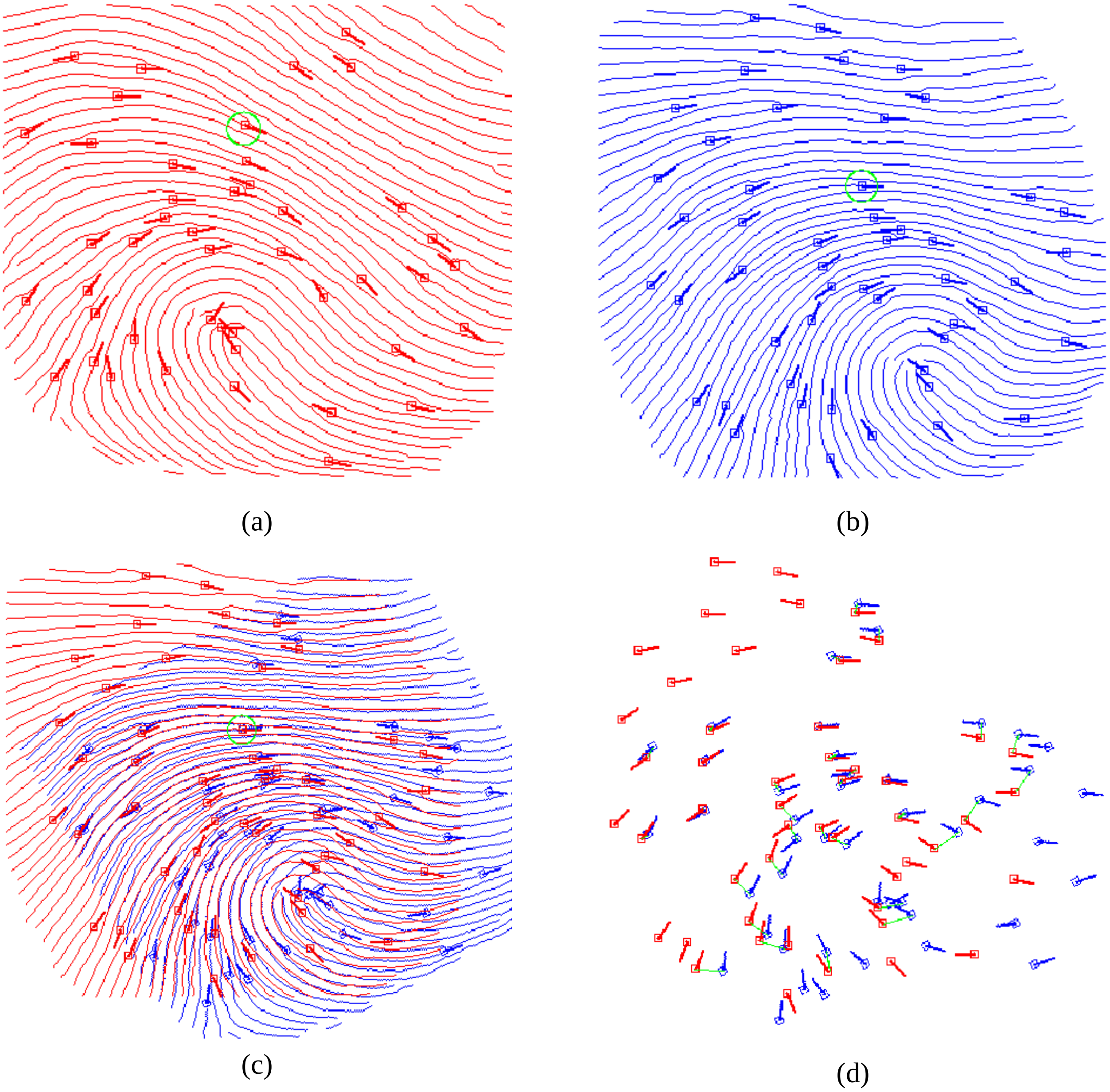
Table 2: False acceptance and false reject rates on two data sets with different threshold values.

for the elastic distortion by adaptively adjusting the parameters of the tolerance window based on the most recent successful tentative match. The tentative matches (and correspondences) are accepted if the edit distance for those correspondences is smaller than any other correspondences.

Figure 15 shows the results of applying the matching algorithm to an input and a template minutiae set pair. The outcome of the matching process is defined by a matching score. Matching score is determined from the number of mated minutia from the correspondences associated with the minimum cost of matching input and template minutiae string. The raw matching score is normalized by the total number of minutia in the input and template fingerprint representations and is used for deciding whether input and template fingerprints are mates. The higher the normalized score, the larger the likelihood that the test and template fingerprints are the scans of the same finger.

The results of performance evaluation of the fingerprint matching algorithm are illustrated in Figure 16 for 1,350 fingerprint images in NIST 9 database [31] and in Figure 10 for 700 images of 70 individuals from the MSU database. Some sample points on the receiver operating characteristics curve are tabulated in Table 2.

