Combination of Symmetric Hash Functions for Secure Fingerprint Matching

Gaurav Kumar
University at Buffalo
gauravku@buffalo.edu

Sergey Tulyakov
University at Buffalo
tulyakov@buffalo.edu

Venu Govindaraju
University at Buffalo
govind@buffalo.edu

Abstract

Fingerprint based secure biometric authentication systems have received considerable research attention lately, where the major goal is to provide an anonymous, multipliable and easily revocable methodology for fingerprint verification. In our previous work, we have shown that symmetric hash functions are very effective in providing such secure fingerprint representation and matching since they are independent of order of minutiae triplets as well as location of singular points (e.g. core and delta). In this paper, we extend our prior work by generating a combination of symmetric hash functions, which increases the security of fingerprint matching by an exponential factor. Firstly, we extract k-plets from each fingerprint image and generate a unique key for combining multiple hash functions up to an order of (k-1). Each of these keys is generated using the features extracted from minutiae k-plets such as bin index of smallest angles in each k-plet. This combination provides us an extra security in the face of brute force attacks, where the compromise of few hash functions as well do not compromise the overall matching. Our experimental results suggest that the EER obtained using the combination of hash functions (5.4%) is comparable with the baseline system (3.0%), with the added advantage of being more secure.

1. Introduction

Due to increased use of biometrics in civilian applications and the concerns about the privacy of biometric data, real world deployment of biometric systems puts additional requirements on them. Three major characteristics of such secured systems can be identified: privacy of biometric templates (the inability of the intruder to obtain original biometric measurements from the stored biometric templates), multiplicity (where same biometric modality can be used for different applications) and easy revocability of stored templates if compromised. The individuality and ease of acquiring fingerprints make it one of the most widely used modality for biometric authentication. Thus, achieving three above characteristics for fingerprint matching systems has become a priority in the fingerprint biometric research.

We have demonstrated earlier the usage and performance of *symmetric hash functions* achieving an EER of 3.0% on triplets of neighboring minutia [12]. We also proposed the ways to enhance the security of the system using a combination of such functions, which would reduce the chances of breaking the system even more. In this paper we propose the approach of combining symmetric hash functions extracted from minutia k-plets and analyse the performance of using multiple hash functions. Specifically, we demonstrate the usage of 4-plets and 5-plets for matching and security and compare the accuracy of the system with our baseline results on triplets. The evaluation and comparison of the performance of the system on partial fingerprints is also covered based on challenges discussed by Jea [6] in matching partial fingerprints such as: (i) few number of minutia points available, thus reducing its discriminating power; (ii) likely absence of singular points (core and delta) and (iii) uncontrolled impression environments resulting in unspecified orientations of partial fingerprints.

The outline of the paper would be as follows. We give a brief overview of our system and related previous research in section 2. The proposed approach of combination of symmetric hash functions is presentedin section 3 and, finally, the experimental results are given in section 4.

2. Related Work

Few techniques have been proposed for generating cancelable biometric templates, which mostly fall in two directions. The first direction is based on the use of the error correcting codes. The techniques in this category assume that biometric templates from the same

user have little variation, so that this variation can be overcome by the use of error correcting codes. In this case it is sufficient to only keep the error correcting data and the cryptographic hash of the original template; the deformed test template will be corrected using error correcting data, and the hash of the corrected template can be directly corrected to the stored hash of the original template [4]. Fuzzy vault schemes [7, 8], biotokens [3], fuzzy extractors [1] and secure sketches [11] follow the this idea. The major drawback of the techniques in this category is being able to extract biometric templates having little variation for the same user; this task can be difficult for fingerprints due to the lack of their natural alignment.

The techniques of the second category try to construct non-invertible transformations of the original biometric templates and perform matching in the transformed domain [9]. Minutia triplet binning [5], cancelable biometric filters [10] and random multispace projections [2] belong to this category. Whereas some methods require here require template prealignment [9, 2], other methods do not require it [5] or find the alignment information from the transformed templates [10].

We proposed in our earlier work [12], the properties of *symmetric hash functions*, they are invariant to ordering of the minutiae. Hence independent of the prealignment or location of singular points. Also, they can be easily represented by lower order functions. Some of the key techniques used by our system in [12] are listed below.

We represented minutia points in the complex plane and assumed that two fingerprints of the same finger can have different position, rotation and scale, coming from (possibly) different scanners and different positioning of the finger on the scanner and considered symmetric complex functions for matching [12]. Given n minutia points c1, c2, . . . , cn we constructed the m symmetric hash functions as shown in Equation 1. If the number of hash functions (m) is less than the number of minutia points(n) participating in the construction of the hash function, then it is not possible to restore the original minutia positions given the hash values. Earlier, the experiments were carried out on minutia triplets, hence we were restricted to use the hash functions of maximum order 2 to ensure the cancellability. One might argue that a brute force attack would still be able to determine the actual minutia locations if we have a single hash function. As a remedy to that we [12] proposed various ways of using multiple hash functions together and design a mapping strategy between the hash function used and minutia triplet. The representation of minutia triplets in triangular parametric space was one such technique. If we represent a minutia triplet in a parametric space and apply a particular hash function to it a similar triplet having similar triangular resemblance would fall close to the enrolled template and same hashing could be done on both triplets [12].

