F

FINGERPRINT IDENTIFICATION

HISTORY OF FINGERPRINTS

Sir William Herschel discovered that fingerprints remain stable over time and distinct across individuals. In 1877, he commenced placing the inked palm and thumb impressions of some members of the local population on contracts. These prints were used as a form of signature on the documents. However, Herschel never claimed that he had developed a method to identify criminals.

In the late 1800s, the most advanced findings in fingerprint study was made by Dr. Henry Faulds. He found that fingerprints will not change even with superficial injury and that the latent prints left on objects can be used to identify criminals. In 1892, Sir Francis Galton published an accurate and in-depth study of fingerprint science in a book called *Finger Prints,* in which he described an attempt at a fingerprint classification system to facilitate the handling of large collections of fingerprints. Although the work of Galton proved to be sound and became the foundation of modern fingerprint science, his approach to classification was inadequate. Juan Vucetich, an Argentinian police officer who corresponded with Galton, devised his own fingerprint classification system, which was put into practice in September 1891. In 1897, Sir Edward Henry established the famous *Henry System*(2), which is a systematic and effective method of classifying fingerprints. He published the book *Classification and Uses of Fingerprints* in 1900. About 10 years later, his classification system was being used widely by police forces and prison authorities in the English-speaking world.

Since the early 1960s, researchers have begun to develop an *automatic* fingerprint identification system (AFIS) to improve the efficiency of fingerprint recognition. Today, almost all law enforcement agencies around the world use an AFIS. And fingerprint science is a well-researched field with research and development activities worldwide. Several publicly available databases (3,4) exist for evaluating the performance of various fingerprint recognition algorithms. New high-resolution electronic sensors, which are quite affordable (5,6), are available for use in portable laptop computers, mobile phones, and personal digital assistants (PDAs).

FINGERPRINT FEATURES

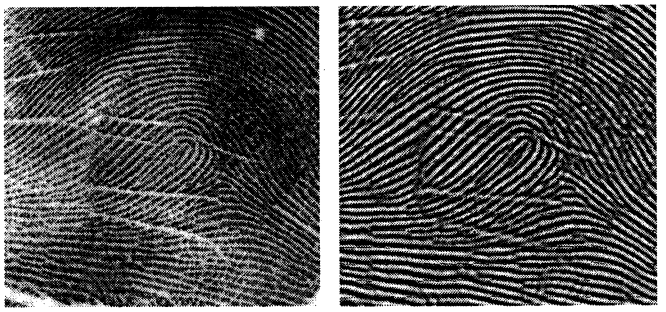
Fingerprints are represented by features that are classified at three levels.

- Level 1 features describe the patterns of the fingerprints, which include ridge flow, core and delta, and pattern type. Ridge flow is the orientation image of the fingerprint. This feature commonly is used for classification. Figure 1 shows an example of the original image and the orientation image. Core and delta are singular points that are defined as the points where the orientation field of a fingerprint is discontinuous. Core is the topmost point on the innermost recurving ridge, and delta is the center of a triangular region where flows from three different directions meet. Figure 2 is an example of the core and delta in a fingerprint. Fingerprints generally are classified into five classes: right loop (R), left loop (L), whorl (W), arch (A), and tented arch (T). Figure 3 shows examples of the fingerprints from these five classes.
- Level 2 features are the points of the fingerprints. They include minutiae, scar, crease, and dot (7). A fingerprint consists of white and dark curves. The white curves are called the *valley* and the dark curves are called the *ridge*. Minutiae features are the ridge characteristics that correspond to the crossings and endings of ridges. They include endpoint, bifurcation, forks, island, and enclosures. The endpoint and bifurcation are used commonly in fingerprint recognition. Figure 4 is an example of the endpoint and bifurcation. Scar is the crossing of two or more adjacent ridges. Figure 5 (a) shows an example of a scar. A crease appears as a white line in a fingerprint. It is a linear depression (or grooves) in the skin. Figure 5 (b) shows an example of a crease. A dot is an isolated ridge unit with a pore on it. Figure 5 (c) shows an example of a dot.
- Level 3 features (5) describe the fingerprint shape that refers to pores and ridge contours. Pores are small openings on ridges. We need a high resolution sensor ($\geq$1000 pixels per inch (ppi)) to get this feature. Figure 6 is an example of sweat pores. Ridge contours are morphological features that include ridge width, shape, path deviation, and so forth.

Fingerprint sensors, the very front end of the fingerprint recognition systems, are used to capture the fingerprint images. The kinds of fingerprint sensors are: optical sensors, semiconductor sensors, and ultrasound sensors. Among these sensors, optical sensors are considered to be stable and reliable, semiconductor sensors are considered to be low cost and portable, and ultrasound sensors are considered to be accurate but more expensive.

