A By Level Privacy Protection of Fingerprint Combination using Hough Transformer

NEELAVATHI¹, A.RAJENDRA BABU²

¹PG Scholar, Dept of ECE, Bharat College of Engineering & Technology for Women, Kadapa, AP, India.
²Asst Prof, Dept of ECE, Bharat College of Engineering & Technology for Women, Kadapa, AP, India.

Abstract: We propose here a novel system for protecting fingerprint privacy by combining two different fingerprints into a new identity using HOUGH TRANSFORMER technique used in image analysis, computer vision, and digital image processing. The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure. In the enrollment, two fingerprints are captured from two different fingers. We extract the minutiae positions from one fingerprint, the orientation from the other fingerprint, and the reference points from both fingerprints. Based on this extracted information and our proposed coding strategies, a combined minutiae template is generated and stored in a database. In the authentication, the system requires two query fingerprints from the same two fingers which are used in the enrollment. A two-stage fingerprint matching process is proposed for matching the two query fingerprints against a combined minutiae template. By storing the combined minutiae template, the complete minutiae feature of a single fingerprint will not be compromised when the database is stolen. Furthermore, because of the similarity in topology, it is difficult for the attacker to distinguish a combined minutiae template from the original minutiae templates. With the help of an existing fingerprint reconstruction approach, we are able to convert the combined minutiae template into a real-look like combined fingerprint. Thus, a new virtual identity is created for the two different fingerprints, which can be matched using minutiae-based fingerprint matching algorithms. The experimental results show that our system can achieve a very low error rate with FRR = 0.4% at FAR = 0.1%. Compared with the state-of-the-art technique, our work has the advantage in creating a better new virtual identity when the two different fingerprints are randomly chosen.

Keywords: Combination, Fingerprint, Minutiae, Privacy, and Protection.

I. INTRODUCTION

With the widespread applications of fingerprint techniques in authentication systems, protecting the privacy of the fingerprint becomes an important issue. Traditional encryption is not sufficient for fingerprint privacy protection because decryption is required before the fingerprint matching, which exposes the fingerprint to the attacker. Therefore, in recent years, significant efforts have been put into developing specific protection techniques for fingerprint. Most of the existing techniques make use of the key for the fingerprint privacy protection, which creates the inconvenience. They may also be vulnerable when both the key and the protected fingerprint are stolen. Teoh et al. [3] propose a biohashing approach by computing the inner products between the user’s fingerprint features and a pseudorandom number (i.e., the key). The accuracy of this approach mainly depends on the key, which is assumed to be never stolen or shared [4]. Ratha et al. [5] propose to generate cancelable fingerprint templates by applying noninvertible transforms on the minutiae. The noninvertible transform is guided by a key, which will usually lead to a reduction in matching accuracy. The work in [3] and [5] are shown to be vulnerable to intrusion and linkage attacks when both the key and the transformed template are stolen [6]. Nandakumar et al. [7] propose to implement fuzzy fault on the minutiae, which is vulnerable to the key-inversion attack [8]. Our work in [9] imperceptibly hides the user identity on the thinned fingerprint using a key. The user identity may also be compromised when both the key and the protected thinned fingerprint are stolen. There are only a few schemes [10] that are able to protect the privacy of the fingerprint without using a key. Ross and Othman [10] propose to use visual cryptography for protecting the privacy of biometrics.

The fingerprint image is decomposed by using a visual cryptography scheme to produce two noise-like images (termed as sheets) which are stored in two separate databases. During the authentication, the two sheets are overlaid to create a temporary fingerprint image for matching. The advantage of this system is that the identity of the biometrics is never exposed to the attacker in a single database. However, it requires two separate databases to work together, which is not practical in some applications. The works in [11] combine two different fingerprints into a single new identity either in the feature level [11] or in the image level [12]. In [11], the concept of combining two different fingerprints into a new identity is first proposed, where the new identity is created by combining the minutiae positions extracted from the two fingerprints. The original
minutiae positions of each fingerprint can be protected in the new identity. However, it is easy for the attacker to identify such a new identity because it contains many more minutiae positions than that of an original fingerprint. The experiment shows that the EER of matching the new identities is 2.1% when the original minutiae positions are marked manually from the original fingerprints. A similar scheme is proposed, where the minutiae positions extracted from a fingerprint and the artificial points generated from the voice are combined to produce a new identity. In this work, the EER are shown to be fewer than 2% according to the experimental results.

