A Hybrid Template Protection Approach using Secure Sketch and ANN for Strong Biometric Key Generation with Revocability Guarantee

Tran-Khanh Dang[1,2], Van-Quoc-Phuong Huynh[1,2], and Hai Truong[2]
[1]Institute for Application Oriented Knowledge Processing (FAW), Johannes Kepler University Linz, Austria
[2]Computer Science and Engineering, HCMC University of Technology, Vietnam

Abstract: *Nowadays, biometric recognition has been widely applied in various aspects of security applications because of its safety and convenience. However, unlike passwords or tokens, biometric features are naturally noisy and cannot be revoked once they are compromised. Overcoming these two weaknesses is an essential and principal demand. With a hybrid approach, we propose a scheme that combines the Artificial Neural Network (ANN) and the Secure Sketch concept to generate strong keys from a biometric trait while guaranteeing revocability, template protection and noisy tolerance properties. The ANN with high noisy tolerance capacity enhances the recognition by learning the distinct features of a person, assures the revocable and non-invertible properties for the transformed template. The error correction ability of a Secure Sketch concept's construction significantly reduces the false rejection rate for the enroller. To assess the scheme's security, the average remaining entropy is measured on the generated keys. Empirical experiments with standard datasets demonstrate that our scheme is able to achieve a good trade-off between the security and the recognition performance when being applied with the face biometrics.*

Keywords: *Biometric cryptography, biometric template protection, ANN, Secure Sketch, remaining entropy.*

Received July 8, 2015; accepted October 7, 2015

1. Introduction

Nowadays, the biometrics has tremendously risen in the use for security purposes because of its safety and convenience. However, the biometrics faces the challenges of heavy noise and non-cancellable issues. Noisy biometric templates cause difficulties in discriminating different people, while non-cancellable property does not allow user to reuse the biometric trait once a biometric template is compromised.

Our motivation is to construct a biometric-based strong key generator satisfying the low error rates, revocable template together with the high template protection ability which ensures the original biometric traits cannot be exploited even if the biometric key is cracked. Our method is a hybrid approach-based scheme flexibly combining and modifying two well-known models: the Artificial Neural Network (ANN) [8, 17] and the Secure Sketch concept [6]. There are some researches utilizing the ANN for improving the recognition performance [1, 18]. Apart from this purpose, our ANN is applied to primarily assure the revocable and non-invertible properties for the transformed template. A construction of the Secure Sketch concept significantly provides the error correction ability which reduces the false rejection rate for the enroller. In our experiments, face biometrics and the Average Remaining Entropy (ARE) are chosen to evaluate the performance and security of the proposed scheme. The ARE is defined as the average number of bits of the generated key, which distinguishes an individual from a given population. This measurement is bounded by A Lower Bound ARE (LARE) and an Upper Bound ARE (UARE). Aside from that, a trade-off between the recognition performance and the remaining entropy when applying the proposed scheme is also presented. Maintaining a good balance between the efficiency and safety is significant since the higher performance a method achieves, the lower security level it gains. The experiments showed that the proposed method achieves a good trade-off when applying with the face biometrics.

Our scheme can be utilized for both authentication and data protection purposes. For examples, it is able to be implemented as a checking system placed at an office door to verify employees through their faces; or a mobile/desktop application to authenticate users and encrypt data.

The remaining parts of this paper are organized as follows. Section 2 reviews related works, followed by our proposal in section 3 and security analysis in section 4. The next section demonstrates experiments about the performance and the remaining entropy. Section 6 presents some intense discussions about our scheme. Finally, section 7 provides conclusions.

2. Related Work

A biometrics-based security system should possess the four following properties: diversity, revocability, template protection and performance [15]. However, these properties are satisfied with different degrees depending on the method applied by the system. In general, a biometrics-based security method could be classified into two categories: Feature Transformation (FT) and Biometric Cryptosystem (BC) [9].

The FT approach applies a transform function, which uses a random key or a user-specific password to convert the biometric template to the transformed template. The FT approach could provide the diversity and revocability properties since with a biometric template of an individual, many keys and corresponding transformed templates could be issued. The FT is categorized into Invertible or Non-invertible transform.