FINGERPRINT RECOGNITION

Depending on an application two kinds of fingerprint recognition systems exist: verification systems and identification systems (8). A verification system generally stores the fingerprint images or feature sets of users in a database. At a future time, it compares the fingerprint of a person with her/his own fingerprint image or feature set to verify that this person is, indeed, who she/he claims to be. This problem is a one-to-one matching problem. The system can accept or reject this person, according to the verification



(a) (b)

Figure 1. An example of the original image and the orientation image: (a) original image, (b) orientation image. The orientation is shown by the arrows above the ridges.

result. An identification system is more complex where, for a query fingerprint, the system searches the entire database to find out if any fingerprint images or feature sets saved in the database can match it. It conducts one-to-many matching (8). Two kinds of identification systems exists: the closed-set identification system and the open-set identification system (9). The closed-set identification system is the identification system for which all potential users are enrolled in the system. Usually, the closed-set identification is used for research purposes. The open-set identification system is the identification system for which some potential users are not enrolled in the system. The open-set identification is performed in real operational systems. The verification and the closed-set identification are special cases of the open-set identification.

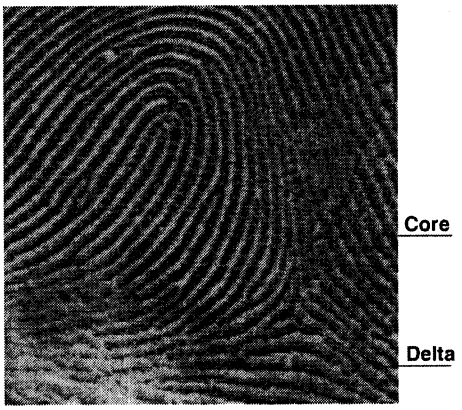


Figure 2. Level 1 features: core and delta.

Three kinds of approaches (see Fig. 7) exist to solve the fingerprint identification problem (10): (1) Repeat the verification procedure for each fingerprint in the database and select the best match; (2) use fingerprint classification followed by verification; and (3) use fingerprint indexing followed by verification (10,11). Fingerprint matching, classification, and indexing are three basic problems in fingerprint identification.

Fingerprint Matching

A fingerprint matching algorithm aligns the two given fingerprints, finds the correspondences between them, and returns a measure of the degree of similarity. Usually, the similarity score is used to represent the degree of similarity between the two given fingerprints. Fingerprint matching is a challenging problem because different impressions of the same finger could be very different because of distortion, displacement, rotation, noise, skin condition, pressure, noise, and so forth (8). Furthermore the impressions from different fingers could be quite similar. Figure 8 shows two impressions of one fingerprint from the NIST-4 database (10). The fingerprint matching algorithms can be classified into three different types: (1) the correlation-based approach, (2) the minutiae-based approach, and (3) the ridge feature-based approach.

1. *Correlation-based matching:* The correlation-based approach uses the gray-level information of fingerprints. For a template fingerprint (in the database) and a query fingerprint, it computes the sum of the squared differences in gray values of all the pixels to evaluate the diversity of the template fingerprint and the query fingerprint. To deal with the distortion

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

Figure 3. Examples of fingerprints for each class based on the *Henry System:* (a) right loop, (b) left loop, (c) whorl, (d) arch, and (e) tented.

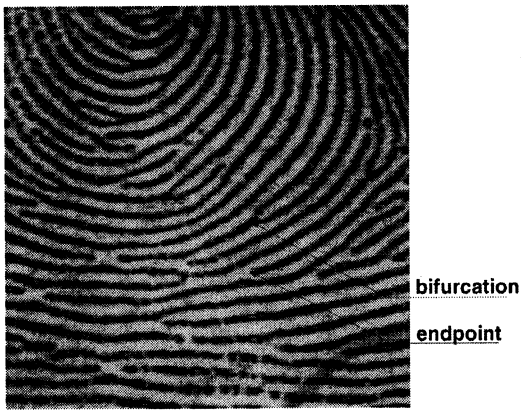


Figure 4. Minutiae: endpoint and bifurcation.

problem, it computes the correlation in local regions instead of the global correlation on the entire image. In Bazen et al. (12), the correlation-based evaluation is used to find the distinctive regions of the template fingerprint. These local regions fit very well at the original locations and much worse at other locations. During the matching, they compute the gray-level distance between the distinctive regions of a template and the corresponding areas in the query fingerprints. Then, they sum up the squared gray-level difference for each local region. The position of a region of the template with the minimal distance is considered as the corresponding region in the query fingerprint.

Compared with the other matching algorithms described below, correlation-based matching approaches use gray-level information of the fingerprint. When the quality of the fingerprint image is not good, especially when a large number of minutiae are missing, the correlation-based matching algorithm may be considered. However, it is expensive computationally.

2. *Minutiae-based matching:* Minutiae-based matching is the most commonly used method for fingerprint recognition systems. In this approach, a fingerprint is represented by a set of minutiae features. Thus, the fingerprint recognition problem is reduced to a point-matching problem. Therefore, any point matching approach, such as the relaxation algorithms, can be used to recognize the fingerprints (13,14).

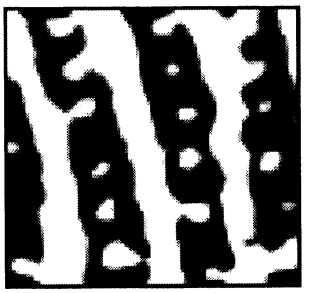
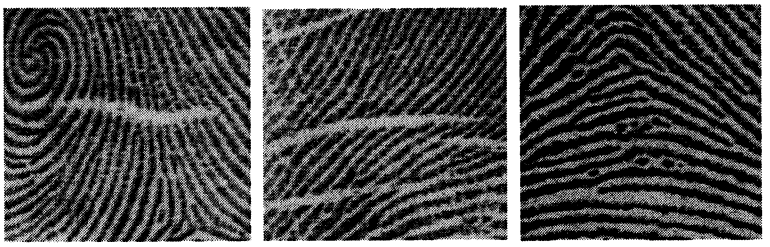


Figure 6. Example of sweat pores.