The Hough transform is a feature extraction technique used in image analysis, computer vision, and digital image processing. The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure. This voting procedure is carried out in a parameter space, from which object candidates are obtained as local maxima in a so-called accumulator space that is explicitly constructed by the algorithm for computing the Hough transform. The classical Hough transform was concerned with the identification of lines in the image, but later the Hough transform has been extended to identifying positions of arbitrary shapes, most commonly circles or ellipses. The Hough transform as it is universally used today was invented by Richard Duda and Peter Hart in 1972, who called it a "generalized Hough transform" after the related 1962 patent of Paul Hough. The transform was popularized in the computer vision community by Dana H. Ballard through a 1981 journal article titled "Generalizing the Hough transform to detect arbitrary shapes". It was initially invented for machine analysis of bubble chamber photographs (Hough, 1959).

The Hough transform was patented as U.S. Patent 3,069,654 in 1962 and assigned to the U.S. Atomic Energy Commission with the name "Method and Means for Recognizing Complex Patterns". This patent uses a slope-intercept parametrization for straight lines, which awkwardly leads to an unbounded transform space since the slope can go to infinity. In [12], the authors first propose to combine two different fingerprints in the image level. First of all, each fingerprint is decomposed into the continuous component and the spiral component based on the fingerprint FM-AM model. After some alignment, the continuous component of one fingerprint is combined with the spiral component of the other fingerprint, so as to create a new virtual identity which is termed as a mixed fingerprint. Compared with the work in [11], such an image level based fingerprint combination technique has two advantages: (i) it is difficult for the attacker to distinguish a mixed fingerprint from the original fingerprints, and (ii) existing fingerprint matching algorithms are applicable for matching two mixed fingerprints. However, this approach produces a visually unrealistic mixed fingerprint due to the variations in the orientation and frequency between the two different fingerprints. Their experimental results show that the EER of matching two mixed fingerprints is about 15% when two different fingerprints are randomly chosen for creating a mixed fingerprint. If the two different fingerprints are carefully chosen according to a compatibility measure, the EER can be reduced to about 4%.

In this paper, we propose a novel system for protecting fingerprint privacy by combining two different fingerprints into a new identity. During the enrollment, the system captures two fingerprints from two different fingers. We propose a combined minutiae template generation algorithm to create a combined minutiae template from the two fingerprints. In such a template, the minutiae positions are extracted from one fingerprint, while the minutiae directions depend on the orientation of the other fingerprint and some coding strategies. The template will be stored in a database for the authentication which requires two query fingerprints. A two-stage fingerprint matching process is further proposed for matching the two query fingerprints against a combined minutiae template. By using the combined minutiae template, the complete minutiae feature of a single fingerprint will not be compromised when the database is stolen. In addition, the combined minutiae template share a similar topology to the original minutiae templates, it can be converted into a real-look like combined fingerprint by using an existing fingerprint reconstruction approach. The combined fingerprint issues a new virtual identity for two different fingerprints, which can be matched using minutiae based fingerprint matching algorithms. The advantages of our technique over the existing fingerprint combination techniques [10] are as follows:

- Our proposed system is able to achieve a very low error rate FRR = 0.4% with FAR = 0.1% when.
- Compared with the feature level based technique [11], we are able to create a new identity (i.e., the combined minutiae template) which is difficult to be distinguished from the original minutiae templates.
- Compared with the image level based technique [12], we are able to create a new virtual identity (i.e., the combined fingerprint) which performs better when the two different fingerprints are randomly chosen.

The organization of the paper is as follows. Section II introduces The Proposed Fingerprint Privacy Protection System. Section III Partial Fingerprint Matching Section IV presents the experimental results, followed by the conclusions in the last section V.

