Privacy and Security of Features Extracted from Minutiae Aggregates

Abhishek Nagar, Shantanu Rane, Anthony Vetro

TR2010-004 March 23, 2010

Abstract

This paper describes our recent analysis on the security and privacy of biometric feature vectors obtained from fingerprint minutiae. A large number of contiguous regions (cuboids) are selected at random in the minutiae space, and several new features are extracted from the minutiae inside each such cuboid. Specifically, the features are extracted from the average minutia coordinate within a cuboid, the standard deviation of the minutiae coordinates, and the aggregate wall distance, i.e., the sum of distance of each minutia from the boundary of the cuboids. In terms of matching performance on a public database, the feature vectors provide an equal error rate of 3% even if the imposter is allowed to use the same local patches as the genuine user. Performance within a secure biometrics framework is evaluated by applying an LDPC code to the feature vectors and storing only the syndrome at the access control device, for use in authentication. The paper concludes with a discussion on methods to analyze security and privacy of biometric systems that use such local-aggregate-based feature vectors in a secure biometric recognition framework. This discussion highlights security attacks via template injection, spoofing, and cancelability compromises and also considers the difficulty of privacy attacks via template inversion.

IEEE International Conference on Acoustics, Speech and Signal Processing

PRIVACY AND SECURITY OF FEATURES EXTRACTED FROM MINUTIAE AGGREGATES

Abhishek Nagar[*]

Michigan State University
East Lansing, MI.

Shantanu Rane, Anthony Vetro

Mitsubishi Electric Research Laboratories
Cambridge, MA.

ABSTRACT

This paper describes our recent analysis on the security and privacy of biometric feature vectors obtained from fingerprint minutiae. A large number of contiguous regions (cuboids) are selected at random in the minutiae space, and several new features are extracted from the minutiae inside each such cuboid. Specifically, the features are extracted from the average minutia coordinate within a cuboid, the standard deviation of the minutiae coordinates, and the aggregate wall distance, i.e., the sum of distance of each minutia from the boundary of the cuboids. In terms of matching performance on a public database, the feature vectors provide an equal error rate of 3 % even if the imposter is allowed to use the same local patches as the genuine user. Performance within a secure biometrics framework is evaluated by applying an LDPC code to the feature vectors and storing only the syndrome at the access control device, for use in authentication. The paper concludes with a discussion on methods to analyze security and privacy of biometric systems that use such local-aggregate-based feature vectors in a secure biometric recognition framework. This discussion highlights security attacks via template injection, spoofing, and cancelability compromises and also considers the difficulty of privacy attacks via template inversion.

Index Terms— secure biometrics, fingerprints, binary features, template inversion

1. INTRODUCTION

With the increasing use of biometric recognition systems in our daily activities, the privacy and security of biometric templates stored in these systems is gaining importance. There are many fears about the possible misuse of stolen templates, e.g., they can be used to create spoof biometrics which compromise the system being protected and may even compromise the privacy of a legitimate user. In order to eliminate such apprehensions, a number of template protection techniques have been proposed. These techniques usually transform the biometric template before storage in such a way that the original biometric cannot be recovered from the stored information. In some systems, the stored template can be revoked if it is known to have been compromised.

In this work, we study the privacy and security of the features extracted from fingerprint minutiae aggregates. Our overall strategy is based on extracting a binary vector from a minutiae map, applying an error correcting code to this vector and storing its syndrome as a secure biometric [1]. During authentication, the system accepts a probe biometric and attempts to recover the original biometric with the help of the syndrome. A cryptographic hash of the recovered original biometric is compared with a hash of the enrollment biometric to confirm the success or failure of authentication. The analysis presented here also applies to previously proposed secure biometrics schemes [2, 3, 4] which are functionally similar to the syndrome-based scheme used herein.