3. Proposed Approach

We propose a mapping strategy for combining multiple hash functions based on certain triplets or k-plets features. In the case of triplets, such features could be angles formed by the triplets, their sides or a combination of altitude and base. We bin the triplets on basis of one of these features and map specific hash function to each bin. Hence each bin acts as the key for the choice of the hash function. A random seed could be generated for the mapping of a bin to specific hash function. Our system first extracts the secondary features that include the angle between the minutiae triplets, (The minutia points are represented as complex number) and bins the triplets on the basis of one of the features mentioned above. The system is illustrated in Figure 1.

The exact location of the minutia triplet should be intractable. We know so far that the higher order symmetric hash functions can be represented in terms of lower order hash function as shown in Equation 2 . We cannot perform matching using a higher order hash function without storing the lower order functions. Hence, for a triplet and with $h > 3$ order hash function we would be having $h > 3$ equations and 3 unknowns. In case the database is compromised the exact location of the minutiae could be retrieved. The solution we propose here is the usage of k-plets instead of triplets such that the maximum order j of the hash functions used is less than k .In which case there would be k unknowns and $j < k$ equations, hence infinite number of solutions. We tested the performance of our system using 4-plets and 5-plets and hash functions combinations of maximum order three and four respectively. In case of a 5-plet ,five nearest neighbors for each minutia are obtained, the features (e.g angles) are obtained as shown in Figure 2. The results are shown in Section 4.

$$\begin{aligned} h'_1 &= rh_1 + nt \\ h'_2 &= r^2h_2 + 2rth_1 + nt^2 \\ h'_3 &= r^3h_3 + 3r^2th_2 + 3rt^2h_1 + nt^3 \\ h'_4 &= r^4h_4 + 6r^2t^2h_2 + 4r^3th_3 + 4rt^3h_1 \end{aligned} \quad (1)$$

4. Experiments & Results

We conducted a sequence of experiments on the minutia 4-plets and 5-plets and compared the perfor-

