An Identity-Authentication System Using Fingerprints

ANIL K. JAIN, FELLOW, IEEE, LIN HONG, SHARATH PANKANTI, ASSOCIATE MEMBER, IEEE, AND RUUD BOLLE, FELLOW, IEEE

Fingerprint verification is an important biometric technique for personal identification. In this paper, we describe the design and implementation of a prototype automatic identity-authentication system that uses fingerprints to authenticate the identity of an individual. We have developed an improved minutiae-extraction algorithm that is faster and more accurate than our earlier algorithm [58]. An alignment-based minutiae-matching algorithm has been proposed. This algorithm is capable of finding the correspondences between input minutiae and the stored template without resorting to exhaustive search and has the ability to compensate adaptively for the nonlinear deformations and inexact transformations between an input and a template. To establish an objective assessment of our system, both the Michigan State University and the National Institute of Standards and Technology NIST 9 fingerprint data bases have been used to estimate the performance numbers. The experimental results reveal that our system can achieve a good performance on these data bases. We also have demonstrated that our system satisfies the response-time requirement. A complete authentication procedure, on average, takes about 1.4 seconds on a Sun ULTRA 1 workstation (it is expected to run as fast or faster on a 200 HMz Pentium [7]).

Keywords*—*Biometrics, dynamic programming, fingerprint identification, matching, minutiae, orientation field, ridge extraction, string matching, verification.

I. INTRODUCTION

There are two types of systems that help automatically establish the identity of a person: 1) authentication (verification) systems and 2) identification systems. In a verification system, a person desired to be identified submits an identity claim to the system, usually via a magnetic stripe card, login name, smart card, etc., and the system either rejects or accepts the submitted claim of identity (Am I who I claim I am?). In an identification system, the system establishes a subject's identity (or fails if the subject is not enrolled in the system data base) without the subject's having to claim an identity (Who am I?). The topic of this paper is a verification system based on fingerprints, and the terms verification, authentication, and identification are used in a loose sense and synonymously.

Accurate automatic personal identification is becoming more and more important to the operation of our increasingly electronically interconnected information society [13], [20], [53]. Traditional automatic personal identification technologies to verify the identity of a person, which use "something that you know," such as a personal identification number (PIN), or "something that you have," such as an identification (ID) card, key, etc., are no longer considered reliable enough to satisfy the security requirements of electronic transactions. All of these techniques suffer from a common problem of inability to differentiate between an authorized person and an impostor who fraudulently acquires the access privilege of the authorized person [53]. Biometrics is a technology that (uniquely) identifies a person based on his physiological or behavioral characteristics. It relies on "something that you are" to make personal identification and therefore can inherently differentiate between an authorized person and a fraudulent impostor [13], [20], [53]. Although biometrics cannot be used to establish an absolute "yes/no" personal identification like some of the traditional technologies, it can be used to achieve a "positive identification" with a very high level of confidence, such as an error rate of 0.001% [53].

A. Overview of Biometrics

Theoretically, any human physiological or behavioral characteristic can be used to make a personal identification as long as it satisfies the following requirements [13]:

1) universality, which means that every person should have the characteristic;
2) uniqueness, which indicates that no two persons should be the same in terms of the characteristic;
3) permanence, which means that the characteristic should be invariant with time;
4) collectability, which indicates that the characteristic can be measured quantitatively.

Manuscript received October 31, 1996; revised April 26, 1997.

A. K. Jain and L. Hong are with the Department of Computer Science, Michigan State University, East Lansing, MI 48824 USA (e-mail: jain@cps.msu.edu; honglin@cps.msu.edu).

S. Pankanti and R. Bolle are with the Exploratory Computer Vision Group, IBM T. J. Watson Research Center, Yorktown Heights, NY 10598 USA (e-mail: sharat@watson.ibm.com; bolle@watson.ibm.com).

Publisher Item Identifier S 0018-9219(97)06635-8.

Table 1 Comparison of Biometric Technologies

Biometrics	Universality	Uniqueness	Permanence	Collectability	Performance	Acceptability	Circumvention
Face	high	low	medium	high	low	high	low
Fingerprint	medium	high	high	medium	high	medium	high
Hand Geometry	medium	medium	medium	high	medium	medium	medium
Hand Vein	medium	medium	medium	medium	medium	medium	high
Iris	high	high	high	medium	high	low	high
Retinal Scan	high	high	medium	low	high	low	high
Signature	low	low	low	high	low	high	low
Voice Print	medium	low	low	medium	low	high	low
F.Thermograms	high	high	low	high	medium	high	high

In practice, there are some other important requirements [13], [53]:

1) performance, which refers to the achievable identification accuracy, the resource requirements to achieve an acceptable identification accuracy, and the working or environmental factors that affect the identification accuracy;
2) acceptability, which indicates to what extent people are willing to accept the biometric system;
3) circumvention, which refers to how easy it is to fool the system by fraudulent techniques.

Biometrics is a rapidly evolving technology that has been widely used in forensics, such as criminal identification and prison security, and has the potential to be widely adopted in a very broad range of civilian applications:

1) banking security, such as electronic fund transfers, ATM security, check cashing, and credit card transactions;
2) physical access control, such as airport access control;
3) information system security, such as access to data bases via login privileges;
4) government benefits distribution, such as welfare disbursement programs [49];
5) customs and immigration, such as the Immigration and Naturalization Service Passenger Accelerated Service System (INSPASS) which permits faster immigration procedures based on hand geometry [35];
6) national ID systems, which provide a unique ID to the citizens and integrate different government services [31];
7) voter and driver registration, providing registration facilities for voters and drivers.

Currently, there are mainly nine different biometric techniques that are either widely used or under investigation, including face, fingerprint, hand geometry, hand vein, iris, retinal pattern, signature, voice print, and facial thermograms [13], [18], [20], [53], [68]. A brief comparison of these nine biometric techniques is provided in Table 1. Although each of these techniques, to a certain extent, satisfies the above requirements and has been used in practical systems [13], [18], [20], [53] or has the potential to become a valid biometric technique [53], not many of them are acceptable (in a court of law) as indisputable evidence of identity. For example, despite the fact that extensive studies have been conducted on automatic face recognition and that a number of face-recognition systems are available [3], [62], [70], it has not yet been proven that 1) face can be used reliably to establish/verify identity and 2) a biometric system that uses only face can achieve an acceptable identification accuracy in a practical environment. Without any other information about the people in Fig. 1, it will be extremely difficult for both a human and a face-recognition system to conclude that the different faces shown in Fig. 1 are disguised versions of the same person. So far, the only legally acceptable, readily automated, and mature biometric technique is the automatic fingerprint-identification technique, which has been used and accepted in forensics since the early 1970's [42]. Although signatures also are legally acceptable biometrics, they rank a distant second to fingerprints due to issues involved with accuracy, forgery, and behavioral variability. Currently, the world market for biometric systems is estimated at approximately $112 million. Automatic fingerprint-identification systems intended mainly for forensic applications account for approximately $100 million. The biometric systems intended for civilian applications are growing rapidly. For example, by the year 1999, the world market for biometric systems used for physical access control alone is expected to expand to $100 million [53].

The biometrics community is slow in establishing benchmarks for biometric systems [20]. Although benchmark results on standard data bases in themselves are useful only to a limited extent and may result in excessive tuning of the

Fig. 1. Multiple personalities: all of the people in this image are the same person. (From *The New York Times Magazine*, Sept. 1, 1996, sect. 6, pp. 48–49. Reproduced with permission of Robert Trachtenberg.)

system parameters to "improve" the system performance,[1] they constitute a good starting point for comparison of the gross performance characteristics of the systems.

No metric is sufficiently adequate to give a reliable and convincing indication of the identification accuracy of a biometric system. A decision made by a biometric system is either a "genuine individual" type of decision or an "impostor" type of decision, which can be represented by two statistical distributions, called genuine distribution and impostor distribution, respectively. For each type of decision, there are two possible decision outcomes, true or false. Therefore, there are a total of four possible outcomes: 1) a genuine individual is accepted, 2) a genuine individual is rejected, 3) an impostor is rejected, and 4) an impostor is accepted. Outcomes 1) and 3) are correct, whereas 2) and 4) are incorrect. In principle, we can use the false (impostor) acceptance rate (FAR), the false (genuine individual) reject rate (FRR), and the equal error rate (EER)[2] to indicate the identification accuracy of a biometric system [18], [19], [53]. In practice, these performance metrics can only be estimated from empirical data, and the estimates of the performance are very data dependent. Therefore, they are meaningful only for a specific data base in a specific test environment. For example, the performance of a biometric system claimed by its manufacturer had an FRR of 0.3% and an FAR of 0.1%. An independent test by the Sandia National Laboratory found that the same system had an FRR of 25% with an unknown FAR [10]. To provide a more reliable assessment of a biometric system, some more descriptive performance measures are necessary. Receiver operating curve (ROC) and d' are the two other commonly used measures. An ROC provides an empirical assessment of the system performance at different operating points, which is more informative than FAR and FRR. The statistical metric d' gives an indication of the separation between the genuine distribution and impostor distribution [19]. It is defined as the difference between the means of the genuine distribution and impostor distribution divided by a conjoint measure of their standard deviations [19]

$$d' = \frac{||M_{\text{impostor}} - M_{\text{genuine}}||}{\sqrt{(\text{SD}^2_{\text{impostor}} + \text{SD}^2_{\text{genuine}})/2}} \tag{1}$$

where M_{genuine}, $\text{SD}_{\text{genuine}}$, M_{impostor}, and $\text{SD}_{\text{impostor}}$ are the means and standard deviations of the genuine distribution and impostor distribution, respectively. Like FAR, FRR, and EER, both ROC and d' also depend heavily on test data and test environments. For such performance metrics to be able to generalize precisely to the entire population of interest, the test data should 1) be large enough to represent the population and 2) contain enough samples from each category of the population [19]. To obtain fair and honest test results, enough samples should be available, and the samples should be representative of the population and adequately represent all the categories (impostor and genuine). Further, irrespective of the performance measure, error bounds that indicate the confidence of the estimates are valuable for understanding the significance of the test results.

B. History of Fingerprints

Fingerprints are graphical flow-like ridges present on human fingers (see Fig. 2). Their formations depend on the initial conditions of the embryonic mesoderm from which they develop. Humans have used fingerprints as a means of identification for a very long time [42]. Modern fingerprint techniques were initiated in the late sixteenth century [25], [53]. In 1684, English plant morphologist N.

[1] Several additional techniques, like data sequestering [51] and third-party benchmarking [9], may also help in obtaining fairer performance results.

[2] Equal error rate is defined as the value where FAR and FRR are equal.

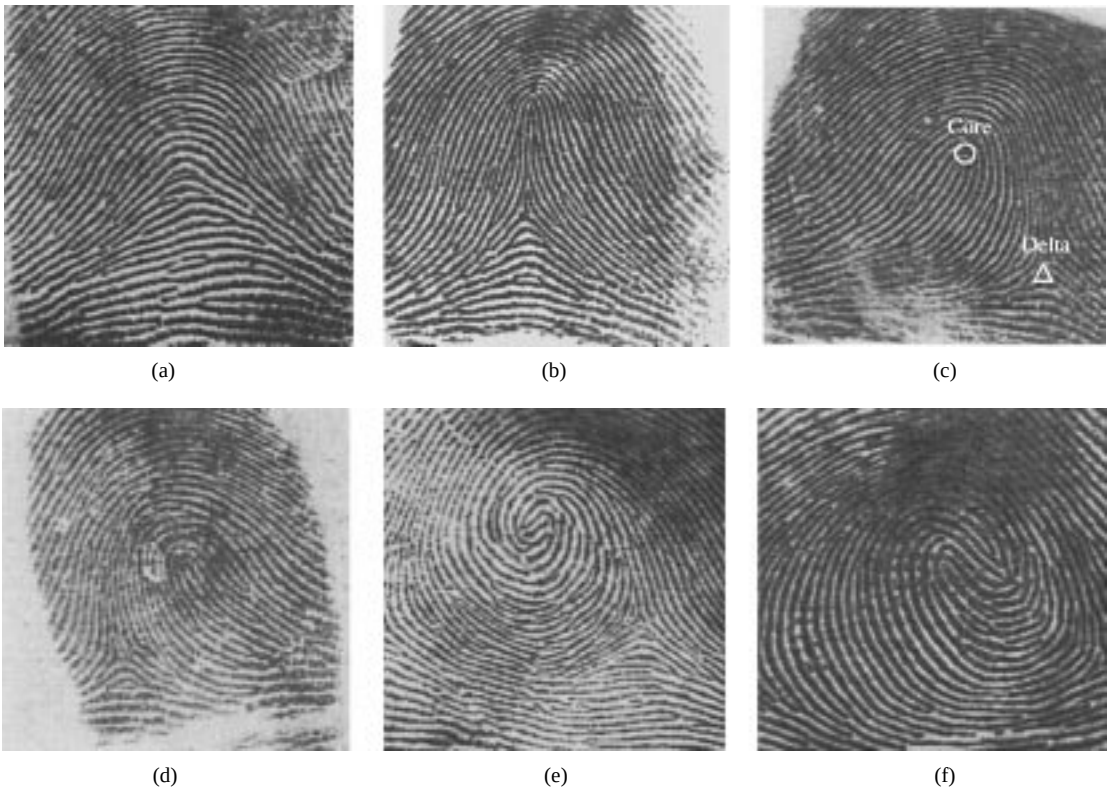


Fig. 2. Fingerprints and a fingerprint classification schema of six categories: (a) arch, (b) tented arch, (c) right loop, (d) left loop, (e) whorl, and (f) twinloop. Critical points in a fingerprint, called core and delta, are marked on (c).