At this point, it is essential to clarify the meaning of security and privacy in the context of this work. Security is measured by the number of attempts needed by an imposter to successfully authenticate as a genuine user. Privacy is measured by the number of attempts needed by an attacker to recover the original biometric which, in our case, is the fingerprint minutia map. Naturally, security and privacy depends on the way in which the features are extracted and secured and the type of side information available to the attacker. In, [1], binary features were extracted from the number of minutiae present in randomly chosen cuboidal patches in the (X, Y, Θ) space occupied by the minutia. The present work differs from [1] in three fundamental ways which will be detailed in the sequel: First, each cuboid now generates a richer feature set from which a larger number of bits can be extracted and those with the highest discriminability are used for matching. Second, the fingerprints belong to a public database and therefore are no longer pre-aligned. We adopt a method first proposed by Nandkumar et al. [5] to align the fingerprints during enrollment, training and testing. Third, the difficulty of template inversion, i.e., recovering the minutiae map given the binary feature vector and all the random cuboids, is considered for the first time. We also discuss the security with respect to different attacks that can be staged on the proposed system, viz., template injection, spoof creation, and cancelability attacks.

The remaining paper is organized as follows: section 2 describes the features extracted, section 3 explains the procedure for selecting specific features. The experimental results are provided in section 4. Section 5 discusses the various security and privacy aspects of the proposed scheme and section 6 provides a discussion on non-invertibility of the template.

2. FEATURE EXTRACTION

The motivation for extracting features from aggregate measures calculated over a contiguous region, such as a cuboid or a sphere, comes from inherent properties of fingerprint minutia. While fingerprint minutia remain stable over many years, their locations on a minutia map vary slightly at every measurement: They may move slightly or even disappear owing to differences in pressure applied to the sensor or due to misalignment. Moreover, new minutia points may be inserted because of dust or cuts on the finger. Therefore, feature vector bits based on individual minutia points is unreliable. However, when features are based on aggregate measures calculated over a region, it is possible to account for, and mitigate, the effects of minutiae movement, insertion and deletion. In this work, the region is a cuboid with randomly chosen dimensions along the spatial (X and Y) axes and along the orientation (Θ) axis. Corresponding to each randomly chosen cuboid, we introduce three minutiae-based

[*] This work was performed while A. Nagar was an intern at MERL.

features, viz. (1) *Aggregate wall distance:* Summation of the closest distance of each minutia from the cuboid boundary, (2) *Minutiae Average:* Average coordinate of all the minutiae present in each cuboid in a given measurement, and (3) *Minutiae Deviation:* Standard deviation of minutiae coordinates present in each cuboid in a given measurement. These are elaborated further below.

The aggregate wall distance (δ) for a cuboid bounded by $(x_{min}, x_{max}, y_{min}, y_{max}, \theta_{min}, \theta_{max})$ is computed as:

$$\delta = \sum_{i=1}^{t} \min(\delta_x, \delta_y, \delta_\theta, \tau_\delta) \quad (1)$$

where t is the number of minutiae in the given cuboid, τ_δ is a threshold used for wall distance, and δ_x, δ_y, and δ_θ are given by $\min(|x_i - x_{min}|, |x_i - x_{max}|)$, $\min(|y_i - y_{min}|, |y_i - y_{max}|)$, and $\min(|\theta_i - \theta_{min}|, |\theta_i - \theta_{max}|)$, respectively. If all the minutiae are at distance atleast τ_δ from the cuboid boundary, the aggregate wall distance is τ_δ times the number of minutiae in the cuboid. The threshold τ_δ deemphasizes the contribution of the minutiae close to boundary that are likely to shift out of the cuboid in the subsequent impressions due to imperfect alignment.

Both minutiae average and minutia deviation features consist of three components each corresponding to the X, Y, and Θ axes coordinates. Standard formulas are used for computing the average and the standard deviation for the X and Y coordinates whereas for Θ coordinate, the mean μ_θ and the standard deviation σ_θ are computed as follows:

$$\mu_s = \frac{1}{t}\sum_{i=1}^{t} \sin\theta_i,\ \mu_c = \frac{1}{t}\sum_{i=1}^{t} \cos\theta_i,\ \mu_\theta = \arctan\left(\frac{\mu_s}{\mu_c}\right)$$

$$\sigma_\theta = \sqrt{\frac{1}{t-1}\sum_{i=1}^{t}\left[\min(|\theta_i - \mu_\theta|, 360 - |\theta_i - \mu_\theta|)\right]^2}$$

where θ_i is the angle corresponding to the i^{th} minutia. If there is no minutia in a particular cuboid, the average features assume the value corresponding to the center of the cuboid whereas the deviation features are set to zero. The deviation features are also set to zero if there is only a single minutia in the cuboid. The extracted features are binarized using the median value of a given feature calculated over all enrolled fingerprints. Using the median value as the threshold ensures that each bit has equal probability of being 1 or 0.