The BC approach combines Biometrics and Cryptography for the purpose of either protecting cryptographic key by using biometric features called Key Binding or directly generating a cryptographic key from biometric features called Key Generating.

- *Key Binding*: The main advantage is the independency between the cryptographic key and the biometric feature. Therefore, the key is easily reissued with high entropy. However, the drawback of this approach is that the diversity, revocability and template protection properties are not guaranteed in general when the key is compromised. The typical works in this approach are Fuzzy Commitment [11] and Fuzzy Vault [10].
- *Key Generating*: A secret key is directly produced from biometrics. This method is especially appropriate for cryptography applications and template protection property. However, one of the principal challenges of this approach is how to optimize the tradeoff between key stability and key entropy. A typical research for this approach is the combination of secure sketch and fuzzy extractor [6]. Several constructions for the Secure Sketch concept are in [13, 20].

Another approach called hybrid scheme that combines two or more fundamental methods introduced above. Nagar *et al*. [14] and Chafia *et al*. [4] integrated the concept of the Fuzzy Commitment in the construction of fingerprint Fuzzy Vault in order to improve the trade-off between authentication performance and security of the biometric templates. Chen and Chen [5] introduced a hybrid scheme integrating Key Binding and Non-invertible transformation to satisfy diversity, template protection and revocability requirements. Nandakumar *et al*. [16] combines key binding and salting when applying fuzzy vault together with password enhanced the revocability and template protection against cross-matching attack. However, the employment of password in the FT approach causes inconveniences that contradict with the motivations of biometric-based security system.

3. Proposed Scheme

3.1. Overview

An overview of our solution is shown in Figure 1. In the enrolment stage, k feature vectors F_j ($j=1...k$) of type R^m from the Feature Extraction and one random vector R of type R^n generated by the Random Vector Generation serve as training inputs and a shared training output respectively for the ANN. After the training process is completed, all k feature vectors F_j are transferred to the trained ANN to generate the k corresponding outputs O_j of type R^n. In the Quantization step, the random vector R is then converted into a discrete domain vector Enrolled Sample (ES) of type N^n with the help from quantization vector Q of type N^n extracted from the set of vectors O_j. Finally, the Sketch Generator takes ES and the error tolerance capacity δ ($\delta \in \mathbb{N}$) to generate a distance vector D of type Z^n which is considered as helper data. In this stage, the vector ES plays as a secret key while ANN structure data and the two vectors Q and D are helper data stored to recover the vector ES in the authentication stage. A hash version *Hash ES* of ES is stored instead of ES.

In the authentication stage, the trained ANN receives a feature vector F' of type R^m from the Feature Extractor to generate an output vector O' of type R^n. The vector O' is then converted to a vector Authentication Sample (AS) of type N^n with the assistance from vector Q by the Discrete Transformation step. The distance vector D is retrieved from the stored helper data, combined with AS to generate an ES' vector of type R^n by the Recover step. Finally, *Hash ES* will be compared with *Hash ES* to validate whether matching or not.

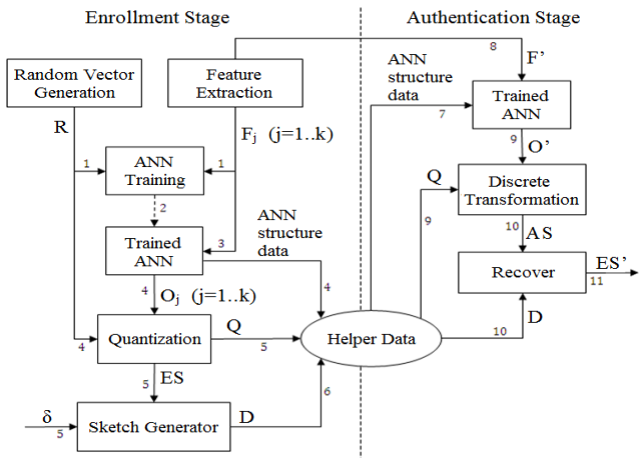

Figure 1. General processing steps in enrolment and authentication stages.