Grew published a paper reporting his systematic study on the ridge, furrow, and pore structure in fingerprints, which is believed to be the first scientific paper on fingerprints [42]. Since then, a number of researchers have invested a huge amount of effort in studying fingerprints. In 1788, a detailed description of the anatomical formations of fingerprints was made by Mayer [16], in which a number of fingerprint ridge characteristics were identified. Starting from 1809, T. Bewick began to use his fingerprint as his trademark, which is believed to be one of the most important contributions in the early scientific study of fingerprint identification [42]. Purkinje proposed the first fingerprint classification scheme in 1823, which classified fingerprints into nine categories according to the ridge configurations [42]. H. 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 approximately 20 years [42]. This discovery established the foundation of modern fingerprint identification. In the late nineteenth century, Sir F. Galton conducted an extensive study of fingerprints [42]. He introduced the minutiae features for single fingerprint classification in 1888. An important advance in fingerprint identification was made in 1899 by E. Henry, who (actually his two assistants from India) established the famous "Henry system" of fingerprint classification [25], [42], an elaborate method of indexing fingerprints very much tuned to facilitating the human experts in performing (manual) fingerprint identification. By the early twentieth century, the formations of fingerprints were well understood. The biological principles of fingerprints are summarized below.

- Individual epidermal ridges and furrows (valleys) have different characteristics for different fingers.
- The configuration types are individually variable but they vary within limits that allow for systematic classification.
- The configurations and minute details of individual ridges and furrows are permanent and unchanging for a given finger.

In the early twentieth century, fingerprint identification was formally accepted as a valid personal-identification method by law-enforcement agencies and became a standard routine in forensics [42]. Fingerprint-identification agencies were set up worldwide, and criminal fingerprint data bases were established [42].

Starting in the early 1960's, the Federal Bureau of Investigation (FBI) home office in the United Kingdom and the Paris Police Department invested a large amount of effort in developing automatic fingerprint-identification systems (AFIS's) [25]. Their efforts were so successful that a large number of AFIS's are currently installed and in operation at law-enforcement agencies worldwide. These systems have greatly improved the operational productivity of these agencies and reduced the cost of hiring and training human fingerprint experts for manual fingerprint identification. Encouraged by the success achieved by AFIS's in law-enforcement agencies, automatic fingerprint identification

rapidly grew beyond law enforcement into civilian applications [25], [53]. In fact, fingerprint-based biometric systems are so popular that they have almost become the synonym of biometric systems [20]. Although significant progress has been made in designing automatic fingerprint-authentication systems over the past 30 years, a number of design factors (lack of reliable minutiae-extraction algorithms [48], [54], difficulty in quantitatively defining a reliable match between fingerprint images [43], [45], poor fingerprint classification algorithms [12], [14] [39], [46], [57], [74], etc.) create bottlenecks in achieving the desired performance [25], [42].

C. Design of a Fingerprint-Verification System

An automatic fingerprint identity authentication system has four main design components: acquisition, representation (template), feature extraction, and matching.

1) Acquisition: There are two primary methods of capturing a fingerprint image: inked (off-line) and live scan (ink-less). 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 for identity verification.[4] The most popular technology to obtain a live-scan fingerprint image is based on the optical frustrated total internal reflection (FTIR) concept [28]. When a finger is placed on one side of a glass platen (prism), ridges of the finger are in contact with the platen while the valleys of the finger are not. The rest of the imaging system essentially consists of an assembly of a light emitting diode (LED) light source and a charge-couple device (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 that is incident on the plate at the glass surface touched by the ridges is randomly scattered, while the light incident at the glass surface corresponding to valleys suffers total internal reflection, resulting in a corresponding fingerprint image on the imaging plane of the CCD.

A number of other live-scan imaging methods are now available, based on ultrasound total internal reflection [61], optical total internal reflection of edge-lit holograms [21], thermal sensing of the temperature differential (across the ridges and valleys) [41], sensing of differential capacitance [47], and noncontact three-dimensional scanning [44]. These alternate methods are primarily concerned with either reducing the size/price of the optical scanning system or improving the quality/resolution/consistency of the image capture. Typical specifications for the optical live-scan fingerprints are specified in [60].

2) Representation (Template): Which machine-readable representation completely captures the invariant and discriminatory information in a fingerprint image? This representation issue constitutes the essence of fingerprint-verification design and has far-reaching implications on the design of the rest of the system. The unprocessed gray-scale values of the fingerprint images are not invariant over the time of capture.

Representations based on the entire gray-scale profile of a fingerprint image are prevalent among the verification systems using optical matching [4], [50]. The utility of the systems using such representation schemes, however, may be limited due to factors like brightness variations, image-quality variations, scars, and large global distortions present in the fingerprint image because these systems are essentially resorting to template-matching strategies for verification. Further, in many verification applications, terser representations are desirable, which preclude representations that involve the entire gray-scale profile fingerprint images. Some system designers attempt to circumvent this problem by restricting that the representation is derived from a *small* (but consistent) part of the finger [50]. If this same representation is also being used for identification applications, however, then the resulting systems might stand a risk of restricting the number of unique identities that could be handled simply because of the fact that the number of distinguishable templates is limited. On the other hand, an image-based representation makes fewer assumptions about the application domain (fingerprints) and therefore has the potential to be robust to wider varieties of fingerprint images. For instance, it is extremely difficult to extract a landmark-based representation from a (degenerate) finger devoid of any ridge structure.

Representations that rely on the entire ridge structure (ridge-based representations) are largely invariant to the brightness variations but are significantly more sensitive to the quality of the fingerprint image than the landmark-based representations described below. This is because the presence of the landmarks is, in principle, easier to verify [75].

An alternative to gray-scale-based representation is to extract landmark features from a binarized fingerprint image. Landmark-based representations are also used for privacy reasons—one cannot reconstruct the entire fingerprint image from the fingerprint landmark information alone. The common hypothesis underlying such representations is the belief that the individuality of fingerprints is captured by the local ridge structures (minute details) and their spatial distributions [25], [42]. Therefore, automatic fingerprint verification is usually achieved with minute-detail matching instead of a pixel-wise matching or a ridge-pattern matching of fingerprint images. In total, there are approximately 150 different types of local ridge structures that have been identified [42]. It would be extremely difficult to automatically, quickly, and reliably extract these different representations from the fingerprint images because 1) some of them are so similar to each other and 2) their characterization

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

[4] Again, MasterCard relies on inked impressions for *enrollment*.

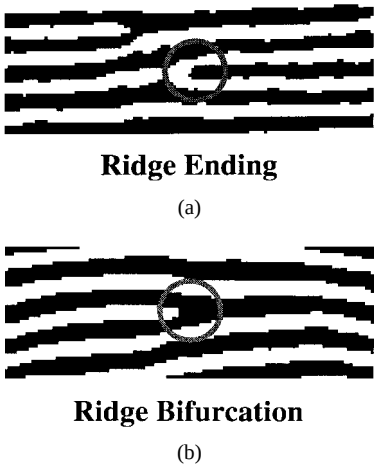


Fig. 3. Ridge ending and ridge bifurcation.

depends upon the fine details of the ridge structure, which are notoriously difficult to obtain from fingerprint images of a variety of quality. Typically, automatic fingerprint identification and authentication systems rely on representing the two most prominent structures[5]: ridge endings and ridge bifurcations. Fig. 3 shows examples of ridge endings and ridge bifurcations. These two structures are background-foreground duals of each other, and pressure variations could convert one type of structure into the other. Therefore, many common representation schemes do not distinguish between ridge endings and bifurcations. Both the structures are treated equivalently and are collectively called minutiae. The simplest of the minutiae-based representations constitute a list of points defined by their spatial coordinates with respect to a fixed image-centric coordinate system. Typically, though, these minimal minutiae-based representations are further enhanced by tagging each minutiae (or each combination of minutiae subset, e.g., pairs, triplets) with additional features. For instance, each minutiae could be associated with the orientation of the ridge at that minutiae; or each pair of the minutiae could be associated with the ridge count: the number of ridges visited during the linear traversal between the two minutiae. The American National Standards Institute–National Institute of Standards and Technology (NIST) standard representation of a fingerprint is based on minutiae and includes minutiae location and orientation [2]. The minutiae-based representation might also include one or more global attributes like orientation of the finger, locations of core or delta,[6] and fingerprint class.

Our representation is minutiae based, and each minutia is described by its location (x, y coordinates) and the orientation. We also store a short segment of the ridge associated with each minutia.

3) 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. In practice, however, 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 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. Reliable minutiae-extraction algorithms should not assume perfect ridge structures and should degrade gracefully with the quality of fingerprint images. We have developed a modified version of the minutiae-extraction algorithm proposed in [58] that is faster and more reliable. Our minutiae-extraction scheme is described in the Section II.

4) Matching: Given two (test and reference) representations, the matching module determines whether the prints are impressions of the same finger. The matching phase typically defines a metric of the similarity between two fingerprint representations. The matching stage also defines a threshold to decide whether a given pair of representations are of the same finger (mated pair) or not.

In the case of the minutiae-based representations, the fingerprint-verification problem may be reduced to a point pattern matching (minutiae pattern matching) problem. In the ideal case, if 1) the correspondence between the template minutiae pattern and input minutiae pattern is known, 2) there are no deformations such as translation, rotation, and deformations between them, and 3) each minutia present in a fingerprint image is exactly localized, then fingerprint verification is only a trivial task of counting the number of spatially matching pairs between the two images. Determining whether two representations of a finger extracted from its two impressions, possibly separated by a long duration of time, are indeed representing the same finger is an extremely difficult problem. Fig. 4 illustrates the difficulty with an example of two images of the same finger. The difficulty can be attributed to two primary reasons. First, if the test and reference representations are indeed mated pairs, the correspondence between the test and reference minutiae in the two representations is not known. Second, the imaging system presents a number of peculiar and challenging situations, some of which are unique to a fingerprint image capture scenario.

1) *Inconsistent contact:* the act of sensing distorts the finger. Determined by the pressure and contact of the finger on the glass platen, the three-dimensional shape of the finger gets mapped onto the two-dimensional surface of the glass platen. Typically, this mapping function is uncontrolled and results in different inconsistently mapped fingerprint images across the impressions.
2) *Nonuniform contact:* The ridge structure of a finger would be completely captured if ridges of the part of the finger being imaged are in complete optical contact with the glass platen. However, the dryness of

[5] Many of the other ridge structures could be described as a combination of ridge endings and bifurcations [42].

[6] Core and delta are the two distinctive global structures in a fingerprint [25]; see Fig. 2(c).

(a)

(b)

Fig. 4. Two different fingerprint impressions of the same finger. To know the correspondence between the minutiae of these two fingerprint images, all of the minutiae must be precisely localized and the deformations must be recovered.

the skin, skin disease, sweat, dirt, and humidity in the air all confound the situation, resulting in a nonideal contact situation: some parts of the ridges may not come in complete contact with the platen, and regions representing some valleys may come in contact with the glass platen. This results in "noisy" low-contrast images, leading to either spurious minutiae or missing minutiae.

3) *Irreproducible contact:* manual work, accidents, etc. inflict injuries to the finger, thereby changing the ridge structure of the finger either permanently or semipermanently. This may introduce additional spurious minutiae.
4) *Feature extraction artifacts:* The feature extraction algorithm is imperfect and introduces measurement errors. Various image-processing operations might introduce inconsistent biases to perturb the location and orientation estimates of the reported minutiae from their gray-scale counterparts.
5) *Sensing act:* the act of sensing itself adds noise to the image. For example, residues are leftover from the previous fingerprint capture. A typical finger-imaging system distorts the image of the object being sensed due to imperfect imaging conditions. In the FTIR sensing scheme, for example, there is a geometric distortion because the image plane is not parallel to the glass platen.

In light of the operational environments mentioned above, the design of the matching algorithms needs to establish and characterize a realistic model of the variations among the representations of mated pairs. This model should include the properties of interest listed below.

a) The finger may be placed at different locations on the glass platen, resulting in a (global) translation of the minutiae from the test representation from those in the reference representation.
b) The finger may be placed in different orientations on the glass platen, resulting in a (global) rotation of the minutiae from the test representation from that of the reference representation.
c) The finger may exert a different (average) downward normal pressure on the glass platen, resulting in a (global) spatial scaling of the minutiae from the test representation from those in the reference representation.
d) The finger may exert a different (average) shear force on the glass platen, resulting in a (global) shear transformation (characterized by a shear direction and magnitude) of the minutiae from the test representation from those in the reference representation.
e) Spurious minutiae may be present in both the reference and the test representations.
f) Genuine minutiae may be absent in the reference or test representations.
g) Minutiae may be locally perturbed from their "true" location, and the perturbation may be different for each individual minutiae. (Further, the magnitude of such perturbation is assumed to be small and within a fixed number of pixels.)
h) The individual perturbations among the corresponding minutiae could be relatively large (with respect to ridge spacings), but the perturbations among pairs of the minutiae are spatially linear.
i) The individual perturbations among the corresponding minutiae could be relatively large (with respect to

Fig. 5. Aligned ridge structures of mated pairs. Note that the best alignment in one part (top left) of the image results in a large displacements between the corresponding minutiae in the other regions (bottom right).

ridge spacings), but the perturbations among pairs of the minutiae are spatially nonlinear.

j) Only a (ridge) connectivity preserving transformation could characterize the relationship between the test and reference representations [73].