3. SELECTION OF DISCRIMINABLE FEATURES

There are two advantages of randomly generating the cuboids. The first advantage is cancelability: If the template is compromised, a new set of randomly generated features can be used to create a completely new binary template. The second advantage is a large choice of features from which to perform feature selection. Appropriate selection of features is required in order to eliminate features that may be too correlated or too noisy. In order to ensure that most of the cuboids occupy the printed region in a fingerprint, each cuboid is centered at a randomly selected minutiae from the database. The remaining three parameters of cuboids i.e., the dimensions along the X, Y, and Θ directions were randomly generated. Prior to this, all the fingerprints in the database are shifted such that center of the bounding rectangle coincides with the center of the fingerprint image.

As shown in [6] the two main criteria of feature selection are dependence among the features (or redundancy) and discriminability (or relevance) of each feature. If the features have high correlation then some of the features can be discarded in order to accommodate additional unrelated features, thereby improving the matching performance. As a first step in eliminating inferior features, those cuboids that have large overlap with other cuboids are discarded, because these would generate highly correlated features. The following procedure is used for this as in [1]: (1) Compute the relative overlap, i.e., the ratio of volume of intersection to the volume of union, for all pairs of cuboids. (2) Select the pair having highest relative overlap. (3) Discard the cuboid whose maximum relative overlap with the other cuboids is greater. (4) Repeat steps 2 and 3 until desired number of nearly non-overlapping cuboids or the desired reduction in overlap is obtained.

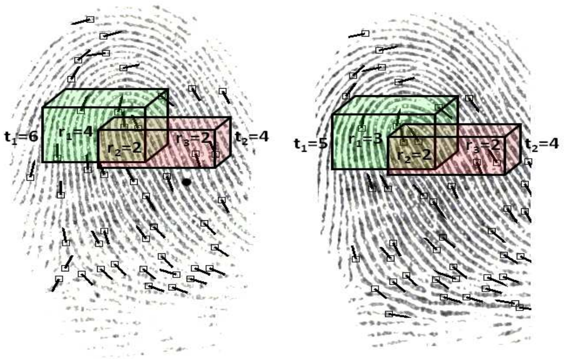
Next, 7 bits are extracted from each of the remaining cuboids as explained in Section 2. Of these, 3 bits are obtained from the minutia average, 3 bits fom the minutia deviation and 1 bit from the aggregate wall distance. Then, replacing the "overlap" criterion in the above 4 steps by "bit correlation," the number of usable features is reduced. After this step, different cuboids contribute a different number of nearly uncorrelated bits to the feature vector.

Having obtained uncorrelated feature bits, it is necessary to consider the user-specific discriminability of each extracted bit. For this purpose, we compute the discriminability of each feature for each of the enrolled fingerprints. The discriminability of each bit (d_i) is computed as $d_i = I_i - G_i$ where G_i is the fraction of times when the i^{th} bit disagrees for the genuine matches and I_i is the fraction of times when the i^{th} bit disagrees for the impostor matches. The bits retained after this discriminability-based rejection are finally used in the LDPC based secure recognition system.

4. EXPERIMENTS

The FVC2002 Database-2 [7] was used in our experiments. This publicly available database contains 100 different users with 8 finger impressions per user. Each fingerprint is captured using an optical fingerprint scanner and digitized at 569 dpi. Fig 1 shows two impressions from a fingerprint. In our experiments, the first impression of each user is enrolled, the next 6 impressions are used for training and the last impression is used for testing. Prior to feature extraction, the fingerprints are aligned using high curvature points as described in [5]. Points along fingerprint ridges, that have high ridge curvature are extracted and stored along with the template. During authentication, similar points are extracted from the query fingerprint and matched with the stored set to align the fingerprints. The high curvature points do not reveal any significant information about the minutiae. Initially, 750 random cuboids are generated, of which 250 are eliminated based on high overlap as explained above. Then, with seven binary features per remaining cuboid, a long feature vector of length $500 \times 7 = 3,500$ bits is generated. Out of these $3,500$ bits, 2000 are eliminated based on high correlation as described above.

