Protecting Biometric Features by Periodic Function-Based Transformation and Fuzzy Vault

Thu Thi Bao Le(✉), Tran Khanh Dang, Quynh Chi Truong, and Thi Ai Thao Nguyen

Faculty of Computer Science and Engineering, HCMC University of Technology, VNUHCM, Ho Chi Minh City, Vietnam
{thule,khanh,tqchi,thaonguyen}@cse.hcmut.edu.vn

Abstract. Biometrics-based authentication is playing an attractive and potential approach nowadays. However, the end-users do not feel comfortable to use it once the performance and security are not ensured. Fuzzy vault is one of the most popular methods for biometric template security. It binds a key with the biometric template and obtains the helper data. However, the main problem of fuzzy vault is that it is unable to guarantee the revocability property. In addition, most of the fuzzy vault schemes are performed on two biometrics modalities, fingerprints and iris. In previous works, authors suggested some cancelable transformations attached to a fuzzy vault scheme to overcome these weaknesses. However, the computational cost of these proposals was quite large. In this paper, we present a new hybrid scheme of fuzzy vault and periodic function-based feature transformation for biometric template protection. Our transformation is not only simpler but also suitable for many kinds of biometrics modalities. The newly proposed fuzzy vault scheme guarantees the revocability property with an acceptable error rate.

Keywords: Biometrics · Fuzzy vault · Biometrics template protection · Cancelable transformations · Face recognition

1 Introduction

As we all know, the traditional authentication schemes usually based on something the user has (such as: smart card) or something the user knows (such as: password, PIN). However, those techniques have several limitations. For example, they cannot distinguish between an authorized user and those who know the correct password [1]. So, we have to choose a strong password and always to keep it in mind.

In recent years, with the rapid development of technologies, biometrics-based authentication systems are becoming potential, because biometrics is literally stuck to an individual, it can prevent the use of several identities by a single individual. The term biometric (from the Greek for bio=life, metric=degree) refers to authentication by means of biological (more accurately, physiological or behavioral) features (such as: face, voice, fingerprint…) [2]. Using biometrics can overcome above limitations. But, it still raises some security and privacy concerns [1]. For example, biometrics is secure but not secret, because voice, face… can be easily recorded and may be misused

A. Hameurlain et al. (Eds.): TLDKS XVI, LNCS 8960, pp. 57–70, 2014.
DOI: 10.1007/978-3-662-45947-8_5

without the user's consent. Another problem is that unlike passwords, cryptographic keys, or PINs, biometrics cannot be changed once compromised. In addition, a user can be tracked by means of cross-matching when he/she uses the same biometrics across all applications and the service-providers collude with each other.

Therefore, the security of biometric template has been emerging increasingly and a lot of research has been done in this field. According to the authors in [3], there are four properties that an ideal biometric template protection scheme should possess:

1. Diversity: the secure template must not be the same in two different applications; therefore, the user's privacy is ensured.
2. Revocability: it should be straightforward to revoke a compromised template and reissue a new one based on the same biometric data.
3. Security: An original biometric template must be computationally hard to recover from the secure template. This property guarantees that an adversary does not have the ability to create a physical spoof of the biometric trait from a stolen template.
4. Performance: the biometric template protection scheme should not degrade the recognition performance of the biometric system.

In biometric template protection, fuzzy vault is considered as a popular method. It binds a key with the biometric template and obtains the helper data for authentication. The template is hidden in the helper data. However, there are some problems that fuzzy vault encounters with. One of these stems from the reason that fuzzy vault cannot provide the diversity and revocability properties. In this paper, that shortcoming is made good by applying a feature transformation in a fuzzy vault scheme. Strictly speaking, the main idea for the marriage of fuzzy vault with feature transformation was introduced in a few recent proposals. Nevertheless, majority of which focus on two biometrics modalities, fingerprints [4, 5, 6], and iris [7, 8]. The transformations for face based fuzzy vault scheme are rare and very complicated. Therefore, this paper will present a hybrid scheme of face based fuzzy vault and feature transformation. Our proposed transformation is not only simpler but also suitable for many kinds of biometrics modalities. The face biometric templates are protected by hidden in the set of chaff points generated by fuzzy vault scheme. Besides, these templates are able to be changed or revoked if the owners have suspicious about being tracked or stolen. Our experimental result will show that the newly proposed scheme guarantees the revocability property with an acceptable error rate.

The structure of this paper is organized as follows. Section 2 provides a brief review of related works. The details of our proposed scheme are described in Section 3. Following them, the evaluation is discussed in Section 4. At last, Section 5 provides the conclusion and future works.

2 Related Works

Biometric template protection is an important issue in a biometric system. Biometric template of a person cannot be replaced or used again once it is compromised. In [9],

the authors presented two approaches to deal with this issue, including feature transformation and biometric cryptosystem.