A matcher may rely on one or more of these assumptions, resulting in a wide spectrum of behavior. At the one end of the spectrum, we have the "Euclidean" matchers, which allow only rigid transformations among the test and reference representations. At the other extreme, we have a "topological" matcher, which may allow the most general transformations, including, say, order reversals.[7] The choice of assumptions often represents verification performance tradeoffs. Only a highly constrained system with not too demanding accuracies could get away with restrictive assumptions. A number of the matchers in the literature assume similarity transformation [assumptions a), b), and c)]; they tolerate both spurious minutiae as well as missing genuine minutiae. "Elastic" matchers in the literature accommodate a small bounded local perturbation of minutiae from their true location but cannot handle large displacements of the minutiae from their true locations [59].

Fig. 5 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 the other regions (bottom right). In addition, observe that the distortion is nonlinear: given distortions at two arbitrary locations on the finger, it is not possible to predict the distortion at all of the intervening points on the line joining the two points. In the authors' opinion, a good matcher needs to accommodate not only global similarity transformations [assumptions a), b), and c)] but also shear transformation [assumption d)] and linear [assumption h)]

[7] Order reversal means that a set of minutiae in the test representation are in totally different spatial order with respect to their correspondences in the reference representation.

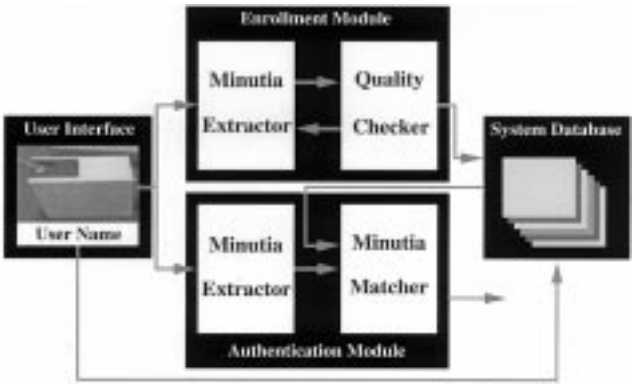


Fig. 6. Architecture of the automatic identity-authentication system.

and nonlinear [assumption i)] differential distortions. In our experience, assumption j) is too general a model to characterize the impressions of a finger, and its inclusion into the matcher design may compromise efficiency and discriminatory power of the matcher. In addition, the matchers based on such assumptions need to use connectivity information, which is notoriously difficult to extract from fingerprint images of poor quality.

We have proposed an alignment-based elastic matching algorithm. This algorithm is capable of finding the correspondences between minutiae without resorting to an exhaustive search and has the ability to compensate adaptively for the nonlinear deformations and inexact transformations between different fingerprints. Given a pair of appropriately aligned fingerprint representations and a set of already matched pairs of minutiae, the matching algorithm incrementally and adaptively stretches (contracts) the positions of candidate minutiae currently being matched as a function of the minutiae pairs that are already matched. Estimated orientations of minutiae are often inaccurate in fingerprint images of poor quality. Our algorithm accommodates noise in the minutiae orientations by permitting large discrepancy between the corresponding minutiae (30°). The matching algorithm is described in detail in Section III.

D. An Automatic Identity-Authentication System

We will introduce a prototype automatic identity authentication system, which is capable of automatically authenticating the identity of an individual using fingerprints. Currently, it is mainly intended for user authentication. For example, our system can be used to replace password authentication during the log-in session in a multiuser computing environment.

The architecture of our automatic identity authentication system is shown in Fig. 6. It consists of four components: 1) user interface, 2) system data base, 3) enrollment module, and 4) authentication module. The user interface provides mechanisms for a user to indicate his identity and input his fingerprints into the system. The system data base 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 that are used for authentication purposes: 1) user name of the person, 2) minutiae templates of the person's fingerprint, and 3) other information.

The task of the enrollment module is to enroll persons and their fingerprints into the system data base. 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 [29] is used to ensure that the records in the system data base consist only of fingerprints of good quality, in which a significant number (default value is 25) of genuine minutiae may be detected. This is important because there is no point in using a minutiae pattern with only a very limited number of genuine minutiae as a template to make an authentication. 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 recovered reliably [29]. The enhanced fingerprint image is fed to the minutiae extractor again. Because the current quality-checking algorithm is very slow [29], it is only used in the enrollment module.

The task of the authentication module is to authenticate the identity of the person who intends to access the system. The person to be authenticated indicates his identity and places his finger on the fingerprint scanner; a digital image of his fingerprint is captured; and a 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 data base to establish the identity.

In the following sections, we will describe in detail the minutiae-extraction algorithm, the minutiae-matching algorithm, and the experimental results on two fingerprint data bases. Section II mainly discusses the fingerprint minutiae-extraction algorithm. Section III presents our minutiae-matching algorithm. Experimental results on the Michigan State University (MSU) fingerprint data bases captured with an inkless scanner and NIST 9 fingerprint data base are described in Section IV. Section V contains the summary and conclusions.

II. Minutiae Extraction

Fingerprint authentication is based on the matching of minutiae patterns. A reliable minutiae-extraction algorithm is critical to the performance of an automatic identity-authentication system using fingerprints. In our system, we have developed a minutiae-extraction algorithm that is an improved version of the technique described in [58]. Experimental results show that this algorithm performs very well in operation. The overall flowchart of this algorithm is depicted in Fig. 7. It mainly consists of three components: 1) orientation field estimation, 2) ridge extraction, and 3) minutiae extraction and postprocessing. In the following subsections, we will describe in detail our minutiae-extraction algorithm. We assume that the resolution of input fingerprint images is 500 dots per inch.

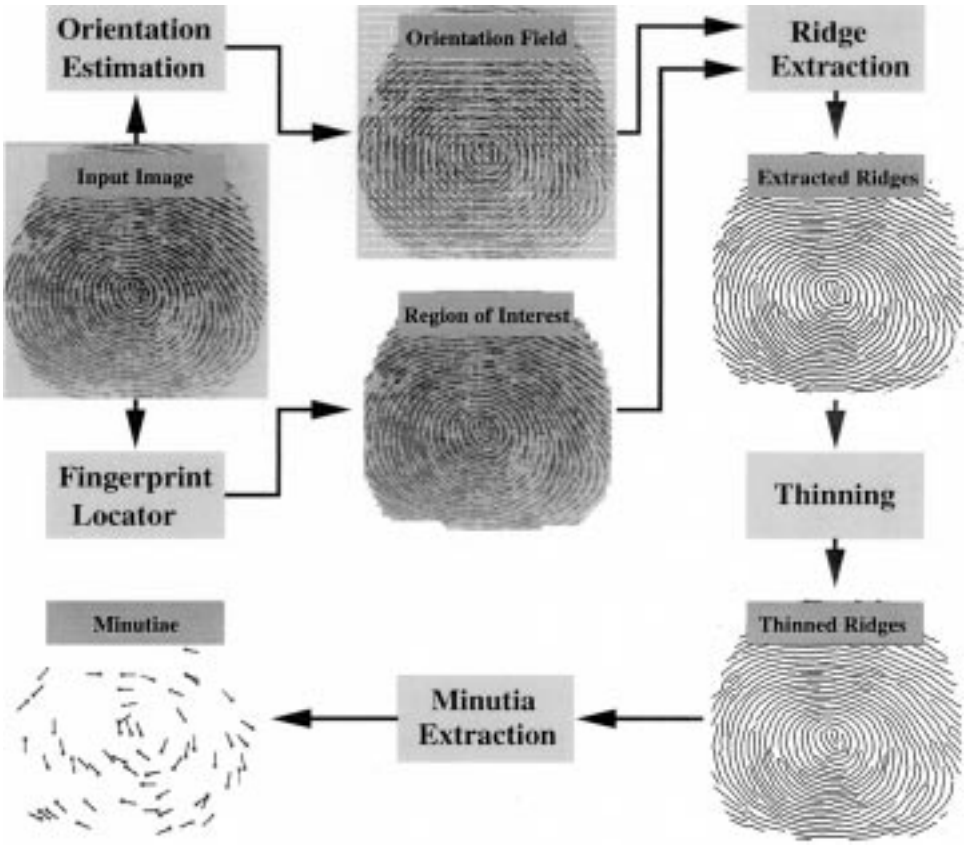


Fig. 7. Flowchart of the minutiae-extraction algorithm.

A. Orientation Field Estimation

The orientation field of a fingerprint image represents the intrinsic nature of 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 [38], [40], [56]. In our system, a new hierarchical implementation of the algorithm proposed in [56] is used (Fig. 8). With this algorithm, a fairly smooth orientation-field estimate can be obtained. Fig. 9 shows the orientation field of a fingerprint image estimated with our hierarchical algorithm.

After the orientation field of an input fingerprint image is estimated, a segmentation algorithm, which is based on the local certainty level of the orientation field, is used to locate the region of interest within the input fingerprint image. The certainty level of the orientation field at pixel (i,j) is defined as follows:

$$\mathrm{CL}(i,j) = \sqrt{\frac{1}{W \times W} \frac{(V_x(i,j)^2 + V_y(i,j)^2)}{V_e(i,j)}} \qquad (2)$$

where

$$V_e(i,j) = \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} \left(G_x^2(u,v) + G_y^2(u,v)\right) \qquad (3)$$

and W is the size of a local neighborhood. For each pixel, if its certainty level of the orientation field is below a certain threshold T_s, then the pixel is marked as a background pixel. In our localization algorithm, we assume that only one fingerprint is present in the image, which is used as a heuristic to find the region of interest.

B. Ridge Detection

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. Pixels can be identified to be ridge pixels based on this property. In our minutiae-detection algorithm, a fingerprint image is first convolved with two masks, $h_t(i,j;u,v)$ and $h_b(i,j;u,v)$, of size $L \times H$ (on an average 11×7), respectively. These two masks are capable of adaptively accentuating the local maximum gray-level values along a direction normal to the local ridge direction

$$h_t(i,j;u,v) = \begin{cases} -\frac{1}{\sqrt{2\pi}\delta} e^{-\frac{u}{\delta^2}}, & \text{if } u = (v\cot(\theta(i,j)) \\ & \quad -\frac{H}{2\cos(\theta(i,j))}), \quad v \in \Omega \\ \frac{1}{\sqrt{2\pi}\delta} e^{-\frac{u}{\delta^2}}, & \text{if } u = (v\cot(\theta(i,j))), \quad v \in \Omega \\ 0, & \text{otherwise} \end{cases} \qquad (4)$$

$$h_b(i,j;u,v) = \begin{cases} -\frac{1}{\sqrt{2\pi}\delta} e^{-\frac{u}{\delta^2}}, & \text{if } u = (v\cot(\theta(i,j)) \\ & \quad +\frac{H}{2\cos(\theta(i,j))}), \quad v \in \Omega \\ \frac{1}{\sqrt{2\pi}\delta} e^{-\frac{u}{\delta^2}}, & \text{if } u = (v\cot(\theta(i,j))), \quad v \in \Omega \\ 0, & \text{otherwise} \end{cases} \qquad (5)$$

$$\Omega = \left[-\left| \frac{L\sin(\theta(i,j))}{2} \right|, \left| \frac{L\sin(\theta(i,j))}{2} \right| \right] \qquad (6)$$

1. *Divide the input fingerprint image into blocks of size $W \times W$.*

2. *Compute the gradients G_x and G_y at each pixel in each block [11].*

3. *Estimate the local orientation at each pixel (i,j) using the following equations [56]:*

$$V_x(i,j) = \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} 2G_x(u,v)G_y(u,v),$$

$$V_y(i,j) = \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} (G_x^2(u,v) - G_y^2(u,v)),$$

$$\theta(i,j) = \frac{1}{2} tan^{-1}\left(\frac{V_x(i,j)}{V_y(i,j)}\right),$$

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

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

$$C(i,j) = \frac{1}{N}\sqrt{\sum_{(i',j')\in D} |\theta(i',j') - \theta(i,j)|^2},$$

$$|\theta' - \theta| = \begin{cases} d & \text{if } (d = (\theta' - \theta + 360) \bmod 360) < 180, \\ d - 180 & \text{otherwise}, \end{cases}$$

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.