- Feature extraction: The first step of the minutiae-matching algorithm is the minutiae extraction. Figure 9 is the block diagram of the minutiae-based feature extraction procedure, which is used widely in most fingerprint recognition systems. As an example, Bhanu and Tan present a learned template-based algorithm for feature extraction (15). Templates are learned from examples by optimizing a criterion function using the Lagrange method. To detect the presence of minutiae in fingerprints, templates for endpoints and bifurcations are applied with appropriate orientation to the binary fingerprints at selected potential minutiae locations.
- Matching: Tan and Bhanu (13,14) present a fingerprint-matching approach, based on genetic algorithms (GA). This method can achieve a globally optimized solution for the transformation between two sets of minutiae extracted from two different fingerprints. In their approach, the fitness function is based on the local properties of triplets of minutiae, such as minimum angle, maximum angle, triangle handedness, triangle direction, maximum side, minutiae density, and ridge counts. These features are described in the "Fingerprint Indexing" section below.

Jiang and Yau (16) use both the local and global structures of minutiae in their minutiae-matching approach. The local structure of a minutia describes the features independent of the rotation and translation in its l-nearest neighborhood. The global structure is variant with the rotation and translation. Using the local structure, the best matched minutiae pair is found and used to align the template and query fingerprint. Then, the elastic bounding box of the global features is used for the fingerprint matching.



(a) (b) (c)

Figure 5. Level 2 features: (a) scar, (b) crease, and (c) dot.

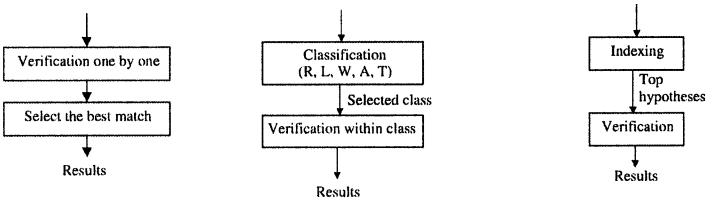


Figure 7. Block diagram of three kinds of approaches to solve the identification problem. (1) Repeat verification (2) Classification followed by verification (3) Indexing followed by verification

Figure 8. Two impressions of one fingerprint.

Kovacs-Vajna (17) used triangular matching to deal with deformations of fingerprints. In this approach, the minutiae regions of the template fingerprint are moved around the query fingerprint to find the possible correspondence. The triangular matching algorithm is used to obtain the matching minutiae set. Then, the dynamic time-warping algorithm is applied to validate the final matching results.

3. *Ridge feature-based matching:* For a good quality fingerprint with size 480 × 512 [500 pixels per inch (ppi)], about 80 minutiae features could exist. Thus, for the triangular minutiae matching, hundreds of thousands of triangles could exist. So, the minutiae matching approach needs high computational power. However, when the image quality is not good, the minutiae extraction would be difficult. Because fingerprints consist of natural valley and ridges, researchers have used them for fingerprint matching. Maltoni et al. (8) present a filter-based algorithm for fingerprint recognition. It is based on the grayscale image. First, they determine a reference point with the maximum curvature of the concave ridges and an interest region in the fingerprint. After tessellating the interest region around the reference point, they use a bank of Gabor filters to capture both local and global details in a fingerprint. Then, they compute the average absolute deviation from the mean to define the compact fixed-length FingerCode as the feature vector. Finally, they match the fingerprint by computing the Euclidean distance between the corresponding FingerCode between the template and query fingerprints.

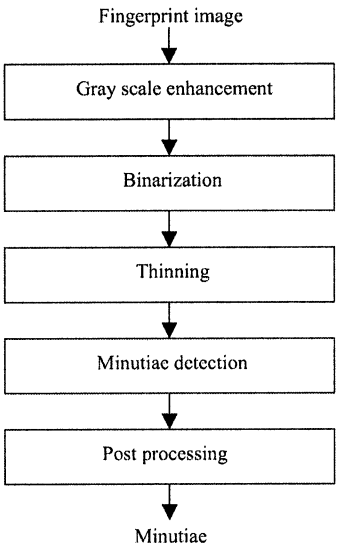


Figure 9. Block diagram for minutiae-based feature extraction.

With the improvement of the fingerprint sensor technology, now it is possible to extract features at a high resolution. In Jain et al. (5), authors use pores and ridge contours combined with minutiae features to improve fingerprint recognition performance. In their approach, they use Gabor filter and wavelet transform to extract pores and ridge contours. During the matching process, they extract orientation field and minutiae features and establish alignment between the template and query fingerprint. If the orientation fields match, then the system uses a minutiae-based matching algorithm to verify the query fingerprint or to reject the query fingerprint. If the number of the corresponding minutiae between the template fingerprint and query fingerprint is greater than a threshold, then these two fingerprints match; if not, then the system extracts pores and ridge contours and they use the Iterative Closest Point (ICP) algorithm to match these features. This hierarchical matching system requires 1000 ppi resolution for the sensor.