II. THE PROPOSED FINGERPRINT PRIVACY PROTECTION SYSTEM

Figs.1 and 2 shows our proposed fingerprint privacy protection system. In the enrollment phase, the system captures two fingerprints from two different fingers, say fingerprints A and B from fingers A and B, respectively. We extract the minutiae positions from fingerprint A and the orientation from fingerprint B using proposed techniques. Then, by using our proposed coding strategies, a combined minutiae template is generated based on the minutiae positions and the orientation fields are detected from both
A By Level Privacy Protection of Fingerprint Combination Using Hough Transformer

fingerprints. Finally, the combined minutiae template is stored in a database.

**Fig.1. Enrollment Phase.**

In the authentication phase, two query fingerprints are required from the same two fingers; say fingerprints A’ and B’ from fingers A and B. As what we have done in the enrollment, we extract the minutiae positions from fingerprint A’ and the orientation from fingerprint B’. Reference points are detected from both query fingerprints. This extracted information will be matched against the corresponding template stored in the database by using a two stage fingerprint matching. The authentication will be successful if the matching score is over a predefined threshold.

**Fig.2. Authentication Phase.**

### A. Minutiae Point Extraction

A fingerprint is the pattern of ridges and valleys; each individual has unique fingerprints. The uniqueness of a fingerprint is exclusively determined by the local ridge characteristics and their relationships. The two most prominent local ridge characteristics, called minutiae, are the ridge ending and the bifurcation ending and the ridge bifurcation. The first is defined as the point where a ridge forks or diverges into branch ridges. A good quality fingerprint typically contains about 40-100 minutiae points. Fingerprint recognition, is an application in pattern recognition, and is used in security to identity authentication. Fingerprint matching has three different Categories, namely, Correlation Based, Minutiae Based, Ridge feature Based. Minutiae based fingerprint matching is the most widely used fingerprint matching algorithm, and this algorithm too is minutiae based. To implement a minutia extractor, a three-stage approach is widely used by researchers. These stages are preprocessing, minutia extraction and post processing stage. The following Flowchart 1 shows the algorithm for Minutiae point extraction.

![Flowchart-1: Minutiae Point Extraction Algorithm](image)

The pre-processing of FRMSM uses binarization to convert gray scale image into binary image by fixing the threshold value. The binarized image is thinned using Block Filter to reduce the thickness of all ridge lines to a single pixel width to extract minutiae points effectively. Thinning does not change the location and orientation of minutiae points compared to original fingerprint which ensures accurate estimation of minutiae points. To calculate the bifurcation angle, the advantage of the fact that termination and bifurcation are dual in nature is used. The termination in an image corresponds to the bifurcation in its negative image hence by applying the same set of rules to the negative image, the bifurcation angles is obtained. The minutiae location and the minutiae angles are derived after minutiae extraction. The terminations which lie at the outer boundaries are not considered as minutiae points, and Crossing Number is used to locate the minutiae points in fingerprint image. Crossing Number is defined as half of the sum of differences between intensity values of two adjacent pixels. If crossing Number is 1, 2 and 3 or greater than 3 then minutiae points are classified as Termination, Normal ridge and Bifurcation respectively. The cross numbering points are shown in fig.3.

![Fig.3. Cross Numbering Technique](image)

### B. Orientation Estimation

From the second fingerprint the orientation field has to be calculated. Least mean square algorithm is proposed for finding the orientation field. In order to find the orientation
normalization of the fingerprint is required. Normalization can be done either locally or globally. The following Fig.4 shows the processing steps in estimation of fingerprint orientation field. The scheme consists of two steps: local normalization, local orientation estimation, which are summarized as follows.

**Local Normalization:** This step is used to reduce the local variations and standardize the intensity distributions in order to consistently estimate the local orientation. The pixel-wise operation does not change the clarity of the ridge and furrow structures but reduces the variations in gray-level values along ridges and furrows, which facilitates the subsequent processing steps. The global normalization method is also used for the fingerprint enhancement employing a Gabor filter. It can normalize all the values into a defined mean and variance. However, because of the quality of the different parts of the fingerprint image, using the global mean and variance for normalization may not be appropriate. Therefore, we propose using a local normalization to reduce local variations in gray level values.