5. *If the* consistency level *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.*

Fig. 8. Hierarchical orientation-field estimation algorithm.

where $\theta(i,j)$ represents the local ridge direction at pixel (i,j) and δ is a large constant. If *both* of the gray-level values at pixel (i,j) of the convolved images are larger than a certain threshold T_{ridge}, then pixel (i,j) is labeled as a ridge. By adapting the mask width to the width of the local ridge, this algorithm can efficiently locate the ridges in a fingerprint image. Due to the presence of noise, breaks, and smudges, etc. in the input image, however, the resulting binary ridge map often contains holes and speckles. When ridge skeletons are used for the detection of minutiae, the presence of such holes and speckles (small spurious fragments detected as ridges) will severely deteriorate the performance of our minutiae-extraction algorithm because these holes and speckles may drastically change the skeleton of the ridges. Therefore, a procedure to remove the holes and speckles needs to be applied before ridge thinning.

C. Minutiae Detection

Minutiae detection is a trivial task when an ideal thinned ridge map is available. Without loss of generality, we assume that if a pixel is on a thinned ridge (eight-connected), then it has a value of one, and zero otherwise. Let (x,y) denote a pixel on a thinned ridge and $N_0, N_1, \cdots, N_7$ denote its eight neighbors. A pixel (x,y) is a ridge ending if $(\sum_{i=0}^{7} N_i) = 1$ and a ridge bifurcation if $(\sum_{i=0}^{7} N_i) > 2$. However, the presence of undesired spikes and breaks present in a thinned ridge map may lead to detection of many spurious minutiae. Therefore, before the minutiae detection, a smoothing procedure is applied to remove spikes and to join broken ridges. Our ridge-smoothing algorithm uses the following heuristics.

- If the angle formed by a branch and the trunk ridge is larger than T_{lower} $(= 70°)$ and less than T_{upper} $(= 110°)$ and the length of the branch is less than T_{branch} $(= 20$ pixels), then the branch is removed.
- If a break in a ridge is shorter than T_{break} $(= 15$ pixels) and no other ridges pass through it, then the break is connected.

The parameters controlling the behavior of the ridge-smoothing heuristic are presently set to very conservative values. Although it is possible that the ridge-smoothing algorithm may occasionally annihilate genuine minutiae, by and large, it deletes the spurious minutiae generated by the poor quality image, image-processing artifacts, and fingerprint creases.

For each detected minutiae, the following parameters are recorded: 1) x-coordinate, 2) y-coordinate, 3) orientation, which is defined as the local ridge orientation of the associated ridge, and 4) the associated ridge segment. The recorded ridges are represented as one-dimensional discrete signals, which are normalized by a preset length parameter that is approximately equal to the average inter-ridge distance of the finger (presently computed manually once for the given imaging setup). About ten locations on the ridge associated with each ridge are sampled per minutiae. The entire representation for a finger when stored in a compressed format takes, on an average, about 250 bytes. These recorded ridges are used for alignment in

(a)

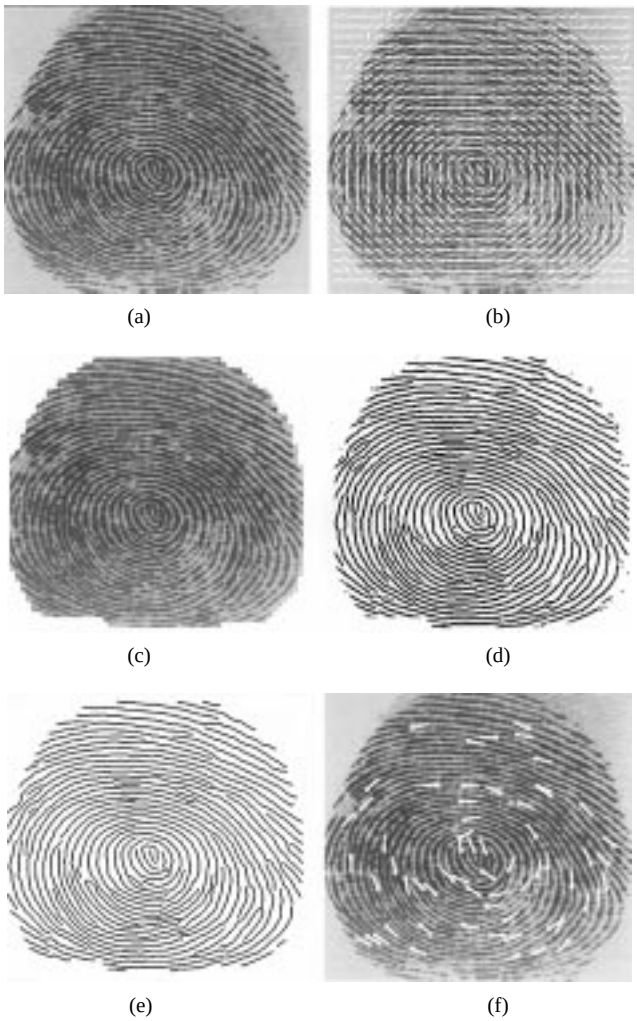
(b)

Fig. 9. Comparison of orientation fields by the method proposed in [56] and the hierarchical method; the block size $(W \times W)$ is 16×16 and the size of D is 5×5. (a) Method proposed in [56]. (b) Hierarchical method.

the minutiae-matching stage. Fig. 10 shows the results of our minutiae-extraction algorithm on a fingerprint image captured with an inkless scanner.

III. Minutiae Matching

Fingerprint matching has been approached from several different strategies, like image-based [4], [50] and ridge-pattern matching of fingerprint representations. There also exist graph-based schemes [22], [23], [26], [27], [34], [36] for fingerprint matching. Our automatic fingerprint-verification algorithm instead is based on point pattern



(a) (b) (c) (d) (e) (f)

Fig. 10. Results of our minutiae-extraction algorithm on a fingerprint image (512×512) captured with an inkless scanner. (a) Input image. (b) Orientation field superimposed on the input image. (c) Fingerprint region. (d) Extracted ridges. (e) Thinned ridge map. (f) Extracted minutiae and their orientations superimposed on the input image.

matching (minutiae matching). The reason for this choice is our need to design a robust, simple, and fast verification algorithm and to keep a small template size. A number of point pattern matching algorithms have been proposed in the literature [1], [55], [63], [66], [69], [71]. A general point matching problem is essentially intractable. Features associated with points and their spatial properties, such as the relative distances between points, are widely used in these algorithms to reduce the exponential number of search paths.

The relaxation approach to point pattern matching [55] iteratively adjusts the confidence level of each corresponding pair based on its consistency with other pairs until a certain criterion is satisfied. Although a number of modified versions of this algorithm have been proposed to reduce the matching complexity [69], these algorithms are inherently slow because of their iterative nature.

The generalized Hough transform-based approach to point pattern matching [6], [67] converts point pattern matching to a problem of detecting peaks in the Hough

1. *Estimate the translation and rotation parameters between the ridge associated with each input minutiae and the ridge associated with each template minutiae and align the two minutiae patterns according to the estimated parameters.*

2. *Convert the template pattern and input pattern into the polar coordinate representations with respect to the corresponding minutiae on which alignment is achieved and represent them as two symbolic strings by concatenating each minutiae in an increasing order of radial angles:*

$$P_p = ((r_1^P, e_1^P, \theta_1^P)^T, ..., (r_M^P, e_M^P, \theta_M^P)^T)$$
$$Q_p = ((r_1^Q, e_1^Q, \theta_1^Q)^T, ..., (r_N^Q, e_N^Q, \theta_N^Q)^T),$$

where r_*, e_*, and θ_* represent the corresponding radius, radial angle, and normalized minutiae orientation with respect to the reference minutiae, respectively.

3. *Match the resulting strings P_p and Q_p with a modified dynamic-programming algorithm described below to find the 'edit distance' between P_p and Q_p.*

4. *Use the minimum edit distance between P_p and Q_p to establish the correspondence of the minutiae between P_p and Q_p. The matching score, S, is then defined as:*

$$S = \frac{100 M_{PQ} M_{PQ}}{MN},$$

where M_{PQ} is the number of minutiae which fall in the bounding boxes of template minutiae. The bounding box of a minutiae specifies the possible positions of the corresponding input minutiae with respect to the template minutiae.

Fig. 11. The alignment-based minutiae-matching algorithm.

space of transformation parameters. It discretizes the parameter space and accumulates evidence in the discretized space by deriving transformation parameters that relate two point patterns using a substructure or feature matching technique. A hierarchical Hough transform-based algorithm may be used to reduce the size of the accumulator array by using a multiresolution approach. If there are only a few minutiae points available, however, it is very difficult to accumulate enough evidence in the Hough transform space for a reliable match.

Tree-pruning approaches attempt to find the correspondence between a pair of point sets by searching over a tree of possible matches while employing different tree-pruning methods, such as branch-and-bound, to reduce the search space [5]. To prune the tree of possible matches efficiently, this approach tends to impose a number of requirements on the input point sets, such as an equal number of points and no outliers. These requirements are difficult to satisfy in practice, especially in a fingerprint identification/verification system.

The energy minimization approach to point pattern matching establishes the correspondence between a pair of point sets by defining an energy function based on an initial set of possible correspondences. It uses an appropriate optimization algorithm such as genetic algorithm [1] and simulated annealing [66] to find a possible suboptimal match. These methods tend to be very slow and are unsuitable for a real-time identification/verification system.

In our system, an alignment-based matching algorithm is developed. It is simple in theory, efficient in discrimination, and fast in speed. The alignment-based matching algorithm decomposes the minutiae matching into two stages:

1) *alignment stage*, where transformations such as translation, rotation, and scaling between an input and a template in the data base are estimated and the input minutiae are aligned with the template minutiae according to the estimated parameters;
2) *matching stage*, where both the input minutiae and the template minutiae are converted to polygons in the polar coordinate system and an elastic string matching algorithm is used to match the resulting polygons.

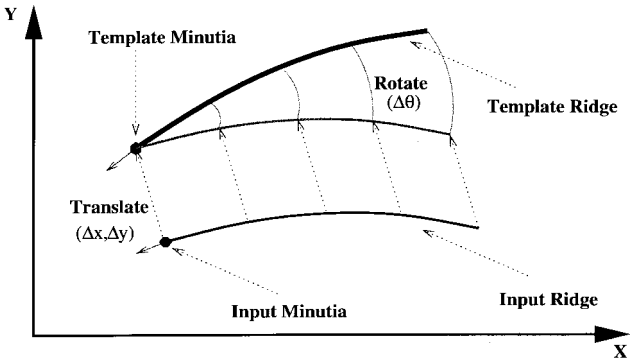


Fig. 12. Alignment of the input ridge and the template ridge.

Let $P = \left((x_1^P, y_1^P, \theta_1^P)^T, \cdots, (x_M^P, y_M^P, \theta_M^P)^T\right)$ and $Q = \left((x_1^Q, y_1^Q, \theta_1^Q)^T, \cdots, (x_N^Q, y_N^Q, \theta_N^Q)^T\right)$ denote the M minutiae in the template and the N minutiae in the input image, respectively. Our alignment-based matching algorithm is depicted in Fig. 11.

A. Alignment of Point Patterns

Ideally, two sets of planar point patterns can be aligned completely by only two corresponding point pairs. A true alignment between two point patterns can be obtained by testing all possible corresponding point pairs and selecting the optimal one. Due to the presence of noise and

1. *For each ridge $d \in R^d$, represent it as an one-dimensional discrete signal and match it against each ridge, $D \in R^D$ according to the following formula:*

$$S = \frac{\sum_{i=0}^{L} d_i D_i}{\sqrt{\sum_{i=0}^{L} d_i^2 D_i^2}},$$

where L is the minimal length of the two ridges and d_i and D_i represent the distances from point i on the ridges d and D to the x-axis, respectively. The sampling interval on a ridge is set to the average inter-ridge distance. If the matching score S ($0 \leq S \leq 1$) is larger than a certain threshold T_r (0.8), then go to step 2, otherwise continue to match the next pair of ridges.

2. *Estimate the transformation between the two ridges (Figure 12). Generally, a least-square method can be used to estimate the pose transformation. However, in our system, we observe that the following method is capable of achieving the same accuracy with fewer computations. The translation vector $(\Delta x, \Delta y)^T$ between the two corresponding ridges is computed as*

$$\begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} x^d \\ y^d \end{pmatrix} - \begin{pmatrix} x^D \\ y^D \end{pmatrix},$$

where $(x^d, y^d)^T$ and $(x^D, y^D)^T$ are the x and y coordinates of the two minutiae, which are called reference minutiae, associated with the ridges d and D, respectively. The rotation angle $\Delta\theta$ between the two ridges is computed as

$$\Delta\theta = \frac{1}{L}\sum_{i=0}^{L}(\gamma_i - \Gamma_i),$$

where L is the minimal length of the two ridges d and D; γ_i and Γ_i are radial angles of the ith point on the ridge with respect to the reference minutiae associated with the two ridges d and D, respectively. The scaling factor between the input and template images is assumed to be 1.