FINGERPRINT CLASSIFICATION

Most fingerprint classification systems use the Henry system for fingerprint classification, which has five classes as shown in Fig. 3. The most widely-used approaches for fingerprint classification are based on the number and relations of the singular points, including the core and

the delta. Karu and Jain (8) present a classification approach based on the structural information around the singular points. Three steps are in this algorithm: (1) Compute the ridge direction in a fingerprint image; (2) find the singular points based on the changes in the directional angle around the curve; and (3) classify the fingerprints according to the number and locations of the core and delta. Other researchers use a similar method: first, find the singular point; then use a classification algorithm to find the difference in areas, which are around the singular points for different classes. Several representations based on principal component analysis (PCA) (3), a self-organizing map (18), and Gabor filters (8) are used. The problems with these approaches are:

- It is not easy to detect singular points, and some fingerprints do not have singular points.
- Uncertainty in the location of the singular points is large, which has a great effect on the classification performance because the features around the singular points are used.

Cappelli et al. (19) present a structural analysis or the orientation field of a fingerprint. In their approach, the directional image is calculated and enhanced. A set of dynamic masks is used in the segmentation step, and each dynamic mask is adapted independently to best fit the directional image according to a cost function. The resulting cost constitutes a basis for the final classification (3). Based on the orientation field, cappelli et al. also present a fingerprint classification system based on the multispace KL transform (20). It uses a different number of principal components for different classes. Jain and Minut (8) propose a classification algorithm based on finding the kernel that best fits the flow field of a given fingerprint. For each class, a kernel is used to define the shape of the fingerprint in that class. In these approaches, it is not necessary to find the singular points.

Researchers also have tried different methods to combine different classifiers to improve the classification performance. Senior (21) combines the hidden Markov model (HMM), decision trees, and PCASYS (3). Yao et al. (22) present new fingerprint classification algorithms based on two machine learning approaches: support vector machines (SVMs) and recursive neural networks (RNNs). In their approach, the fingerprints are represented by the relational graphs. Then, RNNs are used to train these graphs and extract distributed features for the fingerprints. SVMs integrated with distributed features are used for classification. To solve the ambiguity problem in fingerprint classification, an error-correcting code scheme is combined with SVMs.

Tan et al. (23) present a fingerprint classification approach based on genetic programming (GP) to learn composite operators that help to find useful features. During the training, they use GP to generate composite operators. Then, the composite operators are used to generate the feature vectors for fingerprint classification. A Bayesian classifier is used for classification. Fitness values are computed for the composite operators based on the classification results. During the testing, the learned composite operator is applied directly to generate feature vectors. In their approach, they do not need to find the reference points. Table 1 summarizes representative fingerprint classification approaches.

FINGERPRINT INDEXING

The purpose of indexing algorithms is to generate, in an efficient manner, a set of hypotheses that is a potential match to a query fingerprint. Indexing techniques can be considered as front-end processing, which then would be followed by back-end verification processing in a complete fingerprint recognition system.

A prominent approach for fingerprint indexing is by Germain et al. (11). They use the triplets of minutiae in their indexing procedure. The features they use are: the length of each side, the ridge count between each pair of vertices, and the angles that the ridges make with respect to the x-axis of the reference frame. The number of corresponding triangles is defined as the similarity score between the query and the template fingerprints. In their approach, a hash table is built where all possible triplets are

Table 1. Representative fingerprint classification approaches

Authors	Approach
Candela et al. (3), 1995	Probabilistic neural network (PNN)
Karu and Jain (24), 1996	Rule-based classification
Halici and Ongun (18), 1996	Neural network based on self-organizing feature maps (SOM)
Cappelli et al. (19), 1997	Multispace principal component analysis
Qi et al. (25), 1998	Probabilistic neural network based on genetic algorithm (GA) and feedback mechanism
Jain et al. (28), 1999	K-nearest neighbor and neural network based on Gabor features (FingerCode)
Cappelli et al. (26), 1999	Classification based on partitioning of orientation image
Kamijo (27), 1999	A four-layered neural network integrated in a two-step learning method
Su et al. (28), 2000	Fractal analysis
Pattichis et al. (29), 2001	Probabilistic neural network and AM–FM representation for fingerprints
Bernard et al. (30), 2001	Kohonen topological map
Senior (21), 2001	Hidden Markov model and decision tree and PCASYS
Jain and Minut (31), 2002	Model-based method based on hierarchical kernel fitting
Mohamed and Nyongesa (32), 2002	Fuzzy neural network
Yao et al. (22), 2003	Support vector machine and recursive neural network based on FingerCode
Tan et al. (23), 2005	Genetic programming

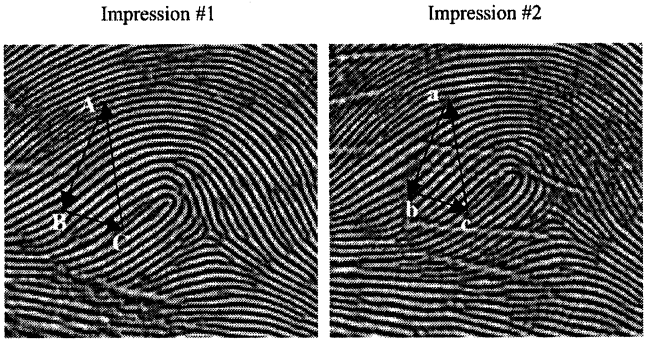


Figure 10. An example of two corresponding triangles in a pair of fingerprints.

saved. For each triplet, a list of IDs, including the fingerprints that have this triplet, is saved. During the identification process, the triplets of the query fingerprint are extracted and—by a hashing process described below—the potential IDs of the query fingerprint are determined.