In the biometric cryptosystem approach, a key is derived from the biometric template or bound with the biometric template. Both the biometric template and the key are then discarded, and only the public helper data is stored in the database. Although public helper data does not reveal any information about the biometrics and the key, it is very useful to regenerate the key from another biometric sample which is closed to the biometric template. The concepts of secure sketch and fuzzy extractor [10], a combination of ANN and secure sketch [11] are kinds of biometric cryptosystem approach. The fuzzy commitment scheme [12] and fuzzy vault [13] are two examples of the key binding approach.

In the feature transformation approach, the biometric templates are transformed before being stored in the database. The transformed templates are hard to be recovered to the original template even with some knowledge of transformation function. Then, the transformed templates are safe to store in the database.

2.1 Fuzzy Vault

Juels and Sudan [13] introduced a construct called a fuzzy vault. The idea is that Alice places a secret k in a fuzzy vault and locks it using a set A of elements from some public universe U. To unlock the vault and retrieve k, Bob must present a set B closed to A, i.e., B and A overlap substantially.

To construct a fuzzy vault, first, Alice selects a polynomial p of variable x that encodes k. Considering the elements of A as distinct x-coordinate values, she computes the polynomial projections for the elements of A. Then, she adds some randomly generated chaff points that do not lie on p. The final set includes real points which lie on p and chaff points. The number of chaff points is far greater the number of real points. It will make the attacker hard to find the real points.

When Bob want to unlock the vault and learn k (i.e., find p), he uses his unordered set B. If B overlaps with A substantially, he will be able to locate many points in the vault that lie on p. By using error-correction coding (e.g., Reed-Solomon), it is assumed that he can reconstruct p and discover k.

There are many researches follow this scheme to construct the vault for fingerprint [4, 5, 6], iris [7, 14], face [15], and some other biometric types.

However, several attacks against fuzzy vaults have been discovered [16, 17]. These are: attacks via record multiplicity, stolen key inversion attack and blended substitution attack. In a stolen key inversion attack, if an adversary somehow recovers the key embedded in the vault, he can decode the vault to obtain the biometric template. Because the vault contains a large number of chaff points, it is possible for an adversary to substitute a few points in the vault with his own biometric features. In this case, the system allows both the genuine user and the adversary to be successfully authenticated. This attack is called blended substitution. In record multiplicity attack, an adversary can access to two different vaults generated from the same biometric data (from two different applications). He can easily identify the genuine points in the two vaults and decode the vault. Thus, the fuzzy vault scheme does not provide diversity and

revocability properties. In this paper, we proposed a hybrid scheme where biometric templates are first transformed based on a periodic function to guarantee diversity and revocability properties.

2.2 Feature Transformation

Transformation functions are classified into two types: invertible (or salting) and non-invertible transformation.

Salting is a method in which the biometric features are transformed using a function defined by a user-specific key or password [9]. With the key, we can invert the transform template to the original one. Therefore, the key needs must be kept secret. Salting can be considered as two-factor authentication in which the users must present both secret key and biometric trail to the authentication system. In [18], the authors generate a user-based random orthonormal $n * n$ matrix A, where n is the size of biometric feature vectors. Then, the original template feature vector x is transformed to a secure domain using matrix product: $y = Ax$. The random orthonormal matrix is generated from a user-based key or token using Gram-Schmidt algorithm[1]. The security in this scheme is relied on the user-specific random matrix which plays a role as a secret key. Another example of salting is using a user-based shuffling key to transform an iris code in [19]. User-based shuffling key which is generated based on users' key or password is an n-bit string. An iris code is also divided into n blocks. The transformation works as follows: beginning from the first to the last block, if the bit i^{th} is 1 (or 0), block i^{th} will be moved to the first (or last) place of the code.

In non-invertible transformation, the biometric template is transformed by a one-way transformation function. A one-way function F is "easy to compute" (in polynomial time) but "hard to invert" [9]. The function F can be public. Non-invertible transformation for fingerprint is proposed in [20]. The authors presented three methods to transform fingerprint. In the first method, the fingerprint image is divided into rectangular grid cells. A shifting map is defined as a transformation function. The minutiae in each cell are moved to a new position which is defined in a shifting map. There may be some minutiae to be shifted to the same cell. Thus, even if the shifting map is public, the attacker cannot infer that a minutia in the transformed template is belonged to which cell in the original template. This is the characteristic of non-invertible transformation. Similarly, in the second method, the fingerprint image is divided into sectors, and the minutiae are shifted among sectors which have the same or nearly the same radius. The third method considers not only the position but also the direction of the minutiae. Scutu et al. [21] proposed a secure authentication based on robust hashing. The idea is to embed each component of a feature vector into a Gaussian function. After that, a number of fake Gaussians are added to hire the true Gaussian.