3. *Denote the minutiae $(x^d, y^d, \theta^d)^T$, based on which the transformation parameters are estimated, as the reference minutiae. Translate and rotate all the N input minutiae with respect to this reference minutiae, according to the following formula:*

$$\begin{pmatrix} x_i^A \\ y_i^A \\ \theta_i^A \end{pmatrix} = \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta\theta \end{pmatrix} + \begin{pmatrix} \cos\Delta\theta & \sin\Delta\theta & 0 \\ \sin\Delta\theta & -\cos\Delta\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_i - x^d \\ y_i - y^d \\ \theta_i - \theta^d \end{pmatrix},$$

where $(x_i, y_i, \theta_i)^T$, $(i = 1, 2, ..., N)$, represents an input minutiae and $(x_i^A, y_i^A, \theta_i^A)^T$ represents the corresponding aligned minutiae.

Fig. 13. The minutiae-alignment algorithm.

deformations, however, the input minutiae cannot always be aligned exactly with respect to those of the templates. To accurately recover pose transformations between two point patterns, a relatively large number of corresponding point pairs need to be used. This leads to a prohibitively large number of possible correspondences to be tested. Therefore, an alignment by corresponding point pairs is not practical even though it is feasible.

It is well known that corresponding curve segments are capable of aligning two point patterns with a high accuracy in the presence of noise and deformations [32]. Each minutiae in a fingerprint is associated with a ridge. Therefore, it is clear that a true alignment can be achieved by aligning corresponding ridges (see Fig. 12). During the minutiae-detection stage, when a minutiae is extracted and recorded, the ridge on which it resides is also recorded. This ridge is represented as a planar curve, with its origin coincident with the minutiae and its x-coordinate being in the same direction as the direction of the minutiae. Also, this planar curve is normalized with the average interridge distance. By matching these ridges, the relative pose transformation between the input fingerprint and the template can be accurately estimated. To be specific, let R^d and R^D denote the sets of ridges associated with the minutiae in the input and the template, respectively. Our alignment algorithm is described in Fig. 13. Note that because the aspect ratio of the pixels in our acquisition devices is not one (nonsquare pixels), a rectification is performed before the alignment.

B. Aligned Point Pattern Matching

If two identical point patterns are exactly aligned with each other, then each pair of corresponding points are completely coincident. In such a case, point pattern matching can be simply achieved by counting the number of overlapping pairs. In practice, however, such a situation is not encountered. On the one hand, the error in determining and localizing minutiae hinders the alignment algorithm to recover the relative pose transformation exactly, while on the other hand, our alignment scheme described in Fig. 13 does not model the nonlinear deformation of fingerprints, which is an inherent property of fingerprint impressions. With the

existence of such a nonlinear deformation, it is impossible to recover the position of each input minutiae exactly with respect to its corresponding minutiae in the template. Therefore, the aligned point pattern matching algorithm needs to be *elastic*, which means that it should be capable of tolerating, to some extent, the deformations due to inexact extraction of minutiae positions and nonlinear deformations. Usually, such an elastic matching can be achieved by placing a bounding box around each template minutiae, which specifies all the possible positions of the corresponding input minutiae with respect to the template minutiae, and restricting the corresponding minutiae in the input image to be within this box [59]. This method does not provide a satisfactory performance in practice because local deformations may be small while the accumulated global deformations can be quite large. We have proposed an adaptive elastic matching algorithm with the ability to compensate the minutiae localization errors and nonlinear deformations.

Our adaptive elastic matching algorithm consists of two main steps: 1) representing minutiae patterns as a *string* in the polar coordinate system and 2) matching the strings with a dynamic programming algorithm to establish the correspondence. Minutiae matching in the polar coordinate system has several advantages. Although the deformation of fingerprints depends on a number of factors, such as impression pressure and impression direction, the deformation in a local region is usually consistent and may become less consistent as one moves further away from the region where the fingerprint patterns are consistent (see Fig. 5). Consequently, it is easier to represent and manipulate the representations in polar space (with origin at a point of maximal consistency between the reference and aligned test template). At the same time, it is easier to formulate rotation, which constitutes the main part of the alignment error between an input image and a template, in the polar space than in the Cartesian space. The symbolic string generated by concatenating points in an increasing order of radial angle in polar coordinates uniquely represents a point pattern. This reveals that point pattern matching can be achieved with a string-matching algorithm.

A number of string-matching algorithms have been reported in the literature [15]. Generally, string matching can be thought of as the maximization/minimization of a certain cost function, such as the edit distance. Including an elastic term in the cost function of a string-matching algorithm

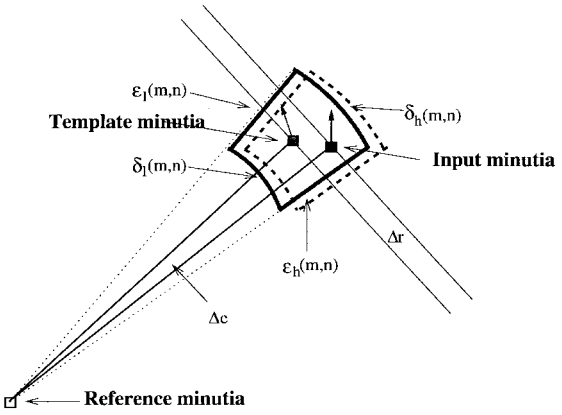


Fig. 14. Bounding box and its adjustment.

can achieve a certain amount of error tolerance. Given two strings P_p and Q_p of lengths M and N, respectively, we define the "edit distance" $C(M,N)$ recursively as shown in (7)–(10), shown at the bottom of the page, where α, β, and γ are the weights associated with radius, radial angle, and minutiae direction, respectively; δ, ϵ, and ϱ specify the bounding box; and Ω is a prespecified penalty for a mismatch. Such an edit distance, to some extent, captures the elastic property of string matching. It represents the cost of changing one string to the other. However, this scheme can only tolerate, but not compensate for, the adverse effect on matching produced by the inexact localization of minutiae and nonlinear deformations. Therefore, an adaptive mechanism is needed. This adaptive mechanism should be able to track the local nonlinear deformation and inexact alignment and try to alleviate them during the minimization process. We do not expect that this adaptive mechanism can handle the "order flip" of minutiae, however, which, to some extent, can be solved by an exhaustive reordering and matching within a local angular window.

In our matching algorithm, the adaptation is achieved by adjusting the bounding box (Fig. 14) when an inexact match is found. It can be represented as follows [(11) and (12) are shown at the bottom of the next page]:

$$\delta_l(m+1,n+1) = \delta_l(m,n) + \eta\Delta r_a \tag{13}$$

$$\delta_h(m+1,n+1) = \delta_h(m,n) + \eta\Delta r_a \tag{14}$$

$$\epsilon_l(m+1,n+1) = \epsilon_l(m,n) + \eta\Delta e_a \tag{15}$$

$$\epsilon_h(m+1,n+1) = \epsilon_h(m,n) + \eta\Delta e_a \tag{16}$$

$$C(m,n) = \begin{cases} 0, & \text{if } m=0 \text{ and } n=0 \\ \min\begin{cases} C(m-1,n)+\Omega \\ C(m,n-1)+\Omega \\ C(m-1,n-1)+w(m,n) \end{cases}, & 0<m\le M \text{ and } 0<n\le N \end{cases} \tag{7}$$

$$w(m,n) = \begin{cases} \alpha\left|r_m^P - r_n^Q\right| + \beta\Delta e + \gamma\Delta\theta, & \text{if } \left|r_m^P - r_n^Q\right| < \delta, \Delta e < \epsilon, \text{ and } \Delta\theta < \varrho \\ \Omega, & \text{otherwise} \end{cases} \tag{8}$$

$$\Delta e = \begin{cases} a, & \text{if } \left(a = \left(e_m^P - e_n^Q + 360\right) \bmod 360\right) < 180 \\ a-180, & \text{otherwise} \end{cases} \tag{9}$$

$$\Delta\theta = \begin{cases} a, & \text{if } \left(a = \left(\theta_m^P - \theta_n^Q + 360\right) \bmod 360\right) < 180 \\ a-180, & \text{otherwise} \end{cases} \tag{10}$$

where $w'(m,n)$ represents the penalty for matching a pair of minutiae $(r_m^P, e_m^P, \theta_m^P)^T$ and $(r_n^Q, e_n^Q, \theta_n^Q)^T$; $\delta_l(m,n)$, $\delta_h(m,n)$, $\epsilon_l(m,n)$, and $\epsilon_h(m,n)$ specify the adaptive bounding box in the polar coordinate system (radius and radial angle), and η is the learning rate. This elastic string-matching algorithm has a number of parameters that are critical to its performance. We have empirically determined the values of these parameters as follows: $\delta_l(0,0) = -8$ pixels, $\delta_h(0,0) = +8$ pixels, $\epsilon_l(0,0) = -7.5$ pixels, $\epsilon_h(0,0) = +7.5$ pixels, $\varrho = 30$, $\alpha = 1.0$, $\beta = 2.0$, $\gamma = 0.1$, $\Omega = 200(\alpha + \beta + \gamma)$, and $\eta = 0.5$. The values of $\delta_l(0,0)$, $\delta_h(0,0)$, $\epsilon_l(0,0)$, and $\epsilon_h(0,0)$ depend on the resolution of fingerprint images. Fig. 15 shows the results of applying the matching algorithm to an input and a template minutiae set pair.

IV. Experimental Results

Here, we present our experimental results on the performance of feature extraction and the entire verification system.

A. Feature-Extraction Performance

It is very difficult to assess the performance of feature-extraction algorithms independently. Accuracy of the extracted minutiae was subjectively confirmed in two ways. Visual inspection of a large number of typical minutiae-extraction results showed that our algorithm rarely missed minutiae in fingerprint images of reasonable quality.

We have also compared the performance of our feature-extraction algorithm with that of our previous feature-extraction algorithm [58]. The premise underlying this experiment is that given an identical matcher, the accuracy of the system indicates the performance of the feature-extraction algorithm. We extracted fingerprint representations from a sample set of fingerprint images using our feature-extraction algorithm. The verification accuracy was estimated using a Hough-transform-based matcher [59] by performing an "all against all" verification test to obtain match and mismatch score distributions. The same test was also performed on the features extracted from our previous feature-extraction algorithm [58]. The ROC's resulting from these two experiments (shown in Fig. 16) indicate a significant improvement in accuracy. We also plan to evaluate the performance of our feature extraction algorithm objectively by using the methodology defined in [58].

B. System Performance

We have tested our system on the MSU fingerprint data base. It contains ten images (640 $\times$ 480) per finger from 70 individuals for a total of 700 fingerprint images, which were captured with a scanner manufactured by Digital Biometrics. When these fingerprint images were captured, no restrictions on the position and orientation of fingers were imposed. The captured fingerprint images vary in quality. Fig. 17 shows some of the fingerprint images in our data base. Approximately 90% of the fingerprint images in our data base are of reasonable quality, similar to those shown in Fig. 17, while about 10% of the fingerprint images in our data base are not of good quality (Fig. 18), mainly due to large creases and smudges in ridges and dryness of the impressed finger. To establish an objective assessment of the performance, the system was also tested on a portion of the NIST 9 fingerprint data base. The NIST 9 fingerprint data base contains 1350 mated fingerprint card pairs (image size is 832 $\times$ 768) that approximate a natural distribution of the National Crime and Information Center fingerprint classes [72]. It is divided into multiple volumes. Each volume has three compact discs (CD's). Each CD contains 900 images of card type 1 and 900 images of card type 2. Fingerprints on card type 1 were scanned using a rolled method, and fingerprints on card type 2 were scanned using a live-scan method. The fingerprint images in the NIST 9 data base are difficult compared to the live-scan fingerprint images for a number of reasons, including:

1) the NIST 9 fingerprints are a combination of dabs and rolled impressions; large discrepancy between the number of minutiae in test and reference templates inherently skews the matching score normalization;
2) a large number of NIST 9 images are of much poorer image quality than a typical live-scan fingerprint image;
3) NIST 9 images often contain extraneous objects like handwritten characters and other artifacts common to inked fingerprints.