Because some features in Ref. (11) may not be reliable, Bhanu and Tan (33) use a novel set of features of a triplet for fingerprint indexing. These features are:

- Minimum angle α_{min} and median angle α_{med}. Assume α_i are three angles in a triplet, where $i = 1, 2, 3$. $\alpha_{min} = min\{\alpha_i\}$, $\alpha_{max} = max\{\alpha_i\}$, $\alpha_{med} = 180° - \alpha_{min} - \alpha_{max}$.
- Triangle handedness ϕ. Let $Z_i = x_i + jy_i$ be the complex number corresponding to the location (x_i, y_i) of point P_i, $i = 1,2,3$. Define $Z_{21} = Z_2 - Z_1$, $Z_{32} = Z_3 - Z_2$, and $Z_{13} = Z_1 - Z_3$. Let triangle handedness $\phi = sign(Z_{21} \times Z_{32})$, where *sign* is the signum function and $\times$ is the cross product of two complex numbers. Points P_1, P_2, and P_3 are noncolinear points, so $\phi = 1$ or -1.
- Triangle direction η. Search the minutiae in the image from top to bottom and left to right. If the minutiae is the start point, then $\upsilon = 1$; otherwise $\upsilon = 0$. Let $\eta = 4\upsilon_1 + 2\upsilon_2 + \upsilon_3$, where υ_i is the υ value of point P_i, $i = 1,2,3$, and $0 \leq \eta \leq 7$.
- Maximum side λ. Let $\lambda = max\{L_i\}$, where $L_1 = |Z_{21}|, L_2 = |Z_{32}|$, and $L_3 = |Z_{13}|$.
- Minutiae density χ. In a local area (32 × 32 pixels) centered at the minutiae P_i, if χ_i minutiae exists then the minutiae density for P_i is χ_i. Minutiae density χ is a vector consisting of all χ_i.
- Ridge counts ξ. Let ξ_1, ξ_2, and ξ_3 be the ridge counts of sides P_1P_2, P_2P_3, and P_3P_1, respectively. Then, ξ is a vector consisting of all ξ_i.

During the offline hashing process, the above features for each template fingerprint (33) are computed and a hash table $H(\alpha_{min}, \alpha_{med}, \phi, \eta, \lambda, \chi, \xi)$ is generated. During the online hashing process, the same features are computed for each query fingerprint and compared with the features represented by H. If the difference in features is small enough, then the query fingerprint is probably the same as the "stored" fingerprints that have similar features. Figure 10 is an example of two corresponding triangles in a pair of fingerprints that are two impressions of one fingerprint. In the first impression, three noncolinear minutiae A, B, and C are picked randomly to form a triangle ΔABC. The features in this triangle are $\{\alpha_{min} = 30°, \alpha_{med} = 65°, \phi = 1, \eta = 6, \lambda = |AC|, \chi = \{0,0,0\}, \Delta\xi = \{6,5,12\}\}$. Similarly, three noncolinear minutiae a, b, and c in the second impression form Δabc. Its features are $\{\alpha_{min} = 31°, \alpha_{med} = 63°, \phi = 1, \eta = 6, \lambda = |ac|, \chi = \{0,2,0\}, \xi = \{6,5,12\}\}$. If the error between these two triangles are within the error tolerance (34), then these two triangles are considered the corresponding triangles. The output of this process, carried out for all the triplets, is a list of hypotheses, which is sorted in the descending order of the number of potential corresponding triangles. Top T hypotheses are the input to the verification process.

PERFORMANCE EVALUATION

Two classes in the fingerprint recognition systems exist: match and nonmatch. Let s and n denote match and non-match. Assume that x is the similarity score. Then, $f(x \mid s)$ is the probability density function given s is true, and $f(x \mid n)$ is the probability density function given n is true. Figure 11 is an example of these two distributions. For a criterion k, one can define

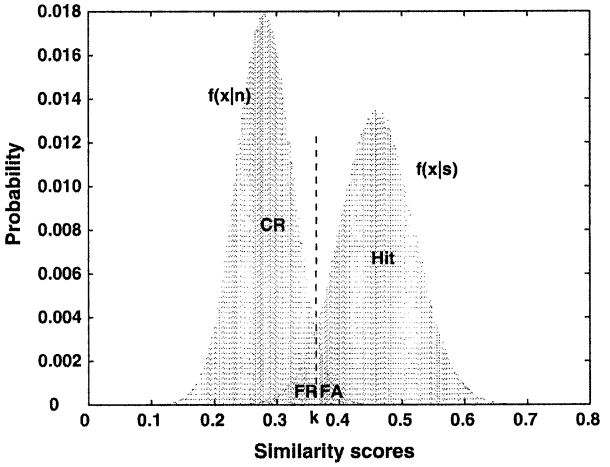


Figure 11. Densities of the match and nonmatch scores.