In order for an automatic identity authentication system to be acceptable in practice, the response time of the system needs to be within a few seconds. Table 3 shows that our implemented system does meet the practical response time requirement.



(a)

(b)

(c)

(d)

Figure 15: Results of applying the matching algorithm to an input minutiae set and a template; (a) input minutiae set; (b) template minutiae set; (c) alignment result based on the minutiae marked with green circles; (d) matching result where template minutiae and their correspondences are connected by green lines.

Minutiae Extraction (seconds)	Minutiae Matching (seconds)	Total (seconds)
1.1	0.3	1.4

Table 3: Average CPU time for minutiae extraction and matching on a Sun ULTRA 1 workstation.

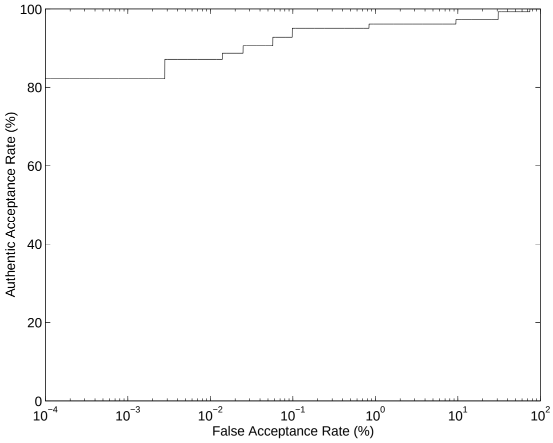


Figure 16: Receiver Operating Characteristic Curve for NIST 9 (CD No. 1).

12 Summary and Future Prospects

With recent advances in fingerprint sensing technology and improvements in the accuracy and matching speed of the fingerprint matching algorithms, automatic personal identification based on fingerprint is becoming an attractive alternative/complement to the traditional methods of identification. We have provided an overview of the fingerprint-based identification and summarized algorithms for fingerprint feature extraction, enhancement, matching, and classification. We have also presented a performance evaluation of these algorithms.

The critical factor for the widespread use of fingerprints is in meeting the performance (e.g., matching speed and accuracy) standards demanded by emerging civilian identification applications. Unlike an identification based on passwords or tokens, performance of the fingerprint-based identification is not perfect. There will be a growing demand for faster and more accurate fingerprint matching algorithms which can (particularly) handle poor quality images. Some of the emerging applications (e.g., fingerprint-based smartcards) will also benefit from a compact representation of a fingerprint. The design of highly reliable, accurate, and foolproof biometrics-based identification systems may warrant effective integration of discriminatory information contained in several different biometrics and/or technologies [13]. The issues involved in integrating fingerprint-based identification with other biometric or non-biometric technologies may constitute an important research topic.

As biometric technology matures, there will be an increasing interaction among the (biometric) market, (biometric) technology, and the (identification) applications. The emerging interaction is expected to be influenced by the added value of the technology, the sensitivities of the population, and the credibility of the service provider. It is too early to predict where, how, and which biometric technology would evolve and be mated with which applications. But it is certain that biometrics-based identification will have a profound influence on the way we conduct our daily business. It is also certain that, as the most mature and well-understood biometric, fingerprints will remain an integral part of the preferred biometric-based identification solutions in the years to come.

References

[1] A. K. Jain, R. Bolle, S. Pankanti (eds), Biometrics: Personal Identification in Networked Society, Kluwer Academic, December 1998.

[2] R. Bahuguna, Fingerprint Verification Using Hologram Matched Filterings, *Proc. Biometric Consortium Eighth Meeting*, San Jose, California, June, 1996.

[3] G. T. Candela, P. J. Grother, C. I. Watson, R. A. Wilkinson, and C. L. Wilson, PCASYS: A Pattern-Level Classification Automation System for Fingerprints, *NIST Tech. Report NISTIR 5647*, August, 1995.

[4] J. Canny, A Computational Approach to Edge Detection, *IEEE Transactions on PAMI*, Vol. 8, No. 6, pp. 679-698, 1986.

[5] L. Coetzee and E. C. Botha, Fingerprint Recognition in Low Quality Images, *Pattern Recognition*, Vol. 26, No. 10, pp. 1441-1460, 1993.

[6] L. Lange and G. Leopold, Digital identification: It's now at our fingertips, *EEtimes at http://techweb.cmp.com/eet/823/, March 24*, vol. 946, 1997.

[7] Federal Bureau of Investigation, The Science of Fingerprints: Classification and Uses, U. S. Government Printing Office, Washington, D. C., 1984.

[8] R. Germain, A Califano, and S. Colville, Fingerprint matching using transformation parameter clustering, *IEEE Computational Science and Engineering*, Vol. 4, No. 4, pp. 42–49, 1997.