3.2. Feature Extraction and Random Vector Generation

The purpose of these two components is to prepare the training data for the ANN. Feature Extraction is a component extracting the feature vectors in numerical format from a certain biometric trait. The output of the Feature Extraction component that applies Eigenface method [21] on face biometric is k feature vectors F_j ($j=1...k$) of type R^m from k images of an enroller. The scheme does not aim to the method of how to achieve the biometric data. Thereby, the biometric feature extraction process is not mentioned here. Our scheme could be applied to any kinds of biometric traits and any feature extraction algorithms. The only requirement is that the extracted feature data is a real number vector.

The Random Vector Generation prepares a vector $R=(r_1, r_2, ..., r_n)$ of type $[0,1]^n$ which each r_i is generated randomly and independently. k training samples are constructed for ANN training phase. Each training sample includes a feature vector as a training input and the same vector R as training output.

3.3. Artificial Neural Network

The ANN has been widely applied in many biometrics-based recognition systems such as [1, 2, 18]. The ANN is suitable to recognize the enrollers and deals with high noise biometric templates since it can approximate any functions [17] and is highly tolerant of noisy data [8]. The authors in [1] proposed a parallel combination of three ANNs: radial basis function, probabilistic and general regressive neural networks together with a majority voting to identify speakers. The experiment achieves 97.5% accuracy. However, this approach faces with two issues. Firstly, the three neural networks need to be retrained when a new user enrols because set of classes have been changed. Secondly, the system is not suitable for large group since the identification ability is reduced when the number of classes increases if there are many enrollers.

The traditional utilization of ANN applies the regression feature which is trained by many input classes and corresponding outputs to approximate the true function. The trained ANN is then used to classify or predict outputs for new input classes. This utilization may suffer from either the issue of over-fitting or under-fitting. While over-fitting produces excessive output errors for new input classes, under-fitting causes quite similar outputs from different input classes. Although the regression feature is used in our scheme, there are some key alterations of the proposed ANN needed to be addressed:

- Each ANN is trained for recognizing only one enroller and not utilized for prediction purpose. The ANN does not care about the true outputs of new input classes. The ANN's main functionality is to verify the enroller through mapping her/his input feature vectors to a random vector R. After the ANN process is the error correction steps to provide cryptographic ability and to improve the recognition performance with the ERR=2.8%. With our approach, when a new person enrols to the system, a new ANN is created and trained for only that user. All ANNs in the system are independent to each other. Accordingly, the retraining processes for other ANNs are not needed, and the group of enrollers is enlarged easily.
- Over-fitting is not a drawback of our scheme. When the trained ANN tends to be over-fitting, the outputs of new input classes will become excessively variant with the random vector R.
- Under-fitting is a critical issue in the worst case because the trained ANN is equivalent to a constant function that produces the same key for all input classes. However, this issue could be easily avoided by sufficient training iterations and hidden layer with the enough number of nodes.

The ANN is specific for an individual case study. In this paper, the ANN is employed rather than researched. Therefore, how to generally train an ANN to optimally approximate a function is out of scope of this paper. However, in our case study, training properties are established as follows.

- The multilayer feed-forward neural network is applied with one hidden layer having the number of nodes indicated by (1). The number of input nodes and output nodes are respectively the number of dimensions of feature vector and vector R.
- Learning rate=0.1 and Momentum=0.8 to guarantee the coverage and to prevent the local minimum issues. The Sigmoid function $f(x) = 1/(1+e^{-x})$ is chosen for the activation function.

$$No.hidden\,nodes = \sqrt{No.input\,nodes * No.output\,nodes} \quad (1)$$

3.4. Quantization

While After the ANN training, all k feature vectors F_j are presented to the trained ANN to produce k corresponding outputs $O_j = (O_{j1}, O_{j2}, ..., O_{jn})$, ($j=1..k$). It is difficult to expect the outputs O_j to exactly equal the random vector R. However, these outputs could be similar with the vector R at some degree of accuracy. The first purpose of Quantization is to generate a quantization vector $Q = (q_1, q_2, \ldots, q_n)$ which extracts the stability degree of the set of vectors O_j by Equation (2).

$$q_i = Max\, q \,|10^q * O_{ji} = 10^q * O_{li}, \forall j, l \in [1, k],\ j \neq l \quad (2)$$

An example with k = 2, n = 3, O_1 = (0.8365, 0.247892, 0.64829) and O_2 = (0.8364, 0.2478968, 0.6477) then the quantization vector Q = (3, 5, 2).