Now, the set of six training impressions per finger is matched with the corresponding enrolled impression to evaluate the discriminability of each binary feature bit. In order to compute the impostor scores for training, second impressions of 20 different users are used. The discriminability is computed separately for each user in order to take into account the difference in the fingerprint area printed and the distribution and quality of minutiae. Finally, a set of 300 bits are selected based on high discriminability and are used to evaluate the security-robustness tradeoff of the binary features. Normalized Hamming Distance (NHD) is used as a distortion measure between



(a) (b)

Fig. 1. Example fingerprints from FVC2002 DB-2 corresponding to the same finger . Two cuboids containing t_1 and t_2 minutiae intersect to create three fragments each having r_1, r_2 and r_3 minutiae.

Word Length	Synd. Length	Rate	FAR (%)	GAR (%)
300	240	0.2	0.01	69
300	255	0.15	0.13	85
300	285	0.05	1.05	95

Table 1. Performance of a secure biometric system using a LDPC (syndrome) code operating on the 300 most discriminable bits.

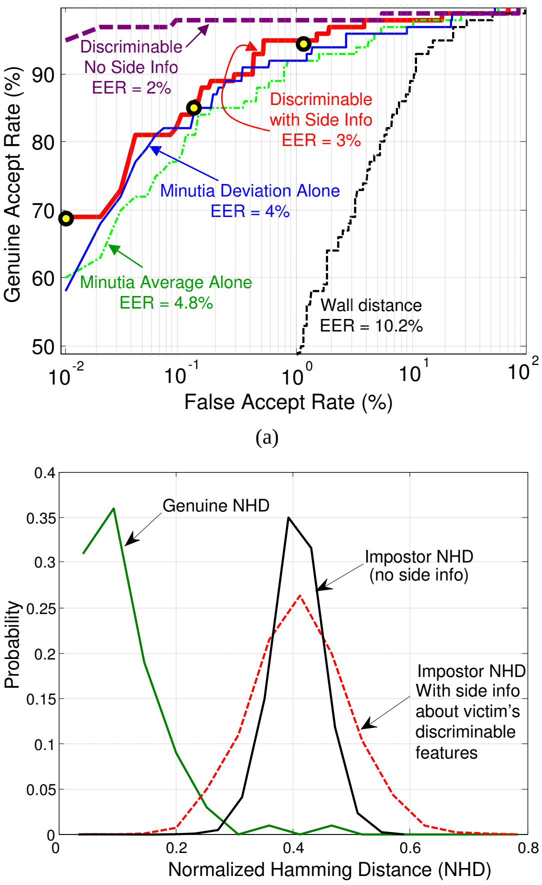
two feature vectors. The Receiver Operating Characteristic (ROC) curves corresponding to the various individual features and the final set of features selected using the discriminability are shown in figure 2. As clear from the results, the matching performance is significantly improved by incorporating the average and variance based minutiae features. When an imposter knows the discriminable feature bits, the equal error rate (EER) was found to be 3 %. Under normal operational circumstances, when an imposter does not know the discriminable feature bits for the victim, the EER was found to be 2 % as shown in the plots.

Lastly, three different irregular LDPC codes with rate 0.2, 0.15, and 0.05 were used to evaluate the security-robustness tradeoff of a secure biometric system built around the feature extraction described earlier. In this case, given the syndrome of the LDPC code, and the probe biometric, the access control device attempts to recover the enrollment biometric using Belief Propagation (BP). Table 1 shows the corresponding results for this system, in which only the syndrome is stored on the access control device, not the enrollment feature vector. The three FAR-FRR data points thus obtained are superimposed on Fig. 2(a). Note that, given only the syndrome as a secure biometric, an attacker can conceivably generate a feature vector which is not identical to the enrolment feature vector, but produces the same syndrome. To prevent access control in attacks of this kind, a cryptographic hash of the enrolment feature vector is stored on the device. The feature vector obtained after BP decoding is hashed and compared with the hash of the enrolment feature vector. Access is granted only if the hashes match.