**Local Orientation:** An orientation image, O(i, j), is defined as an N x N image, where O(i, j) represents the local ridge orientation at pixel (i, j). Local ridge orientation is usually specified for a block rather than at every pixel; an image is divided into a set of w x w non-overlapping blocks and a single local ridge orientation is defined for each block. Note that in a fingerprint image, there is no difference between a local ridge orientation of 90-degree and 270-degree, since the ridges oriented at 90-degree and the ridges oriented at 270-degree in a local neighborhood cannot be differentiated from each other. This step determines the dominant direction of the ridges in different parts of the fingerprint image. This is a critical processing, and errors occurring at this stage are propagated to the frequency filter.

The gradient method for orientation estimation and an orientation smoothing method with a Gaussian window to correct the estimation are used. For a number of non-overlapping blocks with the size of W x W, a single orientation is assigned corresponding to the most probable or dominant orientation of the block. For each pixel in a block, a simple gradient operator, such as the Sobel mask, is applied to obtain the horizontal gradient value Gx(u, v) and vertical gradient value Gy(u, v). The block horizontal and vertical gradients, i.e., Gxx and Gxy, are obtained by adding up all the pixel gradients of the corresponding direction. Then, the block orientation O(x, y) is determined using the block horizontal and vertical gradients. Each block uses a single-orientation value to reduce the computational complexity. This block-wise scheme, however, may be coarse, and it may be difficult to obtain a fine orientation field. In order to estimate orientations more accurately, we use a pixel-wise approach. For each pixel, a block with W x W centered on the pixel is used to compute the average orientation of the pixel. Because of the ambiguity of the orientation values for each position, an orientation smoothing method with a Gaussian window is used to correct the estimation, rather than a simple averaging.

**Fig.4. Orientation Fields Using the Method Proposed.**

**C. Combined Fingerprint Minutiae Template Generation**

The Combined Fingerprint Template is generated by combining the minutiae points extracted from the first fingerprint and the orientation field extracted from the second fingerprint. The combined fingerprint template is generated for various combination of fingerprints. The templates can then be stored in a database which can be used as a reference during the authentication. The following fig.5 shows the basic block diagram of my proposed method.

**Fig.5. Proposed Method for Combined Fingerprint Template Generation.**

### III. PARTIAL FINGERPRINT MATCHING

Our system works on the minutiae-based representation of a fingerprint. Minutiae, in fingerprint context, are the various ridge discontinuities of a fingerprint. More than 100 different types of minutiae have been identified, among which ridge bifurcations and endings (Fig.6) are the most widely used. Minutia-based representation of fingerprints is an ANSI/ISIT standard and contains only local information without relying on global information such as singular points or center of mass of fingerprints. Matching two fingerprints (in minutiae-based representation) is to find the alignment and correspondences between minutiae on both prints. For matching regular sized fingerprint images, a brute-force matching, which examines all the possible solutions, is not feasible since the number of possible solutions increases exponentially with the number of feature points on the prints. In order to increase the efficiency of matching process, other methods instead of brute force matching must be applied. Intuitively, a pre-alignment method may obtain the alignment parameters of two
A By Level Privacy Protection of Fingerprint Combination Using Hough Transformer

fingerprints. Pre-alignment methods that depend on the global singular points are not suitable for partial fingerprint matching. Other pre-alignment techniques need to reprocess all the images thus they cannot be used on already existing databases.

Fig.6. (a) Ridge bifurcations, (b) ridge endings.

There are two major types of features that are used in fingerprint matching: local and global features. Local features, such as the minutiae information and our secondary features, contain the information that is in a local area only and invariant with respect to global transformation. On the other hand, global features, such as number, type, and position of singularities, spatial relationship and geometrical attributes of ridge lines, size and shape of the fingerings, are characterized by the attributes that capture the global spatial relationships of a fingerprint. Because of the nature of partial fingerprints, partial fingerprint matching requires a set of local features that do not depend on global singular structures. Moreover, localized features have the ability to tolerate more distortions has shows that the geometric deformations on local areas can be more easily controlled than global deformations.