Although only one-half of the fingerprint images in the NIST 9 fingerprint data base are live-scan images and there exists a large distortion between a rolled fingerprint and a live-scan fingerprint, we can still use this data base to generate some statistics and comparative performance numbers.

$$w'(m,n) = \begin{cases} \alpha\left|r_m^P - r_n^Q\right| + \beta\Delta e + \gamma\Delta\theta, & \text{if } \begin{cases} \delta_l(m,n) < \left(r_m^P - r_n^Q\right) < \delta_h(m,n) \\ \epsilon_l(m,n) < \Delta e < \epsilon_h(m,n) \\ \Delta\theta < \varrho \end{cases} \\ \Omega, & \text{otherwise} \end{cases} \tag{11}$$

$$\begin{pmatrix} \Delta r_a \\ \Delta e_a \end{pmatrix} = \begin{cases} \begin{pmatrix} r_m^P - r_n^Q \\ \Delta e \end{pmatrix}, & \text{if} \begin{cases} \delta_l(m,n) < (r_m^P - r_n^Q) < \delta_h(m,n) \\ \epsilon_l(m,n) < \Delta e < \epsilon_h(m,n) \\ \Delta\theta < \varrho \end{cases} \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

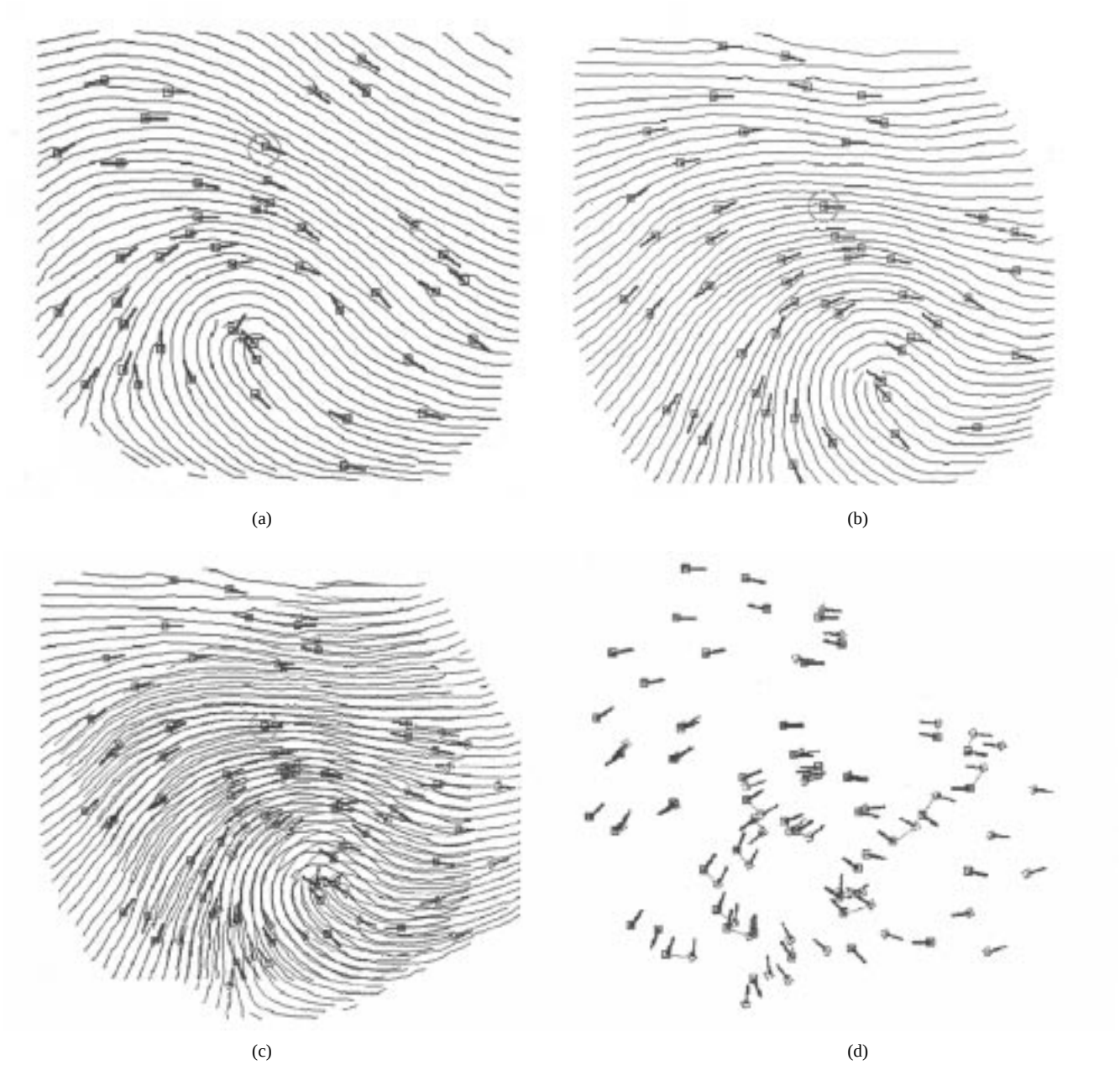


Fig. 15. Results of applying the matching algorithm to an input minutiae set and a template. (a) Input minutiae set. (b) Template minutiae set. (c) Alignment result based on the minutiae marked with green circles. (d) Matching result where template minutiae and their correspondences are connected by green lines.

C. Matching Scores

We first evaluated the matching scores of correct and incorrect matches. In test 1, each fingerprint in the MSU fingerprint data base was matched with all the other fingerprints in the data base. A matching was labeled correct if the matched fingerprint was from the same finger, and incorrect otherwise. A total of 489 300 (700×699) matchings were performed. The distributions of correct and incorrect matching scores are shown in Fig. 19(a). In test 2, each of the 900 fingerprints of card type 1 in the NIST 9 (CD no. 1) was matched with all 900 fingerprints of card type 2. A matching was labeled correct if a matched fingerprint was from the same finger. A total of 810 000 (900×900) matchings were performed on this data base (to our knowledge, no comparative results are available on the NIST 9 data base). The distributions of correct and incorrect matching scores are shown in Fig. 19(b). Table 2 lists the d' values in addition to the mean and standard deviation of correct and incorrect matching scores. The large variance of correct matching scores is mainly due to different numbers of detected minutiae, quality of acquired fingerprint images, and fingerprint distortion. For example, the fingerprint images shown in Fig. 17(a) and (b) are captured from the same finger. However, only a small region of interest is common to these two fingerprint images (approximately 30%). Obviously, it is impossible to make a

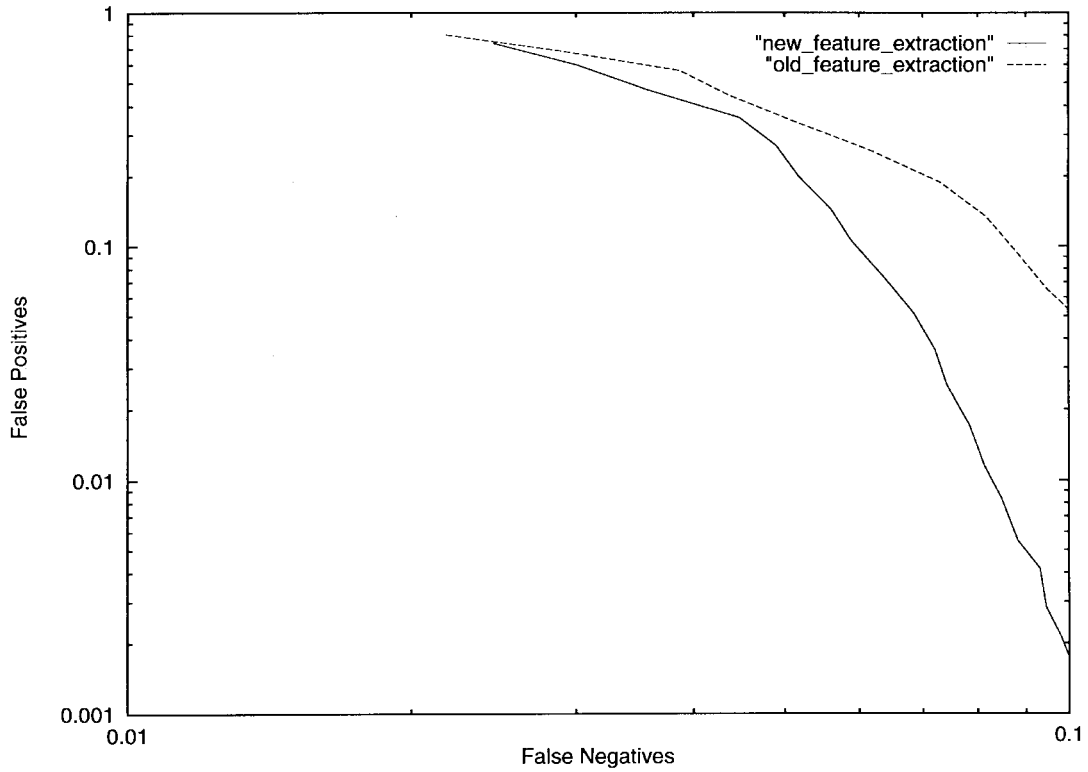


Fig. 16. ROC's showing the improvement in performance of verification due to the new version of the feature extraction.

(a) (b) (c)

Fig. 17. Fingerprint images captured with a scanner manufactured by Digital Biometrics. The size of these images is 640 × 480; all three images are from the same individual's finger.

highly confident decision based only on the limited number of minutiae appearing in the region of interest common to both fingerprints. In practice, such a problem can be solved by requiring that each input fingerprint image should have a sufficient amount of common region of interest with its stored template(s).

D. Authentication Test

In test 1, for each individual, we randomly selected three fingerprint images that passed the quality check as the template minutiae patterns for the individual and inserted them into the system data base. The major reason why we use three fingerprint templates is that a significant number of acquired fingerprint images from the same finger in the MSU data base do not have a sufficient amount of common region of interest due to the unrestricted acquisition process. Two fingerprint images may both be of good quality. If there is only a very limited amount of common region of interest, however, it is unlikely that the matching algorithm can establish a sufficient number of corresponding minutiae pairs to reach a correct decision. Using more than one template is a simple solution, although it may result in a higher FAR. There are six individuals who cannot be enrolled into the system data base because the quality of their captured fingerprints was too poor to pass the quality checking. The remaining 490 (70 × 7) fingerprint images were used as input fingerprints to test the performance of the system. An identity is established if at least one

(a)

(b)

Fig. 18. Fingerprint images of poor quality.

of the three matching scores is above a certain threshold value. Otherwise, the input fingerprint is rejected as an impostor. In test 2, we use 798 out of the 900 fingerprints of card type 1 in the NIST 9 data base (CD no. 1) as templates, which pass the quality checking. The 900 fingerprints of card type 2 were used as input fingerprints. An identity is established if the matching score is above a certain threshold value. The false acceptance rates and false reject rates with different threshold values on the matching score are shown in Table 3, which are obtained based on 31 360 (64×490) matches for test 1 and 718 200 (798×900) matches on test 2. Since the matching scores are discretized with a large sampling interval, only an approximate EER can be obtained by averaging the most

Table 2 d' and Mean and Standard Deviation of the Correct and Incorrect Matching Scores

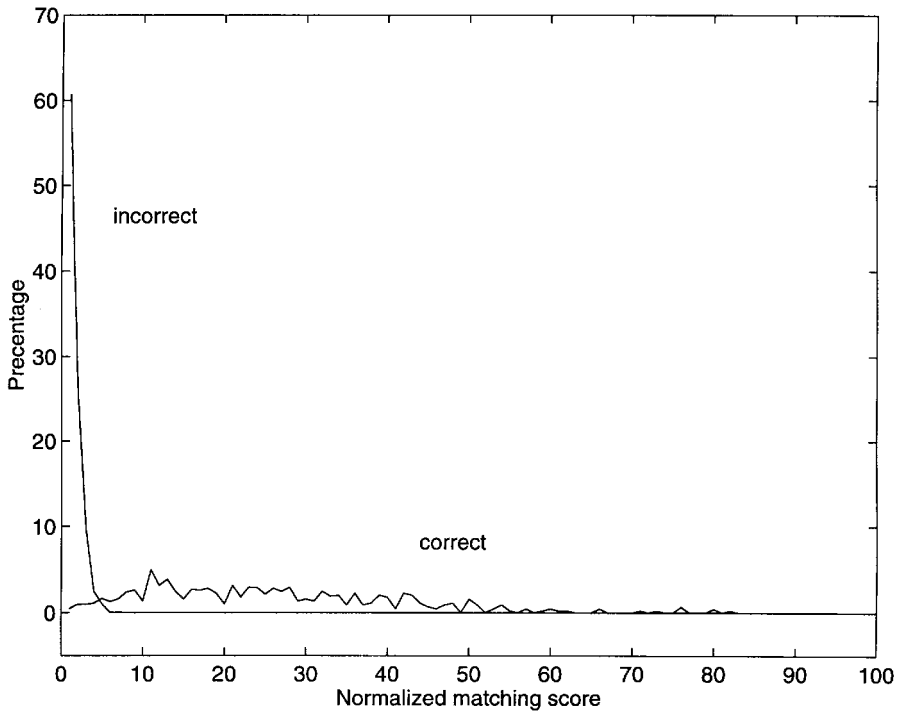
Database	d'	Mean (correct)	Standard Deviation (correct)	Mean (incorrect)	Standard Deviation (incorrect)
MSU	2.26	23.46	13.59	1.56	0.71
NIST-9	2.01	18.76	11.22	2.39	0.83

Table 3 False Acceptance and False Reject Rates on Test Sets with Different Threshold Values