To all of noninvertible transformation, the most challenge is that how to preserve the similarity of distances among transformed templates and among original tem-

[1] Gram–Schmidt algorithm from Wikipedia: http://en.wikipedia.org/wiki/Gram%E2%80%93Schmidt_process (Oct 2014).

plates. It means that two transformed templates must be closed if the two original templates are closed. This characteristic keeps the error rates of the transformed biometric systems similar to the generic biometric systems, but the transformed biometric systems protect the templates from being compromised.

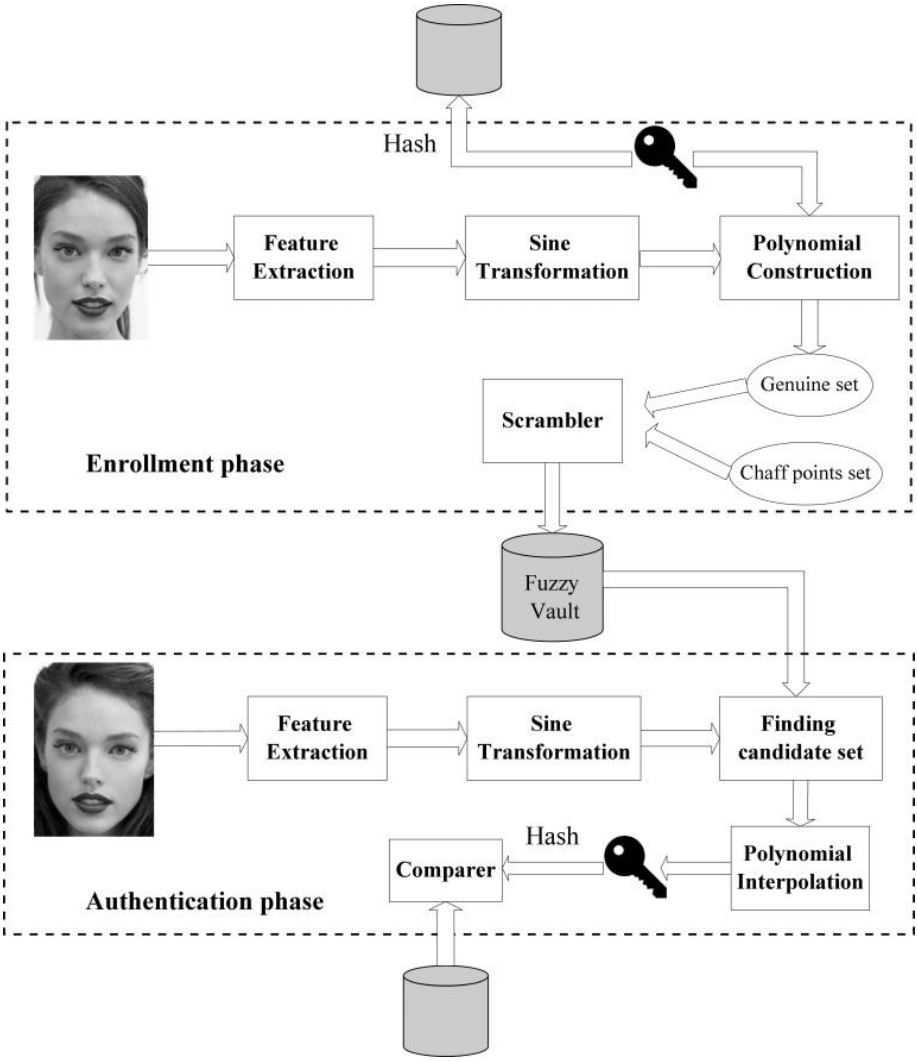

Fig. 1. General architecture

3 The Proposed Scheme

3.1 General Architecture

Our proposed scheme can be applied for many kinds of biometric data whose feature is in vectors. In this paper, we use the face biometric data for demonstration. Our general architecture includes two main phase: enrollment and authentication. In enrollment phase, a set of biometric features is first extracted from the users' face images. After standardized, these features are then transformed by a sine function. The randomly generated key is used to construct a polynomial function and its hashing is stored for matching purpose in authentication phase. The transformed features apply this polynomial function to generate a set of genuine points in fuzzy vault set. To complete the fuzzy vault encoding step, a set of chaff points is also added into fuzzy vault set. After that, all these points are stored in fuzzy vault database.

In authentication phase, sensor will take the image of a user and provide it for the system. This image is also extracted in order to gain the user's biometric feature. The sine transformation is performed on the extracted feature. If this transformed feature has substantial overlap with the enrolled ones, the secret key will be correctly retrieved by the fuzzy vault decoding step. Afterwards, the recovered key is hashed in order to compare with the hashed versions of the keys generated in the enrollment phase. If the new key is matched, the user is authenticated. The overview of the system is illustrated in the Fig. 1.

3.2 Feature Extraction

A feature extraction technique is used to extract the biometric feature. Among many different feature extraction techniques, PCA (Principal Component Analysis), and ICA (Independent Component Analysis) are popular ones for face recognition. In this paper, we choose PCA for its significant outperformance on human face recognition task [22].