A. Secondary Features

The secondary features are derived from minutiae information. We use the minutiae extraction techniques described with some modifications to remove the false minutiae on the edge of the fingerprint foreground to generate the minutiae for our system. The method first gets the image quality maps by checking the low contrast areas, low flow blocks, and high curve regions. And then, a binary representation of the fingerprint is constructed by applying a rotated grid on the ridge flows of the fingerprint. Minutiae are generated by comparing each pixel neighborhood with a family of minutiae templates. Finally, a series of heuristic rules is used to merge and filter out the spurious minutiae (Fig.7). Use relative distance, radial angle, and minutia orientation along with the ridge count and minutia type to generate the features for local matching. The secondary features that we use are similar but the minutiae type and ridge count elements are removed. Minutiae types are difficult to distinguish when impression pressure varies on different applications. Furthermore, ridge count is not universally available and not all minutiae representations in existing databases contain this information.

Fig.7. Minutiae detection process described.

IV. EXPERIMENTAL RESULTS

The experiment is conducted on the first two impressions of the FVC2002 DB2_A database, which contains 200 fingerprints from 100 fingers (with 2 impressions per finger). The Veri Finger 6.3 is used for the minutiae positions extraction and the minutiae matching as shown in Fig.8. The algorithm proposed is used for the orientation extraction.

A. Parameter Settings for Reference Points Detection

The reference point’s detection has a significant impact on the accuracy and efficiency of our proposed system. There are two parameters need to be determined for the reference points detection, i.e., σ for the complex filtering T and which is the threshold for the reference points detection. We set σ = 1.5 as suggested. Next, we explain in detail for the setting of T. A good setting of T should meet the following two requirements for the accuracy and efficiency of our system: (i) the detected reference points should contain the true singular point which is a loop of the fingerprint and (ii) the number of the detected reference points should be small. We manually mark the location of the topmost loop (with the angle pointing upwards) for each of the first two impressions of the FVC2002 DB2_A database (in total 200 fingerprints). Note that if the fingerprint is an arch, the topmost loop is marked at the point with the highest ridge curvature. For each fingerprint, we define the reference point nearest to the marked topmost loop as the nearest reference point for simplicity. The reference points are considered to be truly detected if the Euclidian distance between the marked topmost loop and the nearest reference point is less than 30 pixels as suggested. Otherwise, the reference points are considered to be falsely detected. Table I shows the performance of the reference points detection at different settings of threshold T, where “No.” refers to the total number of reference points.
detected among the 200 fingerprints. It can be seen that setting \( T = 5 \) will achieve a good balance between the accuracy and efficiency. By setting \( \sigma = 1.5 \) and \( T = 5 \), the average Euclidian distance between the marked topmost loop and the nearest reference point is 5.65 pixels. Furthermore, by adopting such settings, we will only detect one reference point for the majority of fingerprints, which is the loop as shown in Figs. 9(a) and 9(b). Figs. 9(c) and 9(d) illustrate an example with two reference points detected.

The reliability of the angle of the nearest reference point (by setting \( \sigma = 1.5 \) and \( T = 5 \)) for each of the 200 fingerprints is measured as follows similar to the work. 

- Rotate the original fingerprint from to \(-30\) degrees with 2 degrees per step based on the topmost loop marked before.
- Perform the reference point’s detection for the original fingerprint and its rotated versions. Let’s denote the angle of the nearest reference point of the original fingerprint as \( \alpha_0 \) and the corresponding angle for each of the rotated versions as \( \alpha_q \), where \( q = -30, -28, ..., 2, 2, 4, ..., 30 \) refers to the degree of rotation.
- Estimate the degree of rotation for each rotated version by
  \[
  \tilde{q} = \alpha_q - \alpha_0
  \]  
- Compute the absolute estimation error for each rotated version, i.e.
  \[
  e = | \tilde{q} - q |
  \]  

Compute the mean and standard deviation of \( e \) for all the rotated versions, which are denoted as \( e_{\text{mean}} \) and \( e_{\text{std}} \), respectively. Among all the 200 fingerprints, the average \( e_{\text{mean}} \) and \( e_{\text{std}} \) are 1.96 degrees and 2.05 degrees, respectively.

Fig. 8. Reconstructing a real-look alike fingerprint image from a set of minutiae points.