- Hit: the probability that x is above k given s, where $Hit = \int_k^{\infty} f(x|s)dx$
- False alarm: the probability that x is above k given n, where $FA = \int_k^{\infty} f(x|n)dx$
- False rejection: the probability that x is below k given s, where $FR = \int_{-\infty}^{k} f(x|s)dx$
- Correct rejection: the probability that x is below k given n, where $CR = \int_{-\infty}^{k} f(x|n)dx$

A *receiver operating characteristic (ROC)* curve is a graphical plot whose x-axis is the *false alarm (FA)* rate and y-axis is the *hit (Hit)* rate. A *ROC* curve is used to evaluate the performance of a recognition system because it represents the changes in *FA* rate and *Hit* rate with different discrimination criteria (thresholds). A *detection error tradoff (DET)* curve plots the *FA* rate and the *false rejection (FR)* rate with the change of discrimination criteria. A confidence interval is an interval within which the estimation is likely to be determined. The ISO standard performance testing report gives the detail of recommendations and requirements for the performance evaluations (9).

Public fingerprint databases are used to evaluate the performance of different fingerprint recognition algorithms and fingerprint acquisition sensors. The National Institute of Standards and Technology (NIST) provides several special fingerprint databases with different acquisition methods and scenarios. Most images in these databases are rolling fingerprints that are scanned from paper cards. The *NIST* special database 24 is a digital video of live-scan fingerprint data. The detail of the *NIST* special databases can be found in Ref. (35). Since 2000, fingerprint verification competition (FVC) has provided public databases for a competition that is held every 2 years. For each competition, four disjoint databases are created that are collected with different sensors and technologies. The performance of different recognition algorithms is addressed in the reports of each competition (36–39). The fingerprint vendor technology evaluation (F_pVTE) 2003 is conducted by *NIST* to evaluate the performance of the fingerprint recognition systems. A total of 18 companies competed in the *FpVTE*, and 34 systems from U.S government were examined. The performance of different recognition algorithms is discussed in Ref. (40).

PERFORMANCE PREDICTION

Several research efforts exist for analyzing the performance of fingerprint recognition. Galton (41) assumes that 24 independent square regions could cover a fingerprint and that he could reconstruct correctly any of the regions with a probability of 1/2 by looking at the surrounding ridges. Accordingly, the Galton formulation of the distinctiveness of a fingerprint is given by $(1/16) \times (1/256) \times (1/2)^{24}$, where 1/16 is the probability of the occurrence of a fingerprint type and 1/256 is the probability of the occurrence of the correct number of ridges entering and exiting each of the 24 regions. Pankanti et al. (8) present a fingerprint individuality model that is based on the analysis of feature space and derive an expression to estimate the probability of false match based on the minutiae between two fingerprints. It measures the amount of information needed to establish correspondence between two fingerprints. Tan and Bhanu (42) present a two-point model and a three-point model to estimate the error rate for the minutiae-based fingerprint recognition. Their approach not only measures the position and orientation of the minutiae but also the relations between different minutiae to find the probability of correspondence between fingerprints. They allow the overlap of uncertainty area of any two minutiae. Tabassi et al. (43) and Wein and Baveja (44) use the fingerprint image quality to predict the performance. They define the quality as an indication of the degree of separation between the match score and non-match score distributions. The farther these two distributions are from each other, the better the system performs.

Predicting large population recognition performance based on a small template database is another important topic for the fingerprint performance characterization. Wang and Bhanu (45) present an integrated model that considers data distortion to predict the fingerprint identification performance on large populations. Learning is incorporated in the prediction process to find the optimal small gallery size. The Chernoff and Chebychev inequalities are used as a guide to obtain the small gallery size given the margin of error and confidence interval. The confidence interval can describe the uncertainty associated with the estimation. This confidence interval gives an interval within which the true algorithm performance for a large population is expected to fall, along with the probability that it is expected to fall there.

FINGERPRINT SECURITY

Traditionally, cryptosystems use secret keys to protect information. Assume that we have two agents, called Alice and Bob. Alice wants to send a message to Bob over the public channel. Eve, the third party, eavesdrops over the public channel and tries to figure out what Alice and Bob are saying to each other. When Alice sends a message to Bob, she uses a secret encryption algorithm to encrypt the message. After Bob gets the encrypted message, he will use a secret decryption algorithm to decrypt the message. The secret keys are used in the encryption and decryption processes. Because the secret keys can be forgotten, lost, and broken, biometric cryptosystems are possible for security.

Uludag et al. (46) propose a cryptographic construct that combines the fuzzy vault with fingerprint minutiae data to protect information. The procedure for constructing the fuzzy vault is like what follows in the example of Alice and Bob. Alice places a secret value k in a vault and locks it using an unordered set A of the polynomial coefficients. She selects a polynomial p of variable x to encode k. Then, she computes the polynomial projections for the elements of A and adds some randomly generated chaff points that do not lie on p to arrive at the final point set R. Bob uses an unordered set B of the polynomial coefficients to unlock

the vault only if B overlaps with A to a great extent. He tries to learn k, that is to find p. By using error-correction coding, he can reconstruct p. Uludag et al. (46) present a curve-based transformed minutia representation in securing the fuzzy fingerprint vault.