In the Eigenfaces method, the PCA is applied to the training set to find a set of standardized face ingredients, called eigenfaces. The training set is a large number of images depicting different human faces, including $\Gamma_1, \Gamma_2, \Gamma_3, \ldots, \Gamma_M$ images. We defined the average face of set as:

$$\psi = \frac{1}{\mathrm{M}} \sum_{\mathrm{i}=1}^{\mathrm{M}} \Gamma_{\mathrm{i}} \tag{1}$$

The difference between each face and the average is shown by vector:

$$\phi_{\mathrm{i}} = \Gamma_{\mathrm{i}} - \psi \tag{2}$$

Then, the covariance matrix is calculated by:

$$\mathrm{C} = \frac{1}{M} \sum_{i=1}^{M} \phi_i {\phi_i}^T = AA^T \tag{3}$$

where the matrix $A = [\phi_1 \phi_2 \ldots \phi_M]$.

We can obtain M eigenvector and eigenvalue of covariance matrix C. Then, we sort out R ($R \leq M$) largest eigenvectors by the corresponding eigenvalues, denoted as: $U = [u_1 u_2 \ldots u_R]_{N^2 xR}$. u_i is the eigenface, these eigenfaces are orthogonal to each other. The image of a user can be transformed to the R-dimensional face space by linear mapping:

$$\Omega = U^T(\Gamma - \psi) = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_R \end{bmatrix}_{Rx1} \quad (4)$$

3.3 Sine Transformation

Before going into details about our transformation, we introduce some notations which are often used in this section.

- $X = \{x_1, x_2, \ldots, x_n\}$: a biometric feature vector extracted from the face image of a user.
- $Y = \{y_1, y_2, \ldots, y_n\}$: an intermediate vector after applied sine transformation on X.
- $X' = \{x'_1, x'_2, \ldots, x'_n\}$: a final transformed vector which is used for fuzzy vault encoding and decoding.

When the feature extraction step completes, the feature vector X of all users will be transformed into an intermediate vector Y by the function (5):

$$\sin(x_i + y_i) = c \text{ (c is chosen randomly)} \quad (5)$$

As we all know, sine function has period of 2Π rad. So, with each value of x_i, we will find exactly one value y_i. But, given a value of y_i, you cannot derive an exactly x_i, because there are many value of x_i corresponding with that y_i. In another words, the sine function is a non-invertible transformation. You can also choose another periodic function has the same characteristic with sine function (such as cosine function). In this paper, we use the sine function to present our works.

The value of y_i is obtained by finding the minimum $y_i > 0$ such as:

$$y_i = \sin^{-1}(c) - x_i \quad (6)$$

It means that the value of y_i lies between $[0; 2\Pi]$. The sine transformation is simply illustrated in the Fig. 2. For example, assume that $c = 1$, $x_i = \Pi/3$, so we have $y_i = 2\Pi/3$.

However, the fuzzy vault requires that the points in vault are disordered. So we cannot use this y_i as x-coordinate values, because if so, it will eliminate the ordered property of feature vector Y. This reason causes the increase of FAR (False Accept Rate). To eliminate this risk, we apply a minor transformation to the vector Y for preserving the order of the elements in fuzzy vault set. The result is the final transformed vector X' generated by the following rules:

$$x'_1 = y_1 \tag{7}$$

$$x'_2 = \begin{cases} y_2 \; if \; y_2 < x'_1 \\ y_2 + k\pi, otherwise \; (where \; k \; is \; minimum \; such \; as \; y_2 + k\pi > x'_1) \end{cases} \tag{8}$$

....

$$x'_n = \begin{cases} y_n \; if \; y_n < x'_{n-1} \\ y_n + k\pi, otherwise \; (where \; k \; is \; minimum \; such \; as \; y_n + k\pi > x'_{n-1}) \end{cases} \tag{9}$$

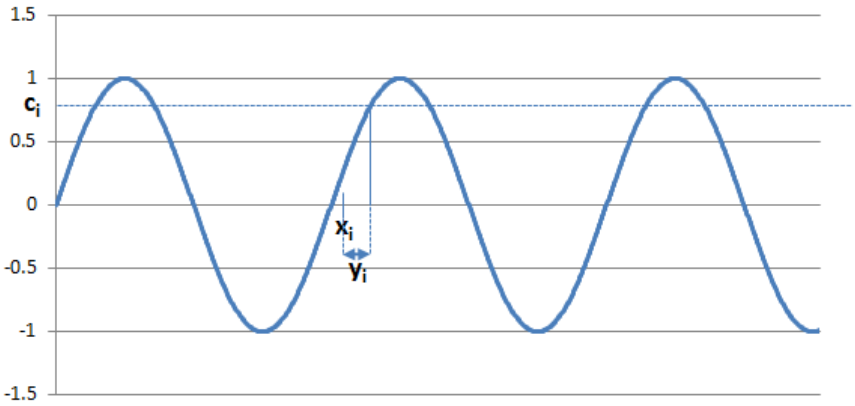

Fig. 2. Sine transformation

3.4 Improve the Performance of Sine Transformation

Easy to see that, the minor transformation (7), (8), (9) in section 3.3 can increase error rate of the system. We have to suppose another way to satisfy the disordered feature of fuzzy vault. The new transformation is:

$$\sin(x_i + y_i) = c_i \; (c_i \text{ is chosen randomly between } [-1, 1]) \tag{10}$$

In addition, the sine function can map two points with large distance to new two closed points. One example is shown in Fig. 3.