5. SECURITY ANALYSIS

We now discuss the security of the proposed system in terms of the following attacks:

1. Template Injection: Given the stored syndrome how difficult is it to guess a binary template which can be used for successful authentication? One strategy is to obtain all the binary vectors that produce the syndrome stored in the database. There are 2^k such vectors where k is the dimension of the ECC employed. Using Gaussian Elimination to convert the parity check matrix of the LDPC code to its systematic form, it is straightforward to obtain a vector that would produce the given syndrome. The remaining $2^k - 1$ vectors can also be obtained using the corresponding generator matrix. The attacker can then compute the cryptographic hash values of these vectors and compare it with the hash of the enrollment feature vector, which is available on the device. Thus the complexity of this attack is at most k-bits. For an LDPC code k is given by the difference between the length of the template and that of syndrome.



(a)

(b)

Fig. 2. (a) ROC curves for 300-bit features extracted from individual properties (average, deviation, wall distance), and 300-bit features extracted from the most discriminable combination of features. Imposter side information refers to knowledge of the cuboids used to generate the victim's feature set (b) Normalized hamming distance (NHD) for genuine and impostor matches for the case when the impostor does or does not know the side information.

2. Spoof Attack: How difficult is it to obtain a fake fingerprint or minutia map that authenticates successfully? To create a spoof biometric, it is necessary to know a fingerprint close enough to the original fingerprint. One way to do this is to try different fingerprints from a database until there is a false accept. If the system has a false accept rate f then, on average, it would require $\approx 1/f$ different fingerprints to obtain a false accept. If $C(f)$ is the number of binary computations required to test $1/f$ different fingerprints then, in terms of guessing a binary vector, this amounts to approximately $\approx (\log_2 C(f)$ bits of security. Since a spoof biometric can be used to construct a binary feature template, the true complexity of a template injection attack is approximately $\min(k, \log_2 C(f))$. Here, it is assumed that the feature extraction procedure, i.e., configuration of the cuboids and the various thresholds are known to the attacker.

3. Cancelability Attack: How difficult is it to guess a stored biometric template given that the attacker already has access to another template created from the same biometric? If compromised, the stored syndrome is replaced with another syndrome of the template generated using a different set of cuboids. Cancelability pertains to the difficulty in guessing the new template given information about another template generated from the same biometric. As the above attacks related to template injection and spoof allow reconstruction of any biometric template given a syndrome, cancelability attack is at most as hard as template injection, spoof and also template invertibility. Thus, the complexity of a cancelability attack is at most $\min(k, \log_2 C(f))$. Note that, the attacker can take advantage of the correlation among the new set of features and the old set in order to reduce this complexity. E.g., if the attacker has a spoof biometric for an earlier template, the same spoof might compromise the new template with minor modifications. Therefore, the features used for the new template must be sufficiently uncorrelated with the earlier template.

4. Template Inversion: How difficult is it to guess the enrolled fingerprint or the original minutia map? This attack will be elaborated upon in the next section. Note that, template inversion can lead to all the aforementioned attacks and even *other* systems using the same biometric and possibly different feature set can also be compromised.