**TABLE I: Performance of the Reference Points Detection at Different Settings of Threshold**

<table>
<thead>
<tr>
<th>( T )</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.</td>
<td>1141</td>
<td>650</td>
<td>291</td>
<td>207</td>
</tr>
<tr>
<td>True Detection Rate (%)</td>
<td>99.5</td>
<td>99.5</td>
<td>99.5</td>
<td>98.5</td>
</tr>
<tr>
<td>False Detection Rate (%)</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1.5</td>
</tr>
</tbody>
</table>

B. Evaluating the Performance of the Proposed System

In order to evaluate the performance of our system, we randomly pair the 100 fingers in the FVC2002 DB2_A database to produce a group of 50 non-overlapped finger pairs, where each finger pair contains two different fingers. The random pairing process is repeated 10 times to have 10 groups of 50 non-overlapped finger pairs. For the two fingerprints captured from two different fingers, we can generate two combined minutiae templates in total, where one fingerprint serves as fingerprint A, the other serves as fingerprint B or vice versa. The system designer can choose to enroll one or both of the two templates in the database, which depends on the applications. Thus, we consider the following two cases in building the system database for each group of finger pairs:

- The first impressions of each finger pair are used to produce only one combined minutiae template for enrollment. Therefore, there are 50 templates stored in the database. To compute the False Rejection Rate (FRR), the second impressions of a finger pair are matched against the corresponding enrolled template, producing 50 genuine tests. To compute the False Acceptance Rate (FAR), the first impressions of a finger pair are matched against the other 49 enrolled templates, producing \( 50 \times 49 = 2450 \) imposter tests.

- The first impressions of each finger pair are used to produce two combined minutiae templates for enrollment. Thus, there are 100 templates stored in the database. Similarly, 100 genuine tests are performed to compute FRR and \( 100 \times 99 = 9900 \) imposter tests are performed to compute FAR. In the following discussions, the above two cases are termed as Case I and Case II, respectively.

Fig. 10 plots the average FRR (at different FAR) computed from the 10 groups of finger pairs for the two cases. We can see that our system performs similarly for the two cases. However, the error rates vary among different coding strategies, where the Coding Strategy 1 achieves the lowest error rate with FRR = 0.4% (at FAR = 0.1%) for both cases. While the results of using Coding Strategy 3 are the worst, with over 1% FRR (at FAR = 0.1%) for both cases.
In order to show the effectiveness of the proposed two stage fingerprint matching, we evaluate the performance of our system by using a conventional minutiae matching technique for the fingerprint matching. That is to say, during the authentication, we generate a combined minutiae template from two query fingerprints, which is then matched against the corresponding enrolled template by using a conventional minutiae matching algorithm. Under such an assumption, the performance of our system for Case II is shown in Fig. 11. Note that the combined minutiae templates generated using Coding Strategy 1 cannot be matched directly using conventional minutiae matching algorithm because of the randomness in the minutiae direction. It can be seen that the error rates significantly increase. Compared with the results shown in Fig. 10(b), there are 4.0% and 5.2% increase in FRR at FAR = 0.1% for Coding Strategy 2 and 3, respectively.

Compared with a traditional fingerprint recognition system (hereinafter referred to as a traditional system for simplicity), our proposed system offers more choices for a single user to do the enrollment and authentication. A traditional system can only enroll 10 fingerprint templates for the ten fingers of a user while our system is able to enroll $10 \times 9 = 90$ combined minutiae templates. Among these 90 combined minutiae templates, many share the same minutiae positions or orientations, which could be easily linked. However, they produce the diversity of the choices of the fingerprints (for the user) like passwords. Next, we examine the difference among all the combined minutiae templates that can be created for a set of 10 fingers based on our proposed system. That is to say, we evaluate the performance of our system when the database stores all the combined minutiae templates generated for 10 fingers. We randomly separate the 100 fingers (in FVC2002 DB2_A) into 10 groups with 10 fingers per group. For each group, there are in total $2^{10}$ possible finger pairs.