To avoid this phenomenon, the value of y_i is obtained by:

$$y_i = \sin^{-1}(c_i) - x_i, \text{ where } \sin^{-1}(c_i) = t + k2\,\Pi \; and \; t = [\tfrac{-\Pi}{2}, \tfrac{\Pi}{2}] \tag{11}$$

It means that, we eliminate another solution: $\sin^{-1}(c_i) = t + k2\,\Pi \; and \; t = [\tfrac{\Pi}{2}, \tfrac{3\Pi}{2}]$. Note that y_i can be a negative number. The value of y_i lies between $[-2\Pi, 2\Pi]$. This new transformation is illustrated in Fig. 4. The output value y_i will be used as an input to fuzzy vault scheme presenting in next sections.

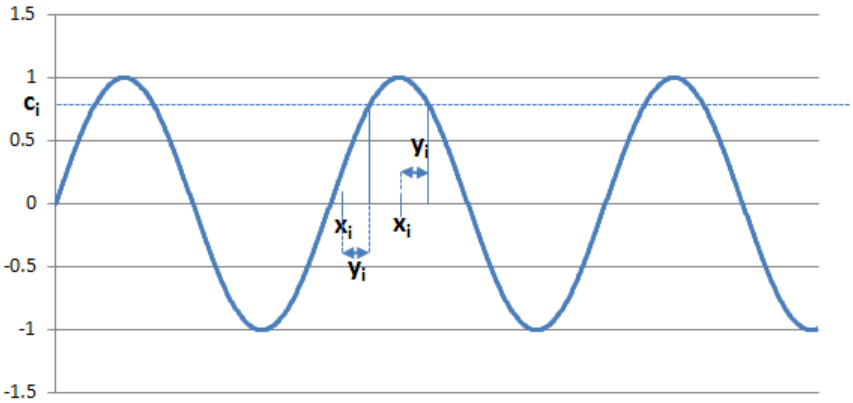

Fig. 3. Mapping two distant points into new two closed points

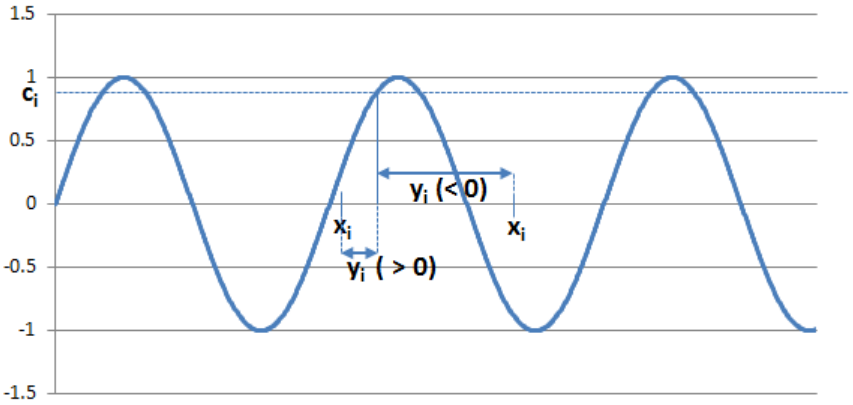

Fig. 4. New sine transformation

3.5 Fuzzy Vault Encoding

The key K is 144-bit which is randomly generated by a random number generator. This key is used for polynomial construction in fuzzy vault scheme. At first, a 8-order polynomial, $f(x) = c_8x^8 + c_7x^7 + \cdots + c_1x + c_0$, needs to be generated. The values of these coefficients are created by truncating the 144-bit K to 9 non-overlapping 16-bit segments. And then, each of them is mapped to the coefficients $c_8 - c_0$ in succession. For example, the first 16 bits is mapped to c_8, and so on. The order of the mapping should be preserved for encoding and decoding of the vault.