6. MEASURING TEMPLATE NON-INVERTIBILITY

Non-invertibility refers to the difficulty in guessing the minutiae configuration of the original fingerprint given the binary template. Let $t_i, i = 1, ..., n$ be the number of minutiae in each of the n cuboids for a given fingerprint impression. Further, consider the fragments of each cuboid obtained due to intersection with other cuboids and let $r_i, i = 1, ..., m, m \gg n$ be the number of minutiae in each of these m non-intersecting fragments belonging to all the n cuboids. Figure 1 shows the cuboids and fragments in two fingerprints from same finger. The association between t_i's and r_i's is captured by the coefficients I_{ij} (3). Since the feature bits in the proposed algorithm are extracted by thresholding the t_i's, access to the final template and the configuration of the cuboids allows the attacker to set up the following set of inequalities:

$$\begin{aligned} w_1(I_{11}r_1 + I_{12}r_2 + ... + I_{1m}r_m) &< \tau_1' \\ w_2(I_{21}r_1 + I_{22}r_2 + ... + I_{2m}r_m) &< \tau_2' \\ &\vdots \\ w_n(I_{n1}r_1 + I_{n2}r_2 + ... + I_{nm}r_m) &< \tau_n' \\ r_i \geq 0, i = 1, ..., m & \end{aligned} \tag{2}$$

where w_i is the average wall distance for the minutiae in the i^{th} cuboid and

$$I_{ij} = \begin{cases} 1 & \text{if fragment} j \text{ belongs to cuboid } i \text{ and } b_i = 0 \\ -1 & \text{if fragment} j \text{ belongs to cuboid } i \text{ and } b_i = 1 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where b_i is the bit corresponding to the aggregate wall distance in the i^{th} cuboid and $\tau_i' = \tau_i$

$$\tau_i' = \begin{cases} -\tau_i & \text{if } b_i = 0 \\ \tau_i & \text{if } b_i = 1 \end{cases} \tag{4}$$

where τ_i is the corresponding median wall distance threshold. Note that, since $m \gg n$, this system of equations is heavily underdetermined. The above problem can be transformed to a linear system of inequalities by replacing all the w_i's by τ_δ (see (1)) and then solving 2 for a integer-valued solution for r_i. If there are unsatisfied constraints, some of those can be satisfied by appropriately selecting the corresponding w_i.

For every possible solution of the r_i, the feature bits corresponding to the minutiae average and minutiae deviation can be used to obtain estimates for the mean and variance of each of the X, Y and Θ coordinates of minutiae by solving another similar set of inequalities. Note that, solving for mean and variance of minutiae would have additional constraints due to the position and size of each fragment. Finally, given the mean and variance in each fragment, minutiae can be appropriately placed to obtain an estimate of the original template. This attack is extremely difficult and moreover, it will not reproduce the original biometric. However, it is likely that the obtained minutiae distribution is closer to the original as compared to the one obtained via a spoof attack. A less complex but less precise way to obtain the original biometric is to tessellate the feature space into equal sized fragments which can be associated with each cuboid instead of explicitly computing the intersection regions of a large number of cuboids. Then, slack variables can be introduced in the set of equations 2 to take into account the difference in the space covered by cuboid and the associated fragments.

7. REFERENCES

[1] Y. Sutcu, S. Rane, J.S. Yedidia, S.C. Draper, and A. Vetro, "Feature Extraction for a Slepian-Wolf Biometric System Using LDPC Codes," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, July 2008.

[2] G. I. Davida, Y. Frankel, and B. J. Matt, "On Enabling Secure Applications Through Off-Line Biometric Identification," in *Proc. IEEE Symposium on Security and Privacy*, Oakland, USA, May 1998, pp. 148–157.

[3] A. Juels and M. Wattenberg, "A Fuzzy Commitment Scheme," in *Proc. Sixth ACM Conf. on Computer and Communications Security*, Singapore, November 1999, pp. 28–36.

[4] P. Tuyls, A. H. M. Akkermans, T. A. M. Kevenaar, G.-J. Schrijen, A. M. Bazen, and R. N. J. Veldhuis, "Practical Biometric Authentication with Template Protection," in *Proc. Fifth Intl. Conf. on Audio- and Video-based Biometric Person Authentication*, Rye Town, USA, July 2005, pp. 436–446.

[5] K. Nandakumar, A. K. Jain, and S. Pankanti, "Fingerprint-based Fuzzy Vault: Implementation and Performance," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 4, pp. 744–757, December 2007.

[6] H. Peng, F. Long, and C. H. Q. Ding, "Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.

[7] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, and A. K. Jain, "Fvc2002: Second fingerprint verification competition," 2002, vol. 3, pp. 811–814.