The second purpose is to convert the vector R from continuous domain to discrete domain by the Equation (3). The Round(x) is a function returning the nearest integer of the input x. The output $ES = (ES_1, ES_2, \ldots, ES_n)$ is a vector which plays as the enrolled sample and the secret key of the system.

$$ES_i = \text{Round}\left(r_i * 10^{q_i}\right) \tag{3}$$

3.5. Sketch Generator

As introduced above, all feature vectors F_j in the enrolment stage are mapped to the secret key ES. In the authentication stage, a feature vector F_e' of the enroller which may not present in the training dataset is not guaranteed to output exactly the ES. Therefore, an error correction step needs to be applied to precisely recover the enrolled sample ES. Our construction bases on the Secure Sketch concept described in Figure 2. Given an ES, the Sketch Generator (SG) produces a distance vector D and discards ES. In the authentication stage, an authentication sample AS which is closed enough to ES could drive exactly to ES with the assistance from D through the Recover component. The construction and terminologies are described as follows.

- *Definition 1*. δ-codebook (which is notated as cb_δ) is a set of codewords spreading along the domain $\mathbb{N}$ with a given error tolerance capacity δ ($\delta \in \mathbb{N}$). The first codeword is zero, and the distance θ between any two consecutive codewords is measured by Equation (4).

$$\theta = 2 * \delta + 1$$
$$cb_\delta = \{c \mid c \bmod \theta = 0,\ c \in \mathbb{N}\} \tag{4}$$

- *Definition 2*. cb_δ^U is a set of all codewords which belong to cb_δ and spread through the domain [0, U] with $U \in \mathbb{N}$.

$$cb_\delta^U = \{c \mid c \le U,\ c \bmod \theta = 0,\ c \in \mathbb{N}\}$$

- *Definition 3*. Map(x, δ) is a mapping function, which returns the nearest codeword of x in the cb_δ. The following formula could be inferred.

$$\text{Map}(x, \delta) = \langle c \mid c - \delta \le x \le c + \delta,\ x \in \text{N},\ \delta \in \text{N},\ c \in cb_\delta \rangle$$

The distance θ guarantees that a value x ($x \in \mathbb{N}$) is always mapped to a unique codeword in a given cb_δ. The public information generated from the Sketch Generator (SG) is a distance vector $D = \left(d_1, d_2, \ldots, d_n\right)$ showed in Equation (5).

$$\forall i \in [1,n] : SG\left(ES_i, \delta\right) = d_i = ES_i - \text{Map}\left(ES_i, \delta\right) \tag{5}$$

- *Lemma 1*. The function Map(x, δ) of type [0, U] → $cb_\delta^{U+\delta}$ is a surjective function.

- *Proof*[1]. The lemma 1 holds if two following conditions are satisfied.

$$\forall x \in [0,U] : \exists! c = \text{Map}(x,\delta) \wedge\ c \in cb_\delta^{U+\delta} \tag{6}$$
$$\forall c \in cb_\delta^{U+\delta} : \exists x \in [0,U] \wedge c = \text{Map}(x,\delta) \tag{7}$$

First, Equation (6) will be proofed by the contrary method with the following assumption.