To create the genuine points, the polynomial $f(x)$ is evaluated on each of the transformed feature points x'_i. As a result, the genuine set G consists of a set of pairs $\{x'_i, f(x'_i)\}_{i=1}^{M}$, where M is the dimension of the feature vector (also is the dimension of the transformed features). The next step we need to do is to generate the chaff

points set $C = \{a_j, b_j\}_{j=1}^{N_C}$, where N_C is the number of the chaff points and $N_C \gg M$. The randomly generated set C needs to guarantee the following requirements:

$$\begin{cases} |a_j - x'_i| > \Delta & \forall i \ (\Delta \neq 0) \\ b_j \neq f(a_j) & \forall j \end{cases} \tag{12}$$

The final vault V is obtained by taking the union of the two sets G and C. Before storing the set V into the database, we pass it through a scrambler component so that it is hard to figure out which points are genuine points and which are not.

$$V = C \cup G = \{r_k, s_k\}_{k=1}^{M+N_C} \tag{13}$$

3.6 Fuzzy Vault Decoding

The fuzzy vault decoding is mainly based on the Lagrange polynomial interpolation. Assume that the transformed feature vector of a user is $Z = \{z_1, z_2, \dots, z_M\}$. For each z_i, we find the x-coordinate (r_j) of one point in the vault set V in such a way that the r_j satisfy the following rules:

$$\begin{cases} |z_i - r_j| \leq \varepsilon, \ \varepsilon \ll \Delta \ (\varepsilon \text{ is a designed threshold of the system}) \\ |z_i - r_j| \leq |z_i - r_k|, \qquad k = \{1, 2, \dots, M + N_C\} \ and \ k \neq j \end{cases} \tag{14}$$

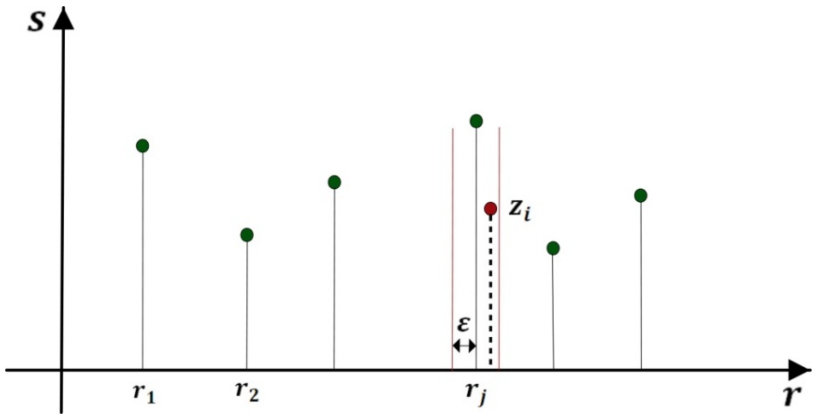

Fig. 5. Fuzzy vault decoding

As a result, the set of points $\{r_j, s_j\}_{j=1}^{L \leq M}$, whose r_j has just found, is the set of the candidate points. These points are ranked by the corresponding nearest distance between $r_j$and z_i. To recover the 8-order polynomial, the Lagrange interpolation technique [2] needs 9 points. We choose the first I points ($9 \leq I$) of the ranked candidate set (the points have the highest possibility to be the real points) and then make the

[2] Wolfram MathWorld, Lagrange Interpolating Polynomial: http://mathworld.wolfram.com/LagrangeInterpolatingPolynomial.html (Oct 2014).

combinations of 9 points from the I ranked candidate set (C_I^9). For each combination, we find one polynomial. Its coefficients are mapped back and concatenated in the same order as encoding phase in order to obtain a 144-bit key K'. To check whether the key K' is matched with the initial K or not, we hash K' and compare the result with the hashed versions of the keys in the database. The authentication is successful if and only if we can recover one key K' matched with K. It means that we do not have to compute all the combinations. Otherwise, if no matched key from all the combinations is found, the authentication is failed.

4 Evaluation

The proposed scheme with the original transformation (section 3.3) and new transformation (section 3.4) is tested with the Face94 database[3]. In the training procedure, we use 100 images of 50 people, i.e. 2 images per person, to construct 40 eigenfaces. Then, we test the scheme with 152 people, including 50 people participated in the training process and 102 new people. Each person has 5 images in which 1 is used to create the vault and 4 are used to unlock the vault. We measure the performance of this scheme, include: FAR (False Acceptance Rate) and FRR (False Rejection Rate)[4].

- The FAR defines the probability that the system incorrectly matches the input pattern to a non-matching template in the database. It measures the percent of invalid inputs which are incorrectly accepted.
- By analogy, the FRR defines the probability that the system fails to detect a match between the input pattern and a matching template in the database. It measures the percent of valid inputs which are incorrectly rejected.