The first impressions of each finger pair are used to produce two combined minutiae templates for enrollment. The corresponding second impressions serve as the testing fingerprints. As such, 90 combined minutiae templates are generated and stored in the system database. There are 90 genuine tests for computing FRR and $90 \times 89 = 8010$ imposter tests for computing the FAR for each group, where the average FRR for the ten groups (with 10 fingers per group) is shown in Fig. 12. We can see that the error rates of our system increase a lot because some templates either share the same minutiae positions or the same orientation. Among the different coding strategies, the Coding Strategy 3 achieves the lowest error rates with FRR = 6% at FAR = 0.1%. While the corresponding FRR of using Coding Strategy 1 and Coding Strategy 2 is 9.67% and 11.67%, respectively.

C. Evaluating the Performance of the Combined Fingerprints

In this section, we compare the performance of our combined fingerprints with the mixed fingerprints generated by the proposed technique (hereinafter referred to as the
mixed fingerprints for simplicity). The Veri Finger 6.3 is adopted for matching two combined fingerprints or two mixed fingerprints. We use the same 10 groups of 50 non-overlapped finger pairs that are randomly paired at the beginning of Section IV-B. For each group of finger pairs, we consider the same two cases for enrollment as in Section IV-B, i.e.

- The first impressions of each finger pair are used to produce only one combined fingerprint for enrollment. The corresponding second impressions are used to generate a query combined fingerprint. The query combined fingerprint will be matched against its counterpart enrolled in the database to compute the FRR, producing 50 genuine tests. The FAR is computed by matching an enrolled combined fingerprint against other 49 enrolled combined fingerprints, producing 50×49/2=1225 imposter tests, where the symmetric imposter tests are not executed.
- The first impressions of each finger pair are used to produce two combined fingerprints for enrollment. The corresponding second impressions are used to generate two query combined fingerprints. Similarly, we have 100 genuine tests and 100×99/2=4950 imposter tests.

![Fig.13. Performance comparison between the combined fingerprints and the mixed fingerprints for (a) Case I, and (b) Case II.](image)

The above evaluation is also performed by using the mixed fingerprint approach for comparison. Note that the work does not incorporate a noising and rendering step to create the mixed fingerprints. Therefore, in order to do a fair comparison, all our combined fingerprints are created without noising and rendering. Fig.13 shows the performance comparison between the combined fingerprints and the mixed fingerprints. It can be seen that our combined fingerprint achieves a lower error rate than the mix fingerprint. Especially for Case II, our work performs much better when the FAR is less than 1%. The combined fingerprints of Coding Strategy 2 perform the best, with FRR around 15% at FAR = 0.1% for the two cases. A visual comparison among different types of new identities is shown in Fig. 14.

The poor performance of the mixed fingerprints approach for Case II is due to the small overlapping area of some finger pairs. The two mixed fingerprints generated by such a finger pair will be quite similar, which may produce a very high imposter matching score. However, it should be noted that the performance of the mixed fingerprints approach can be much improved by incorporating a compatibility measure, which is used to determine whether two different fingerprints are suitable to generate a visually realistic mixed fingerprint (EER is around 4% according to the results reported). Our combined fingerprint scheme is applicable for minutiae based fingerprint matching algorithms. On a separate note, generating a combined fingerprint would cost more time than creating a mixed fingerprint because of the fingerprint reconstruction.

**D. Evaluating the Probability to Attack Other Systems by Using the Combined Minutiae Templates**

In case the combined minutiae templates are stolen, the attacker can use them to attack other traditional systems which store the original fingerprints. He can reconstruct a fingerprint image from a stolen combined minutia template and make a fake finger based on the reconstructed fingerprint. By scanning the fake finger, the attacker may be able to break into other traditional systems. Similarly, if a combined fingerprint or a mixed fingerprint is stolen, the attacker can directly make a fake fingerprint from the fingerprints and launch the attack. In this section, we evaluate the successful rates to attack other traditional systems by using the combined minutiae templates. Assume that the attacker can do a perfect job to reconstruct a full fingerprint image from combined minutiae template, i.e., the minutiae of the reconstructed fingerprint is exactly the same as the combined minutiae template. Under such an assumption, the successful attack rate by using a combined minutiae template should be higher than using the corresponding combined fingerprint because our current fingerprint reconstruction approach is not perfect. Generally speaking, the attacker can launch the following two types of attacks based on a combined minutiae template:

- The combined minutiae template is used to attack the system which stores the corresponding fingerprint A (mainly provides the minutiae positions).
- The combined minutiae template is used to attack the system which stores the corresponding fingerprint B (mainly provides the minutiae directions).