We know that the biometrics is permanently related with a user. So if the biometrics is lost, then the biometrics recognition system will be compromised forever. Also, a user can be tracked by cross-matching with all the potential uses of the fingerprint biometrics, such as access to the house, bank account, vehicle, and laptop computer. Ratha et al. (47) presents a solution to overcome these problems in the fingerprint recognition systems. Instead of storing the original biometrics image, authors apply a one-way transformation function to the biometrics. Then, they store the transformed biometrics and the transformation to preserve the privacy. If a biometrics is lost, then it can be restored by applyings a different transformation function. For different applications of the same biometrics, they use different transformation functions to avoid a user being tracked by performing a cross match.

Like other security systems, fingerprint sensors are prone to spoofing by fake fingerprints molded with artificial materials. Parthasaradhi et al. (48) developed an anti spoofing method that is based on the distinctive moisture pattern of live fingers contacting fingerprint sensors. This method uses the physiological process of perspiration to determine the liveness of a fingerprint. First, they extract the gray values along the ridges to form a signal. This process maps a 2 fingerprint image into a signal. Then, they calculate a set of features that are represented by a set of dynamic measurements. Finally, they use a neural network to perform classification (live vs. not live).

CONCLUSIONS

Because of their characteristics, such as distinctiveness, permanence, and collectability, fingerprints have been used widely for recognition for more than 100 years. Fingerprints have three levels of features: pattern, point, and shape. With the improvement of the sensor resolution, more and better fingerprint features can be extracted to improve the fingerprint recognition performance. Three kinds of approaches exist to solve the fingerprint identification problem: (1) Repeat the verification procedure for each fingerprint in the database and select the best match; (2) perform fingerprint classification followed by verification; and (3) create fingerprint indexing, followed by verification. Fingerprint verification is done by matching a query fingerprint with a template. The feature extraction, matching, classification, indexing and performance prediction are the basic problems for fingerprint recognition.

Fingerprint prediction, security, liveness detection, and cancelable biometrics are the important current research problems. The area of biometric cryptosystems, especially the fingerprint cryptosystem, is an upcoming area of interest because the traditional secret key can be forgotten, lost, and broken.

BIBLIOGRAPHY

1. R. Wang and B. Bhanu, Predicting fingerprint biometric performance from a small gallery, *Pattern Recognition Letters*, **28** (1): 40–48, 2007.
2. E. R. Henry, *Classification and Uses of Fingerprints*. George Routledge and Sons, 1900.
3. G. T. Candela, P. J. Grother, C. I. Watson, R. A. Wilkinson, and C. L. Wilson, PCASYS-A pattern-level classification automation system for fingerprints, *NIST Technical Report. NISTIR 5467*, 1995.
4. D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, FVC2000: fingerprint verification competition, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **24** (3): 402–412, 2002.
5. A. K. Jain, Y. Chen, and M. Demirkus, Pores and ridges: High resolution fingerprint matching using level 3 features, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **29** (1): 15–27, 2007.
6. http://authentec.com.
7. Scientific Working Group on Friction Ridge Analysis, Study and Technology (SWGFAST), *Available*: http://fingerprint.nist.gov/standard/cdef f s/Docs/SWGFAST_Memo.pdf.
8. D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*. New York: Springer, 2003.
9. ISO/IEC19795-1, Information Technology-Biometric Performance Testing and Reporting-Part 1: Principles and Framework. ISO/IEC JTC1/SC37 N908, 2006.
10. X. Tan and B. Bhanu, A robust two step approach for fingerprint identification, *Pattern Recognition Letters*, **24** (13): 2127–2134, 2003.
11. R. S. Germain, A. Califano, and S. Colville, Fingerprint matching using transformation parameter clustering, *IEEE Computational Science and Engineering*, **4** (4): 42–49, 1997.
12. A. M. Bazen, G. T. B. Verwaaijen, S. H. Gerez, L. P. J. Veelenturf, and B. J. vander Zwaag, A correlation-based fingerprint verification system, *Proc. IEEE Workshop on Circuits Systems and Signal Processing*, Utrecht, Holland, 2000, pp. 205–213.
13. X. Tan and B. Bhanu, Fingerprint matching by genetic algorithm, *Pattern Recognition*, **39** (3): 465–477, 2006.
14. B. Bhanu and X. Tan, *Computational Algorithms for Fingerprint Recognition*. Kluwer Academic Publishers, 2003.
15. B. Bhanu and X. Tan, Learned templates for feature extraction in fingerprint images, *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Hawaii, 2001, Vol 2, pp. 591–596.
16. X. Jiang and W. Y. Yau, Fingerprint minutiae matching based on the local and global structures, *Proc. IEEE Int. Conf. Pattern Recognition*, Barcelona, Spain, 2000, pp. 1038–1041.
17. Z. M. Kovacs-Vajna, A fingerprint verification system based on triangular matching and dynamic time warping, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **22** (11): 1266–1276, 2000.
18. U. Halici and G. Ongun, Fingerprint classification through self-organizing feature maps modified to treat uncertainties, *Proc. IEEE*, **84** (10): 1497–1512, 1996.