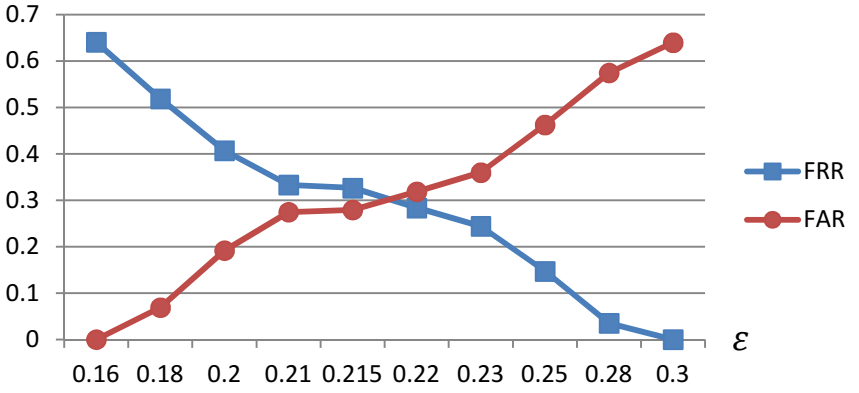

Fig. 6. FAR and FRR of proposed scheme with original transformation

[3] Libor Spacek's Faces94 database: http://cswww.essex.ac.uk/mv/allfaces/faces94.html. (Oct 2014).

[4] FAR and FRR from Wikipedia: http://en.wikipedia.org/wiki/Biometrics (Oct 2014).

The results of are shown in the Fig. 6 and Fig. 7 correspond to original transformation and new transformation. The vertical axis is the error rate which is range [0, 1], and the horizontal axis is the epsilon (ε) as defined in Section 3.6. Epsilon is the minimum distance among points in the vault. Using the original transformation, we can get the acceptable error rates at $\varepsilon = 0.22$. With this value, the error rates are: $FRR \approx FAR \approx 0.3$. Otherwise, using the new transformation, we can get the better error rates $FRR \approx FAR \approx 0.25$ at $\varepsilon = 0.0175$.

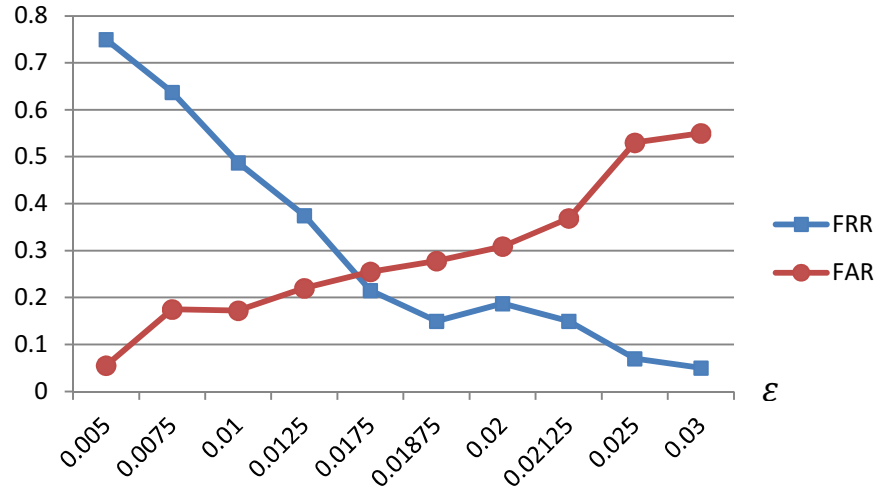

Fig. 7. FAR and FRR of proposed scheme with new transformation

5 Conclusion and Future Works

In this paper, we proposed a hybrid scheme which combines two approaches, namely fuzzy vault and periodic function-based feature transformation, to protect biometric templates. In this scheme, we perform a transformation on the biometric template and then let it as an input to the fuzzy vault. By this way, we can strengthen the fuzzy vault with the revocability property. Our transformation function is non-invertible because the transformation function, i.e. sine function, is periodic with the period 2π. With the knowledge of sin(x), we cannot infer the true value of x.

The results of the evaluation confirm the effective and the practical properties of our scheme to protect biometric template.

The next works we do with our research are to reduce the error rates (FAR, FRR) so that it can be used in practice and to find a proper way to add chaff points to the vault. For the first task, we need more researches on the range of each component of the feature vectors so that we can adjust our parameters, namely the minimum distance among points in the vault and the maximum range of x in the vault. If we can decide these parameters more precisely we can archive a lower error rates in our scheme. The other approach is that we can apply multimodel biometrics schema to

improve performance of the system [23]. For the second task, in this paper, the chaff points are added randomly without the consideration that the transform values (with respect to x value in the vault) of true points have a maximum distance of a predefined number of pi to their nearest true points. This property can be exploited to limit an amount of possible cases in a brute force attack. Therefore, we need to find a better solution to add chaff points to the vault in our scheme.

Acknowledgements. This research is funded by Vietnam National University - Ho Chi Minh City (VNU-HCM) under grant number B2013-20-02. We also want to show a great appreciation to each member of D-STAR Lab (www.dstar.edu.vn) for their enthusiastic supports and helpful advices during the time we have carried out this research.