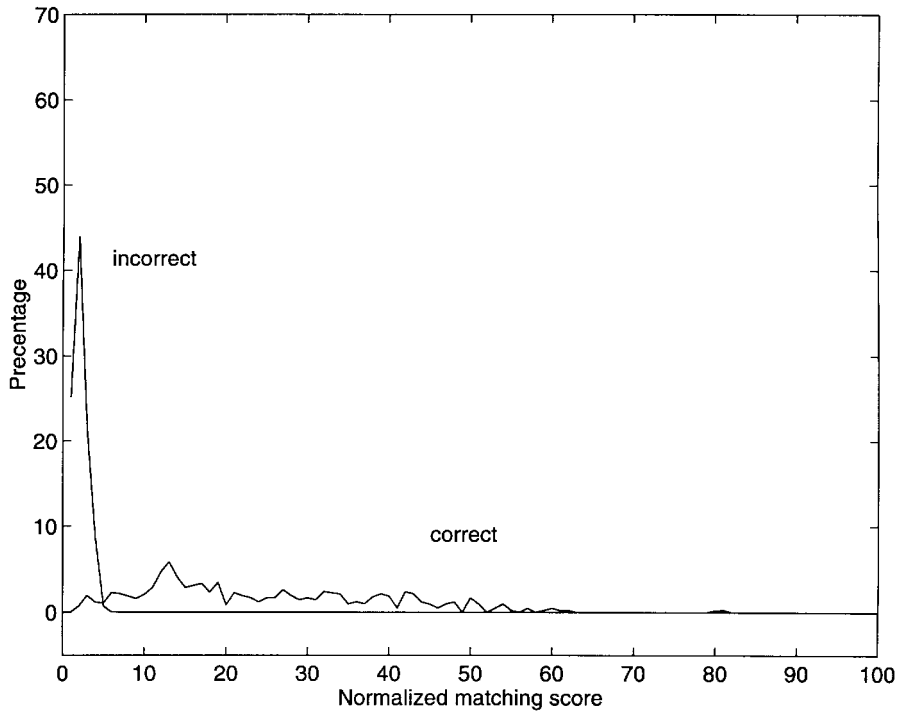
Threshold Value	False Acceptance Rate (MSU)	False Reject Rate (MSU)	False Acceptance Rate (NIST 9)	False Reject Rate (NIST 9)
7	0.07%	7.1%	0.073%	12.4%
8	0.02%	9.4%	0.023%	14.6%
9	0.01%	12.5%	0.012%	16.9%
10	0	14.3%	0.003%	19.5%

similar FAR and FRR. The EER was approximated to be 3.07% in test 1 and 2.69% in test 2. The ROC's of the two tests are shown in Fig. 20. In each ROC, authentic acceptance rate (the percentage of a genuine individual's being accepted) is plotted against the FAR. Each point on the curve corresponds to a decision criterion. In the ideal cases, if the genuine distribution and the imposter distribution are disjoint, i.e., each genuine individual is accepted and each impostor is rejected correctly, then the ROC is a horizontal line segment hovering at the authentic acceptance rate of 100%. On the other hand, if the genuine distribution and the imposter distribution are exactly the same, then the ROC is a 45° line segment with one end point at the origin (in Fig. 20, it corresponds to the dotted curve in the lower-right corner since the ROC's are plotted in semilog space). In this case, decisions can only be made by a random choice. In practice, an ROC is a curve between these two extremes. The closer the ROC is to the upper boundary, the better the system performance. The numbers shown in Table 3 are the performance measures of our verification algorithm. They should not be treated as the ultimate performance numbers of the system. In practice, a number of techniques can be employed to ensure a sufficient amount of common region of interest and good image quality and to restrict the distortion of input images, which can substantially decrease both the FAR and the FRR.

The number of tests conducted on an automatic fingerprint identification/verification system is never enough. Performance measures are as much a function of the algorithm as they are a function of the data base used for testing. The biometrics community is slow at establishing benchmarks, and the ultimate performance numbers of a fingerprint identification/verification system are those that you find in a deployed system. Therefore, one can carry out only a limited amount of testing in a laboratory environment



(a)

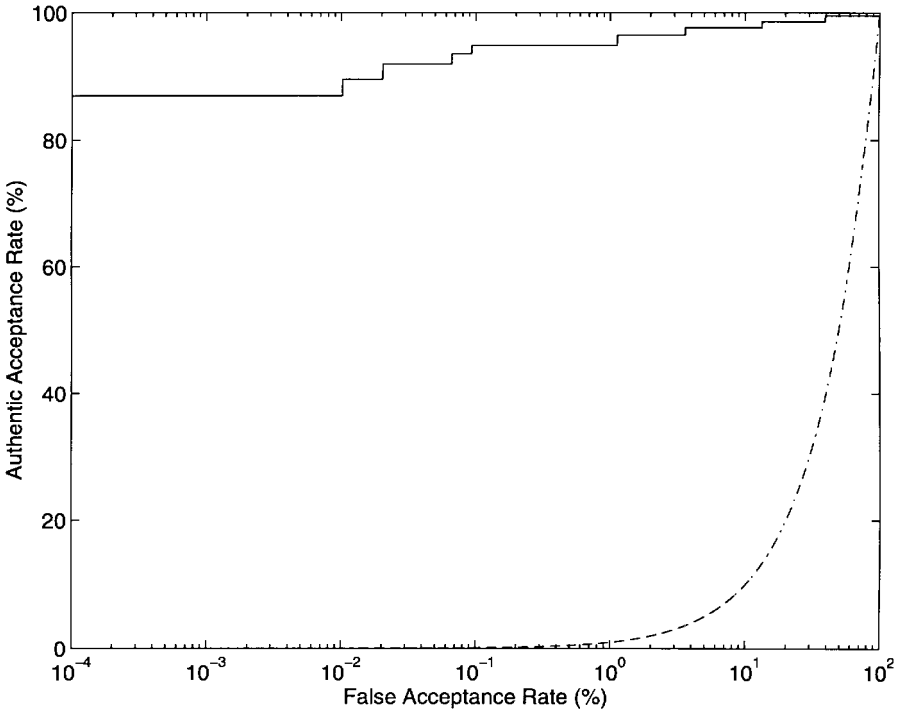


(b)

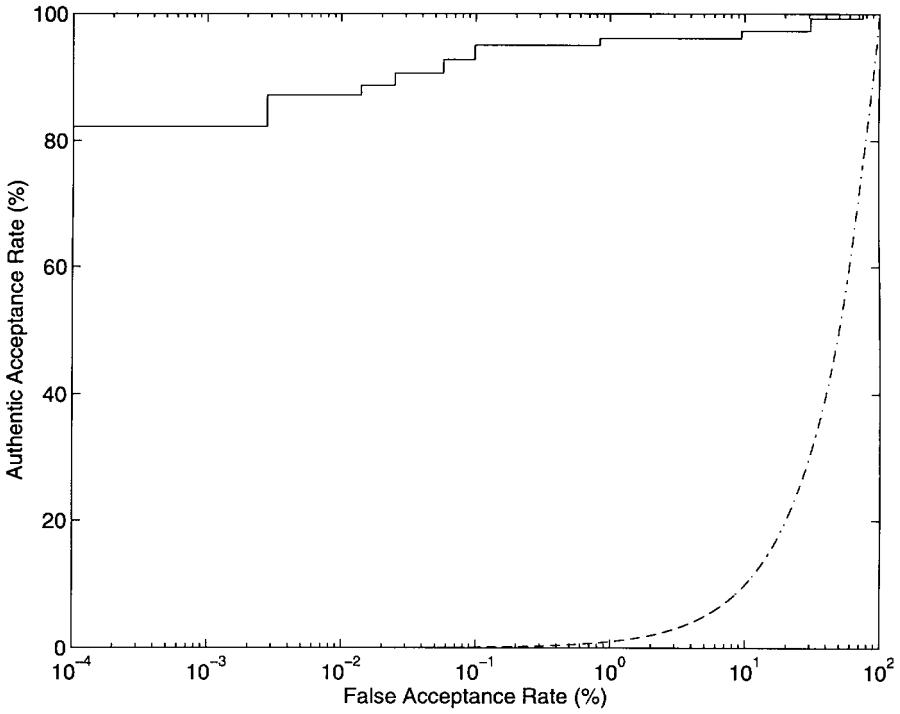
Fig. 19. Distributions of correct and incorrect matching scores; vertical axis represents distribution of matching scores in percentage. (a) MSU data base. (b) NIST 9 (CD no. 1).

to show the anticipated system performance. In field testing, in addition to the real performance of the system, the system designer has to pay attention to the perceived performance of the system, especially in the context of the authentication applications, which are sensitive to false negatives. The presentation of the matcher outcome, work flow, ergonomics, engineering, rejection criteria, operating point on the ROC, etc. play an important role in the user's perception of the system performance.

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 4 shows that our implemented system does meet the practical response-time requirement.



(a)



(b)

Fig. 20. ROC. (a) MSU data base. (b) NIST 9 (CD no. 1).

V. Summary and Conclusions

We have introduced an automatic identity-authentication system using fingerprints. The implemented minutiae-extraction algorithm is much more accurate and faster than our previous feature-extraction algorithm [58]. The proposed alignment-based elastic matching algorithm is capable of finding the correspondences between minutiae without resorting to an exhaustive search. The system achieves an excellent performance because it has the ability to compensate adaptively for the nonlinear deformations and inexact transformations between different fingerprints. Experimental results show that our system performs very well. It meets the response-time requirements as well as the accuracy requirements.

The current system is designed as a prototype system to evaluate the performance of our algorithms under different

Table 4 Average CPU Time for Minutiae Extraction and Matching on a Sun ULTRA 1 Workstation.

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

types of inputs. It should not be confused with a practical system. In practice, a number of mechanisms need to be developed besides the minutiae extraction and minutiae matching.

Based on the experimental results, we observe that the matching errors of the system mainly result from 1) insufficient number of corresponding minutiae, 2) missing minutiae and spurious minutiae, 3) inaccurate alignment, and 4) large distortion.

In practice, simple mechanisms like providing visual feedback about the fingerprint image being captured (in terms of a live display) make a significant difference in the system performance [8]. Further, prompting the user to place the finger in a desirable position and orientation also improves the false negative performance of the system. Detecting bad quality images, treating the finger with proper remedies (e.g., application of moisturizer), and subsequent recapturing of the fingerprint images are some of the useful strategies to improve the system performance.

A number of factors are detrimental to the correct localization of minutiae. Among them, poor image quality is the most serious. By integrating an enhancement mechanism into the minutiae-extraction module, this problem can, to a limited extent, be solved. Image enhancement is usually an expensive operation [29], however, which may increase the response time of the system.

Although the current minutiae-matching algorithm can compensate for the inexact alignment and distortion between an input fingerprint and its template, it cannot handle large alignment errors and large distortions. Currently, we are investigating a possible solution to this problem by incorporating a dynamic model in string matching.

Fingerprint classification is to categorize a fingerprint into a certain prespecified category based on its global pattern configuration. If two fingerprints are from the same finger, they must belong to the same category. Although fingerprint classification is still a challenging problem and it is very difficult to achieve a high classification rate, it is beneficial to incorporate the category information into a minutiae-matching algorithm to improve its discrimination performance.

A biometric system based solely on a single biometric feature may not be able to meet the practical performance requirement in all aspects. By integrating two or more biometric features, overall verification performance may be improved. For example, it is well known that fingerprint verification tends to have a larger false reject rate due to the reasons discussed above, but it has a very low false accept rate. On the other hand, face recognition is not reliable in establishing the true identity but it is efficient in searching a large data base to find the top n matches. By combining fingerprint matching and face recognition, the false reject rate may be reduced without sacrificing the false accept rate, and the system may then be able to operate in the identification mode. Currently, we are investigating a decision-fusion schema to integrate fingerprint and face.

The expected error rate of a deployed biometric system is usually a very small number ($\ll 1\%$). To estimate such a small number reliably and accurately, large representative data sets that satisfy the two requirements mentioned in Section I are needed. Generally, under the assumption of statistical independence, the number of tests conducted should be larger than ten divided by the error rate [24]. Currently, we are evaluating the system on a large data set of live-scan fingerprint images.

REFERENCES

[1] N. Ansari, M. H. Chen, and E. S. H. Hou, "A genetic algorithm for point pattern matching," in *Dynamic, Genetic, and Chaotic Programming*, B. Souĉek and the IRIS Group, Eds. New York: Wiley, 1992, ch. 13.

[2] "American national standard for information systems—Data format for the interchange of fingerprint information," American National Standards Institute, New York, NY, Doc. No. ANSI/NIST-CSL 1-1993.

[3] J. Atick, P. Griffin, and A. Redlich, "Statistical approach to shape from shading: Reconstruction of 3D face surfaces from single 2D images," *Neural Computation*, to be published.

[4] R. Bahuguna, "Fingerprint verification using hologram matched filterings," in *Proc. Biometric Consortium Eighth Meeting*, San Jose, CA, June 1996.

[5] H. Baird, *Model Based Image Matching Using Location*. Cambridge, MA: MIT Press, 1984.

[6] D. H. Ballard, "Generalized Hough transform to detect arbitrary patterns," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-3, no. 2, pp. 111–122, 1981.

[7] MFLOPS Benchmark Results. (Feb. 1997.) [Online]. Available: ftp://ftp.nosc.mil/pub/aburto/_flops_1.tbl.

[8] T. Biggs, personal communication, Department of Immigration and Naturalization Services, 1997.

[9] F. Bouchier, J. S. Ahrens, and G. Wells. (1996). Laboratory evaluation of the iriscan prototype biometric identifier. [Online]. Available: http://infoserve.library.sandia.gov/sand_doc/1996/961033.pdf.

[10] J. P. Campbell, Jr., L. A. Alyea, and J. S. Dunn. (1996). Biometric security: Government applications and operations. [Online]. Available: http://www.vitro.bloomington.in.us:8080/~BC/.

[11] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[12] 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," National Institute of Standards and Technology, Gaithersburg, MD, NIST Tech. Rep. NISTIR 5647, Aug. 1995.