[9] L. O'Gorman and J. V. Nickerson, An Approach to Fingerprint Filter Design, *Pattern Recognition*, Vol. 22, No. 1, pp. 29-38, 1989.

[10] L. Hong, A. K. Jain, S. Pankanti, and R. Bolle, Fingerprint Enhancement, *Proc. IEEE Workshop on Applications of Computer Vision*, Sarasota, FL, pp. 202-207, 1996.

[11] L. Hong, Automatic Personal Identification Using Fingerprints, *PhD Thesis*, Michigan State University, 1998.

[12] L. Hong and A. K. Jain, Classification of fingerprint images, *MSU Technical Report*, MSU Technical Report MSUCPS:TR98-18, June 1998.

[13] L. Hong and A. K. Jain, Integrating Faces and Fingerprints, *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 20, No. 12, pp. 1295-1307, December 1998.

[14] D. C. Douglas Hung, Enhancement and Feature Purification of Fingerprint Images, *Pattern Recognition*, Vol. 26, No. 11, pp. 1661-1671, 1993.

[15] A. K. Hrechak and J. A. McHugh, Automated Fingerprint Recognition using Structural Matching, *Pattern Recognition*, Vol. 23, No. 8, 1990.

[16] D. K. Isenor and S. G. Zaky, Fingerprint Identification using Graph Matching , *Pattern Recognition*, Vol. 19, No. 2, 1986.

[17] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," *Pattern Recognition*, Vol. 24, No. 12, pp. 1167-1186, 1991.

[18] A. Jain, L. Hong, S. Pankanti, and R. Bolle, On-line identity-authentication system using fingerprints, *Proceedings of IEEE (Special Issue on Automated Biometrics)*, vol. 85, pp. 1365–1388, September 1997.

[19] A. K. Jain, S. Prabhakar, and L. Hong, A Multichannel Approach to Fingerprint Classification, *Proc. of Indian Conference on Computer Vision, Graphics, and Image Processing (ICVGIP'98)*, New Delhi, India, December 21-23, 1998.

[20] T. Kamei and M. Mizoguchi. Image filter design for fingerprint enhancement. In *Proc. ISCV' 95*, pages 109–114, Coral Gables, FL, 1995.

[21] K. Karu and A. K. Jain, Fingerprint Classification, *Pattern Recognition,* Vol. 29, No. 3, pp. 389-404, 1996.

[22] M. Kawagoe and A. Tojo, Fingerprint Pattern Classification, *Pattern Recognition,* Vol. 17, No. 3, pp. 295-303, 1984.

[23] H. C. Lee and R. E. Gaensslen, *Advances in Fingerprint Technology,* Elsevier, New York, 1991.

[24] D. Maio, D. Maltoni, Direct Gray-Scale Minutiae Detection in Fingerprints, *IEEE Trans. Pattern Anal. Machine Intell.,* Vol. 19, No. 1, pp. 27-40, 1997.

[25] B. M. Mehtre and B. Chatterjee, Segmentation of fingerprint images – A composite method, *Pattern Recognition,* Vol. 22, No. 4, pp. 381–385, 1989.

[26] N.J. Naccache and R. Shinghal, An Investigation into the Skeletonization Approach of Hilditch, Pattern Recognition Journal, Vol. 17, No. 3, pp. 279-284, 1984.

[27] T. Pavlidis, *Algorithms for Graphics and Image Processing,* Computer Science Press, 1982.

[28] N. Ratha, K. Karu, S. Chen and A. K. Jain, A Real-time Matching System for Large Fingerprint Database, *IEEE Trans. on Pattern Anal. Machine Intell.,* Vol. 18, No. 8, pp. 799-813, 1996.

[29] H. T. F. Rhodes, *Alphonse Bertillon: Father of Scientific Detection.* Abelard-Schuman, New York, 1956.

[30] M. K. Sparrow and P. J. Sparrow, A topological approach to the matching of single fingerprints: development of algorithms for use of rolled impressions, *National Bureau of Standards,* Tech. Report, Gaithersburg, MD, May, 1985.

[31] C. I. Watson, *NIST Special Database 9, Mated Fingerprint Card Pairs,* National Institute of Standards and Technology, May 1993.

[32] C. L. Wilson, G. T. Candela and C. I. Watson, Neural-Network Fingerprint Classification, *Journal of Artificial Neural Networks,* Vol. 1, No. 2, pp. 203-228, 1994.

[33] J. D. Woodward, Biometrics: Privacy’s foe or privacy’s friend?, *Proceedings of the IEEE (Special Issue on Automated Biometrics)*, vol. 85, pp. 1480–1492, September 1997.

[34] N. D. Young, G. Harkin, R. M. Bunn, D. J. McCulloch and R. W. Wilks and A. G. Knapp, Novel Fingerprint Scanning Arrays Using Polysilicon TFT’s on Glass and Polymer Substrates, *IEEE Electron Device Letters*, Vol. 18, No. 1, pp. 19–20, Jan. 1997.