![Fig.14. Different types of new identities that are generated from two different fingerprints the second row (from left to right): the combined minutiae template, the combined fingerprint from our proposed method (without noising and rendering), and the mixed fingerprint obtained using the approach proposed.](image)
A By Level Privacy Protection of Fingerprint Combination Using Hough Transformer

For simplicity, the above two types of attacks are termed as Attack Type A and Attack Type B, respectively. Suppose the 10 databases built for Case II in Section IV-B are stolen, where each database contains 100 combined minutiae templates. To evaluate the successful rates of the two types of attacks, each stolen template is matched against the corresponding fingerprint and fingerprint using the Veri Finget 6.3, respectively. Thus, we have 1000 matches for Attack Type A and 1000 matches for Attack Type B. Again, in order to compare with the work, the same evaluation is performed by using the mixed fingerprint approach.

Fig.15 shows the successful rates of the two types of attacks by using the combined minutiae templates and the mixed fingerprints. Note that the security thresholds (FAR) of the tradition system are computed over FVC2002 DB2_A based on the FVC2002 protocol. We can see that the coding strategies do not have a significant impact on the successful rates of the two types of attacks. Compared with using the mixed fingerprints, it is more difficult to launch the attacks by using the combined minutiae templates. For Attack Type A, the successful rate is around 25% at using the combined minutiae templates while the corresponding successful rate is 57.5% for the mixed fingerprints. For Attack Type B, the successful rates significantly reduce for both the combined minutiae templates and the mixed fingerprints. At FAR = 0.1%, the successful rate is almost 0% by using the combined minutiae templates. While the corresponding successful rate is 36.7% for the mixed fingerprints, such a reduction shows that it is more dangerous to lose the minutiae positions than losing the orientation of the fingerprint. If the attacker knows that a stolen template has been protected by using our technique, he would try to launch the aforementioned attacks based on the minutiae positions only, i.e., he would try to modify the minutiae matcher such that the minutiae directions are ignored during the matching. We implement a minutiae matcher based on the work proposed, where we only use the minutiae positions for the matching. By using this matcher, the successful rates of Attack Type A and Attack Type B are 86.0% and 0.3% at FAR = 0.1%, respectively.

---

Fig.15. Attack a traditional system using a combined minutiae template or a mixed fingerprint. (a) Attack Type A; (b) Attack Type B.

V. CONCLUSION

In this paper, we introduce a novel system for fingerprint privacy protection by combining two fingerprints into a new identity. In the enrollment, the system captures two fingerprints from two different fingers. A combined minutiae template containing only a partial minutiae feature of each of the two fingerprints will be generated and stored in a database. To make the combined minutiae template look real as an original minutiae template, three different coding strategies are introduced during the combined minutiae template generation process. In the authentication process, two query fingerprints from the same two fingers are required. A two-stage fingerprint matching process is proposed for matching the two query fingerprints against the enrolled template. Our combined minutiae template has a similar topology to an original minutiae template. Therefore, we are able to combine two different fingerprints into a new virtual identity by reconstructing a real-look alike combined fingerprint from the combined minutiae template. The experimental results show that our system achieves a very low error rate with FRR = 0.4% at FAR = 0.1%. It is also difficult for an attacker to break other traditional systems by using the combined minutiae templates. Compared with the state-of-the-art technique, our technique can generate a better new virtual identity (i.e., the combined fingerprint) when the two different fingerprints are randomly chosen. The analysis shows that it is not easy for the attacker to recover the original minutiae templates from a combined minutiae template or a combined fingerprint.

VI. REFERENCES


Author’s Profile:
S. Neelavathi, Completed BTech in Rajoli Veera Reddy Padmaja Engineering college, Kadapa and Studying M.Tech in Bharat College of Engineering & Technology for Women, Kadapa, AP, India.

A. Rajendra Babu, working as an Associate Professor in Bharat College of Engineering & Technology for Women, Kadapa, AP, India.