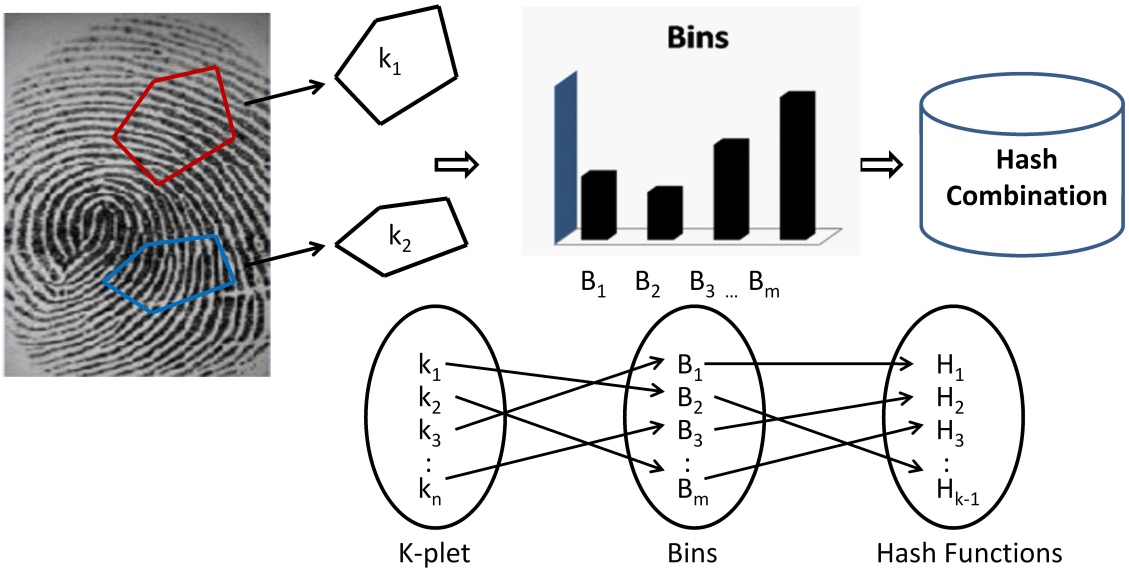

Figure 1. Overall Schematic Diagram of Hashing using K-plets

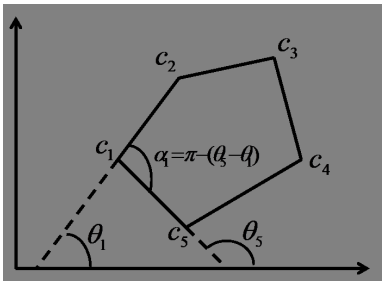

Figure 2. Retrieving feature from a K-plet

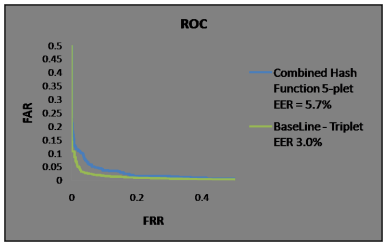

Figure 3. Performance Comparison with baseline

mance with baseline results on triplets. The experiments were conducted on individual hash functions of order 3 and on the system with a combination of hash functions.The equations for the hybrid system for 5-plet are shown in Equation 2. The dataset used in all cases was FVC2002 DB2. The performance of the system with the combination of Hash functions , on 4-plet and 5-plet is shown in Fig 4. The EER dropped down to 5.7% in case of 5-plet. However, we can prove theoretically the increase in the security. Assuming a brute force attack, for a 'm' hash function combination, and assuming average 'c' k-plets a total of m^c possible combination need to be tried to actually break the system. Also, consider a scenario where one or more hash functions are broken, even then the possibility of a match is very less because of different hash functions being applied on different k-plets of same fingerprint template.in case of a compromise , a new key could be generated with different bin to hash function mapping.

Partial Fingerprint test was performed on images extracted from FVC 2002 DB2 dataset using 5-plets. As in [6] five partial fingerprints were generated for target sizes 70%, 75%, 80%, 85%, 90% and the results are shown in Figure 4.

$$
\begin{aligned}
h_1(c_1,c_2,c_3,c_4,c_5) &= (c_1+c_2+c_3+c_4+c_5)\\
h_2(c_1,c_2,c_3,c_4,c_5) &= (c_1-c_2)^2+(c_2-c_3)^2\\
&\quad +(c_3-c_4)^2+(c_4-c_5)^2\\
&\quad +(c_5-c_1)^2 \qquad (2)\\
h_3(c_1,c_2,c_3,c_4,c_5) &= c_1^3+c_2^3+c_3^3+c_4^3+c_5^3\\
h_4(c_1,c_2,c_3,c_4,c_5) &= c_1^4+c_2^4+c_3^4+c_4^4+c_5^4
\end{aligned}
$$

Percentage	FAR %	FRR %	TER %	EER %
90	5.3%	13.1%	18.4%	10.4%
85	5.1%	13.3%	18.4%	10.2%
80	8.8%	16%	24.8%	12.5%
75	7.0%	21.6%	28.6%	16.46%
70	12%	20.05%	32.05%	18.13%

Figure 4. Performance on Partial finerprint

References

[1] A. Arakala, J. Jeffers, and K. Horadam. Fuzzy extractors for minutiae-based fingerprint authentication. In *ICB 2007, LNCS 4642*, pages 760–769. Springer-Verlag, 2007.

[2] A. Beng Jin Teoh and C. T. Yuang. Cancelable biometrics realization with multispace random projections. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 37(5):1096–1106, 2007.

[3] T. Boult. Robust distance measures for face-recognition supporting revocable biometric tokens. In *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*, pages 560–566, 2006.

[4] G. I. Davida, Y. Frankel, and B. J. Matt. On enabling secure applications through off-line biometric identification. *Security and Privacy, IEEE Symposium on*, 0:0148, 1998.

[5] F. Farooq, R. Bolle, T.-Y. Jea, and N. Ratha, N. A4 Ratha. Anonymous and revocable fingerprint recognition. In R. Bolle, editor, *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–7, 2007.

[6] T.-Y. Jea. *Minutiae-based partial fingerprint recognition*. PhD thesis, Buffalo, NY, USA, 2005. Adviser-Govindaraju, Venugopal.

[7] A. Juels and M. Sudan. A fuzzy vault scheme. In *In International Symposium on Information Theory (ISIT*, page 408. IEEE Press, 2002.

[8] K. Nandakumar, A. Jain, and S. Pankanti. Fingerprint-based fuzzy vault: Implementation and performance. *Information Forensics and Security, IEEE Transactions on*, 2(4):744–757, 2007.

[9] N. Ratha, S. Chikkerur, J. Connell, and R. Bolle. Generating cancelable fingerprint templates. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(4):561–572, 2007.

[10] M. Savvides, B. Vijaya Kumar, and P. Khosla. Cancelable biometric filters for face recognition. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 922–925 Vol.3, 2004.

[11] Y. Sutcu, Q. Li, and N. Memon. Protecting biometric templates with sketch: Theory and practice. *Information Forensics and Security, IEEE Transactions on*, 2(3):503–512, 2007.

[12] S. Tulyakov, F. Farooq, P. Mansukhani, and V. Govindaraju. Symmetric hash functions for secure fingerprint biometric systems. *Pattern Recognition Letters*, 28(16):2427–2436, 2007.