References

1. Ratha, N., Chikkerur, S., Connell, J., Bolle, R.: Privacy Enhancements for Inexact Biometric Templates. In: Security with Noisy Data, pp. 153–168. Springer, London (2007)
2. Salomon, D.: Elements of Computer Security. Springer (2010). 978-0-85729-005-2
3. Maio, D., Jain, A.K.: Handbook of fingerprint recognition. Springer (2009)
4. Clancy, T C., Kiyavash, N., Lin, D.J.: Secure smartcard-based fingerprint authentication. In: Proceedings of the 2003 ACM SIGMM Workshop on Biometrics Methods and Applications, pp. 45–52. ACM (2003)
5. Nandakumar, K., Jain, A.K., Pankanti, S.: Fingerprint-based fuzzy vault: Implementation and performance. IEEE Transactions on Information Forensics and Security **2**(4), 744–757 (2007)
6. Uludag, U., Jain, A.K.: Fuzzy fingerprint vault. In: Proceedings of Workshop Biometrics: Challenges Arising from Theory to Practice, pp. 13–16 (2004)
7. Lee, Y.-J., Bae, K., Lee, S.-J., Park, K.R., Kim, J.H.: Biometric key binding: Fuzzy vault based on iris images. In: Lee, S.-W., Li, S.Z. (eds.) ICB 2007. LNCS, vol. 4642, pp. 800–808. Springer, Heidelberg (2007)
8. Hao, F., Anderson, R., Daugman, J.: Combining crypto with biometrics effectively. IEEE Transactions on Computers **55**(9), 1081–1088 (2006)
9. Jain, A.K., Nandakumar, K, Nagar, A.: Biometric template security. EURASIP Journal on Advances in Signal Processing **2008**(113) (2008)
10. Dodis, Y., Reyzin, L., Smith, A.: Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 523–540. Springer, Heidelberg (2004)
11. Huỳnh, V.Q.P., Thai, T.T.T., Dang, T.K., Wagner, R.: A Combination of ANN and Secure Sketch for Generating Strong Biometric Key. Journal of Science and Technology, Vietnamese Academy of Science and Technology **51**(4B), 203–212 (2013). ISSN 0866-708X
12. Juels, A., Wattenberg, M.: A fuzzy commitment scheme. In: Proceedings of the 6th ACM Conference on Computer and Communications Security, pp. 28–36. ACM (1999)
13. Juels, A., Sudan, M.: A fuzzy vault scheme. Designs, Codes and Cryptography **38**(2), 237–257 (2006)
14. Vo, T.T.L., Dang, T.K., Küng, J.: A Hash-Index Method for Securing Fuzzy Vaults. In: Eckert, C., Katsikas, Sokratis K., Pernul, G. (eds.) TrustBus 2014. LNCS, vol. 8647, pp. 60–71. Springer, Heidelberg (2014)

15. Wu, Y., Qiu, B.: Transforming a pattern identifier into biometric key generators. In: IEEE International Conference on Multimedia and Expo (ICME), pp. 78–82 (2010)
16. Scheirer, W.J., Boult, T.E.: Cracking fuzzy vaults and biometric encryption. In: Proceedings of the Biometrics Symposium, Baltimore, Md, USA (September 2007)
17. Nguyen, M.T., Truong, Q.H., Dang, T.K.: Enhance Fuzzy Vault Security using Nonrandom Chaff Point Generator. In: Dang, T.K., Wagner, R., Neuhold, E., Takizawa, M., Küng, J., Thoai, N. (eds.) FDSE 2014. LNCS, vol. 8860, pp. 204–219. Springer, Heidelberg (2014)
18. Jin, A.T.B., Ling, D.N.C., Goh, A.: Bio hashing: two factor authentication featuring fingerprint data and tokenised random number. Pattern Recognition **37**(11), 2245–2255 (2004)
19. Kanade, S., et al.: Three factor scheme for biometric-based cryptographic key regeneration using iris. In: Biometrics Symposium 2008 (BSYM 2008), pp. 59–64. IEEE (2008)
20. Ratha, N.K., Chikkerur, S., Connell, J.H., Bolle, R.M.: Generating cancelable fingerprint templates. IEEE Transactions on Pattern Analysis and Machine Intelligence **29**(4), 561–572 (2007)
21. Sutcu, Y., Sencar, H.T., Memon, N.: A secure biometric authentication scheme based on robust hashing. In: Proceedings of the 7th Workshop on Multimedia and Security, pp. 111–116. ACM (2005)
22. Baek, K., Draper, B.A., Beveridge, J.R., She, K.: PCA vs. ICA: A Comparison on the FERET Data Set. In: Proc. of the 4th International Conference on Computer Vision (ICCV 2002), pp. 824–827 (2002)
23. Nguyen, V.N., Nguyen, V.Q., Nguyen, M.N.B., Dang, T.K.: Fuzzy Logic Weight Estimation in Biometric-Enabled Co-authentication System. In: Linawati, Mahendra, M.S., Neuhold, E.J., Tjoa, A.M., You, I. (eds.) CT-EurAsia 2014. LNCS, vol. 8407, pp. 365–374. Springer, Heidelberg (2014)