[13] R. Clarke, "Human identification in information systems: Management challenges and public policy issues," *Info. Technol. People*, vol. 7, no. 4, pp. 6–37, 1994.

[14] L. Coetzee and E. C. Botha, "Fingerprint recognition in low quality images," *Pattern Recognit.*, vol. 26, no. 10, pp. 1441–1460, 1993.

[15] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*. New York: McGraw-Hill, 1990.

[16] H. Cummins and C. Midlo, *Finger Prints, Palms and Soles*. New York: Dover, 1961.

[17] P. E. Danielsson and Q. Z. Ye, "Rotation-invariant operators applied to enhancement of fingerprints," in *Proc. 8th ICPR*, Rome, Italy, 1988, pp. 329–333.

[18] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 11, pp. 1148–1161, 1993.

[19] J. G. Daugman and G. O. Williams, "A proposed standard for biometric decidability," in *Proc. CardTech/SecureTech Conf.*, Atlanta, GA, 1996, pp. 223–234.
[20] S. G. Davies, "Touching Big Brother: How biometric technology will fuse flesh and machine," *Info. Technol. People*, vol. 7, no. 4, pp. 60–69, 1994.
[21] Edge lit hologram for livescan fingerprinting. (1997). [Online]. Available: http://eastview.org/ImEdge.
[22] M. Eshera and K. S. Fu, "A graph distance measure for image analysis," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, no. 3, 1984.
[23] M. Eshera and K. S. Fu, "A similarity measure between attributed relational graphs for image analysis," in *Proc. 7th Int. Conf. Pattern Recognition*, Montreal, Canada, July 30-Aug. 3, 1984.
[24] V. Fabian and J. Hannan, *Introduction to Probability and Mathematical Statistics*. New York: Wiley, 1985.
[25] Federal Bureau of Investigation, *The Science of Fingerprints: Classification and Uses*. Washington, D.C.: GPO, 1984.
[26] K. S. Fu, *Syntactic Pattern Recognition and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1982.
[27] S. Gold and A. Rangarajan, "A graduated assignment algorithm for graph matching," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, no. 4, pp. 377–388, 1996.
[28] M. Hartman, "Compact fingerprint scanner techniques," in *Proc. Biometric Consortium 8th Meeting*, San Jose, CA, June 1996.
[29] L. Hong, A. K. Jain, S. Pankanti, and R. Bolle, "Fingerprint enhancement," in *Proc. IEEE Workshop on Applications of Computer Vision*, Sarasota, FL, 1996, pp. 202–207.
[30] D. C. D. Hung, "Enhancement and feature purification of fingerprint images," *Pattern Recognit.*, vol. 26, no. 11, pp. 1661–1671, 1993.
[31] S. Hunt, "National ID programs around the world," in *Proc. CardTech/SecurTech, Vol. II: Applications*, Atlanta, GA, May 1996, pp. 509–520.
[32] D. P. Huttenlocher and S. Ullman, "Object recognition using alignment," in *Proc. 1st IEEE Int. Conf. Computer Vision*, London, U.K., 1987, pp. 102–111.
[33] A. Jain and L. Hong, "On-line fingerprint verification," in *Proc. 13th ICPR*, Vienna, Austria, 1996, pp. 596–600.
[34] A. K. Hrechak and J. A. McHugh, "Automated fingerprint recognition using structural matching," *Pattern Recognit.*, vol. 23, no. 8, 1990.
[35] INS passenger accelerated service system (INSPASS). (1996). [Online]. Available: http://www.vitro.bloomington.in.us:8080/~BC/REPORTS/INSPASS.html.
[36] D. K. Isenor and S. G. Zaky, "Fingerprint identification using graph matching," *Pattern Recognit.*, vol. 19, no. 2, 1986.
[37] A. Jain, L. Hong, and R. Bolle, "On-line fingerprint verification," IEEE *Trans. Pattern Anal. Machine Intell.*, vol. 19, no. 4, pp. 302–314, 1997.
[38] M. Kass and A. Witkin, "Analyzing oriented patterns," *Comput. Vision, Graph. Image Processing*, vol. 37, no. 4, pp. 362–385, 1987.
[39] K. Karu and A. K. Jain, "Fingerprint classification," *Pattern Recognit.*, vol. 29, no. 3, pp. 389–404, 1996.
[40] M. Kawagoe and A. Tojo, "Fingerprint pattern classification," *Pattern Recognit.*, vol. 17, no. 3, pp. 295–303, 1984.
[41] J. Klett, "Thermal imaging fingerprint technology," in *Proc. Biometric Consortium 9th Meeting*, Crystal City, VA, Apr. 1997.
[42] H. C. Lee and R. E. Gaensslen, Eds., *Advances in Fingerprint Technology*. New York: Elsevier, 1991.
[43] Z. R. Li and D. P. Zhang, "A fingerprint recognition system with micro-computer," in *Proc. 6th ICPR*, Montreal, Canada, 1984, pp. 939–941.
[44] F. R. Livingstone, L. King, J.-A. Beraldin, and M. Rioux, "Development of a real-time laser scanning system for object recognition, inspection, and robot control," in *Proc. SPIE Telemanipulator Technology and Space Telerobotics*, Boston, MA, Sept. 1993, vol. 2057, pp. 454–461.
[45] D. Maio, D. Maltoni, and S. Rizzi, "An efficient approach to online fingerprint verification," in *Proc. 8th Int. Symposium on AI*, Monterrey, Mexico, Oct. 1995, pp. 132–138.
[46] ____, "A structural approach to fingerprint classification," in *Proc. 13th ICPR*, Vienna, Austria, 1996, pp. 578–585.
[47] K. McCalley, D. Setlak, S. Wilson, and J. Schmitt, "A direct fingerprint reader," in *Proc. CardTech/SecurTech, Volume I: Technology*, Atlanta, GA, May 1996, pp. 271–279.
[48] B. M. Mehtre, N. N. Murthy, and S. Kapoor, "Segmentation of fingerprint images using the directional image," *Pattern Recognit.*, vol. 20, no. 4, pp. 429–435, 1987.
[49] D. Mintie, "Welfare ID at the point of transaction using fingerprint and 2D bar codes," in *Proc. CardTech/SecurTech, Volume II: Applications*, Atlanta, GA, May 1996, pp. 469–476.
[50] Access control applications using optical computing. (1997). [Online]. Available: http://www.mytec.com/.
[51] P. J. Phillips, P. J. Rauss, and S. Z. Der, "The FERET (Face Recognition Technology) evaluation methodology," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition 97*, San Juan, Puerto Rico, June 17–19, 1997, pp. 137–143.
[52] B. Miller, "Vital signs of identity," *IEEE Spectrum*, vol. 31, no. 2, pp. 22–30, 1994.
[53] E. Newham, *The Biometric Report*. New York: SJB Services, 1995. (Available: http://www.sjb.co.uk/.)
[54] L. O'Gorman and J. V. Nickerson, "An approach to fingerprint filter design," *Pattern Recognit.*, vol. 22, no. 1, pp. 29–38, 1989.
[55] A. Ranade and A. Rosenfeld, "Point pattern matching by relaxation," *Pattern Recognit.*, vol. 12, no. 2, pp. 269–275, 1993.
[56] A. R. Rao, *A Taxonomy for Texture Description and Identification*. New York: Springer-Verlag, 1990.
[57] K. Rao and K. Balk, "Type classification of fingerprints: A syntactic approach," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-2, no. 3, pp. 223–231, 1980.
[58] N. Ratha, S. Chen, and A. K. Jain, "Adaptive flow orientation based feature extraction in fingerprint images," *Pattern Recognit.*, vol. 28, no. 11, pp. 1657–1672, 1995.
[59] N. Ratha, K. Karu, S. Chen, and A. K. Jain, "A real-time matching system for large fingerprint database," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, no. 8, pp. 799–813, 1996.
[60] Scanner specifications. (1997). [Online]. Available: ftp://ard.fbi.gov/pub/IQS/spec/.
[61] J. Schneider, "Improved image quality of live scan fingerprint scanners using acoustic backscatter measurements," in *Proc. Biometric Consortium 8th Meeting*, San Jose, CA, June 1996.
[62] S. Sclaroff and A. P. Pentland, "Modal matching for correspondence and recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, no. 6, pp. 545–561, 1995.
[63] G. Scott and C. Longuet-Higgins, "An algorithm for associating the features of two images," *Proc. Royal Society of London*, vol. 244, pp. 21–26, 1991.
[64] D. B. G. Sherlock, D. M. Monro, and K. Millard, "Fingerprint enhancement by directional Fourier filtering," *Proc. Inst. Elect. Eng. Visual Image Signal Processing*, vol. 141, no. 2, pp. 87–94, 1994.
[65] A. Sherstinsky and R. W. Picard, "Restoration and enhancement of fingerprint images using M-lattice a novel nonlinear dynamical system," in *Proc. 12th ICPR-B*, Jerusalem, Israel, 1994, pp. 195–200.
[66] J. P. P. Starink and E. Backer, "Finding point correspondence using simulated annealing," *Pattern Recognit.*, vol. 28, no. 2, pp. 231–240, 1995.
[67] G. Stockman, S. Kopstein, and S. Benett, "Matching images to models for registration and object detection via clustering," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 4, no. 3, pp. 229–241, 1982.
[68] Technology recognition systems homepage. (1997). [Online]. Available: http://www.betac.com/~imagemap/subs?155,150.
[69] J. Ton and A. K. Jain, "Registering landsat images by point matching," *IEEE Trans. Geosci. Remote Sensing*, vol. 27, no. 5, pp. 642–651, 1989.
[70] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.
[71] V. V. Vinod and S. Ghose, "Point matching using asymmetric neural networks," *Pattern Recognit.*, vol. 26, no. 8, pp. 1207–1214, 1993.
[72] C. I. Watson, "NIST special database 9," *Mated Fingerprint Card Pairs*, National Institute of Standards and Technology, Gaithersburg, MD, May 1993.
[73] J. H. Wegstein, *An Automated Fingerprint Identification System*, National Bureau of Standards Special Publication 500-89, republished by National Technical Information Service, U.S. Dept. Commerce, Springfield, VA, 1982.
[74] C. L. Wilson, G. T. Candela, and C. I. Watson, "Neural-network fingerprint classification," *J. Artificial Neural Networks*, vol. 1, no. 2, pp. 203–228, 1994.

[75] Q. Xiao and Z. Bian, "An approach to fingerprint identification by using the attributes of feature lines of fingerprint," in *Proc. 7th ICPR*, Paris, France, 1986, pp. 663–665.

Anil K. Jain (Fellow, IEEE) is a University Distinguished Professor and Chair of the Department of Computer Science, Michigan State University, East Lansing. His research interests include statistical pattern recognition, Markov random fields, texture analysis, neural networks, fingerprint matching, document image analysis, and three-dimensional object recognition. He was Editor-in-Chief of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and currently is an Associate Editor of a number of journals. He is coauthor of *Algorithms for Clustering Data* (Englewood Cliffs, NJ: Prentice-Hall, 1988), editor of *Real-Time Object Measurement and Classification* (New York: Springer-Verlag, 1988), and coeditor of *Analysis and Interpretation of Range Images* (New York: Springer-Verlag, 1989), *Markov Random Fields* (New York: Academic, 1992), *Artificial Neural Networks and Pattern Recognition* (Amsterdam: Elsevier, 1993), and *3D Object Recognition* (Amsterdam: Elsevier, 1993).

Dr. Jain received the best paper awards from the Pattern Recognition Society in 1987 and 1991. He received the 1996 IEEE TRANSACTIONS ON NEURAL NETWORKS Outstanding Paper Award. He received a Fulbright research fellowship.

Lin Hong received the B.S. and M.S. degrees in computer science from Sichuan University, China, in 1987 and 1990, respectively. He currently is pursuing the Ph.D. degree in the Department of Computer Science, Michigan State University, East Lansing.

His current research interests include pattern recognition, image processing, multimedia, biometrics, computer graphics, and computer vision application.

Sharath Pankanti (Associate Member, IEEE) is with the Exploratory Computer Vision and Intelligent Robotics group at the IBM T. J. Watson Research Center, Yorktown Heights, NY. He works on the Advanced Identification Solutions project dealing with reliable and scalable identification systems. His research interests include biometrics, pattern recognition, computer vision, and human perception.

Ruud Bolle (Fellow, IEEE) received the B.S. degree in analog electronics and the M.S. degree in electrical engineering from Delft University of Technology, The Netherlands, in 1977 and 1980, respectively. He received the M.S. degree in applied mathematics and the Ph.D. degree in electronic engineering from Brown University, Providence, RI, in 1983 and 1984, respectively.

In 1984, he joined the research staff in the Artificial Intelligence Group of the Computer Science Department, IBM T. J. Watson Research Center, Yorktown Heights, NY. In 1988, he became Manager of the newly formed Exploratory Computer Vision Group, which is part of IBM's digital library effort. His current research interests are video data base indexing, video processing, and biometrics. He is Associate Editor of *Computer Vision and Image Understanding* and Guest Editor of a special issue on computer vision applications for network-centric computers.

Dr. Bolle is on the Advisory Council of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.