19. R. Cappelli, D. Maio, and D. Maltoni, Fingerprint classification based on multi-space KL, *Proc. Workshop Autom. Identific. Adv. Tech.*, 1999, pp. 117–120.
20. N. K. Ratha and R. Bolle, *Automatic Fingerprint Recognition Systems*. Springer, 2003.
21. A. Senior, A combination fingerprint classifier, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **23** (10): 1165–1174, 2001.
22. Y. Yao, G. L. Marcialis, M. Pontil, P. Frasconi, and F. Roli, Combining flat and structured representations for fingerprint classification with recursive neural networks and support vector machines, *Pattern Recognition*, **36**, (2): 397–406, 2003.
23. X. Tan, B. Bhanu, and Y. Lin, Fingerprint classification based on learned features, *IEEE Trans. on Systems, Man and Cybernetics, Part C, Special issue on Biometrics*, **35**, (3): 287–300, 2005.
24. K. Karu and A. K. Jain, Fingerprint classification, *Pattern Recognition*, **29**, (3): pp. 389–404, 1996.
25. Y. Qi, J. Tian and R. W. Dai, Fingerprint classification system with feedback mechanism based on genetic algorithm, *Proc. Int. Conf. Pattern Recog.*, **1**: 163–165, 1998.
26. R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, Fingerprint classification by directional image partitioning, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **21** (5): 402–421, 1999.
27. M. Kamijo, Classifying fingerprint images using neural network: Deriving the classification state, *Proc. Int. Conf. Neural Network*, **3**: 1932–1937, 1993.
28. F. Su, J. A. Sun, and A. Cai, Fingerprint classification based on fractal analysis, *Proc. Int. Conf. Signal Process.*, **3**: 1471–1474, 2000.
29. M. S. Pattichis, G. Panayi, A. C. Bovik, and S. P. Hsu, Fingerprint classification using an AM-FM model, *IEEE Trans. Image Process.*, **10** (6): 951–954, 2001.
30. S. Bernard, N. Boujemaa, D. Vitale, and C. Bricot, Fingerprint classification using Kohonen topologic map, *Proc. Int. Conf. Image Process.*,**3**: 230–233, 2001.
31. A. K. Jain and S. Minut, Hierarchical kernel fitting for fingerprint classification and alignment, *Proc. IEEE Int. Conf. Pattern Recognition*, **2**: 469–473, 2002.
32. S. M. Mohamed and H. O. Nyongesa, Automatic fingerprint classification system using fuzzy neural techniques, *Proc. Int. Conf. Fuzzy Systems*, **1**: 358–362, 2002.
33. B. Bhanu and X. Tan, Fingerprint indexing based on novel features of minutiae triplets, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **25**, (5): 616–622, 2003.
34. X. Tan and B. Bhanu, Robust fingerprint identification, *Proc. IEEE Int. Conf. on Image Processing*, New York, 2002, pp. 277–280.
35. http://www.itl.nist.gov/iad/894.03/databases/defs/dbases.html.
36. http://bias. csr. unibo. it/fvc2000/.
37. http://bias.csr.unibo.it/fvc2002/.
38. http://bias. csr. unibo. it/fvc2004/.
39. http://bias.csr.unibo.it/fvc2006/.
40. C. Wilson, R. A. Hicklin, M. Bone, H. Korves, P. Grother, B. Ulery, R. Micheals, M. Zoepfl, S. Otto, and C. Watson, *Fingerprint Vendor Technology Evaluation 2003: Summary of Results and Analysis Report*, 2004.
41. F. Galton, *Finger Prints*. McMillan, 1892.
42. X. Tan and B. Bhanu, On the fundamental performance for fingerprint matching, *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Madison, Wisconsin, 2003, pp. 18–20.
43. E. Tabassi, C. L. Wilson, and C. I. Watson, Fingerprint image quality, *National Institute of Standards and Technology International Report 7151*, 2004.
44. L. M. Wein and M. Baveja, Using fingerprint image quality to improve the identification performance of the U.S. visitor and immigrant status indicator technology program, *The National Academy of Sciences*, **102** (21): 7772–7775, 2005.
45. R. Wang and B. Bhanu, Learning models for predicting recognition performance, *Proc. IEEE Int. Conf. on Computer Vision*, Beijing, China, 2005, pp. 1613–1618.
46. U. Uludag, S. Pankanti, and A. K. Jain, Fuzzy vault for fingerprints, *Proc. Audio- and Video-based Biometric Person Authentication*, Rye Brook, New York, 2005, pp. 310–319.
47. N. K. Ratha, S. Chikkerur, J. H. Connell, and R. M. Bolle, Generating cancelable fingerprint templates, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **29** (4): 561–572, 2007.
48. S. T. V. Parthasaradhi, R. Derakhshani, L. A. Hornak, and S. A. C. Schuckers, Time-series detection of perspiration as a liveness test in fingerprint devices, *IEEE Trans. on System, Man, and Cybernetics-Part C: Applications and Reviews*, **35** (3): 335–343, 2005.

RONG WANG
BIR BHANU
University of California
Riverside, California