$$\exists c_1, c_2 \in cb_\delta^{U+\delta} \wedge c_1 < c_2 : \text{Map}(x,\delta) c_1 \wedge \text{Map}(x,\delta) = c_2$$
$$\Leftrightarrow \{\ \textit{definition 3}\ \}$$
$$\exists c_1, c_2 \in cb_\delta^{U+\delta} \wedge c_1 < c_2$$
$$: (c_1 - \delta \le x \le c_1 + \delta) \wedge (c_2 - \delta \le x \le c_2 + \delta)$$
$$\Rightarrow \{\ \textit{arithmetic}\ \}$$
$$\exists c_1, c_2 \in cb_\delta^{U+\delta} \wedge c_1 < c_2 : c_2 - \delta \le x \le c_1 + \delta$$
$$\Rightarrow \{\ \textit{arithmetic}\ \}$$
$$\exists c_1, c_2 \in cb_\delta^{U+\delta} \wedge c_1 < c_2 : c_2 - c_1 \le 2\delta$$
$$\Rightarrow \{\ c_2 - c_1 \ge 2\delta\ \textit{+ 1 according to}\ (5)\ \}$$

False.

Second, Equation (7) will be proofed by the contrary method with the following assumption:

$$\exists c \in cb_\delta^{U+\delta} : \nexists x \in [0,U] \wedge c = \text{Map}(x,\delta)$$

We have:

$\text{Map}(x,\ \delta) = c$
$\Rightarrow$ { definition 3 }
$c - \delta \le x \le c + \delta$
$\Rightarrow$ { $\nexists x \in [0,U]$; so the range [$c - \delta$, $c + \delta$] must be out of [0, U] }
$c - \delta > U$
$\Rightarrow$ { arithmetic }
$c > U + \delta$
$\Rightarrow$ { $c \in cb_\delta^{U+\delta}$ }

False.

The surjective property of the map function guarantees that the $cb_\delta^{U+\delta}$ does not include any redundant and deficient codewords when mapping the values of domain [0, U]. Consequently, the precise number of codewords for the domain [0, U] using the cb_δ is $U + \delta \big/ 2\delta + 1 + 1$.

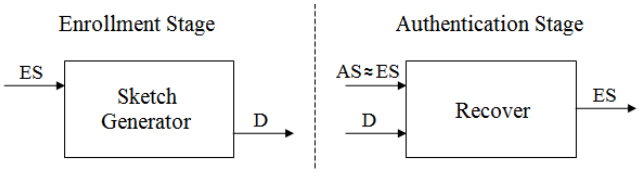

Figure 2. The Secure Sketch concept.

[1]We use a notation exemplified by
⟨Statement 1⟩
⇔ {Condition}
⟨Statement 2⟩
to denote that Statement 2 is inferred from Statement 1 via the Condition

3.6. Sample Recover

In the authentication stage, a feature vector from feature extractor is fetched into the trained ANN to generate an output vector *O*'. With the help of vector *Q*, an authentication sample *AS* is generated in Discrete Transformation step by applying Equation (3) on the output vector *O*'. After that, the Recover step takes *AS* and the distance vector $D = (d_1, d_2, \ldots, d_n)$ to yield a recovered vector *ES*' through Equation (8).

$$\forall i \in [1,n] : ES'_i = \mathrm{Map}(AS_i - d_i, \delta) + d_i \quad (8)$$

- *Lemma 2*. Given an error tolerance capacity δ, the Recover step can precisely reproduce the *ES* if and only if for all dimensions, the difference between AS_i and ES_i is less than or equal to δ.

$$\forall i \in [1,n] : ES'_i = ES_i \Leftrightarrow |AS_i - ES_i| \le \delta$$

- *Proof*: According to Equation (8),

$$\forall i \in [1,n] : ES'_i = \mathrm{Map}(AS_i - d_i, \delta) + d_i$$
$$\Leftrightarrow \{ (5) \wedge \mathrm{Map}(ES_i, \delta) = c \}$$
$$\langle \forall i \in [1,n] : ES'_i = \mathrm{Map}(AS_i - ES_i + c, \delta) + ES_i - c \rangle \quad (*)$$

Two following clauses need to be proved:

1. $\langle \forall i \in [1,n] : |AS_i - ES_i| \le \delta \Rightarrow ES'_i = ES_i \rangle$

   According to (*)

   $\langle \forall i \in [1,n] : ES'_i = \mathrm{Map}(AS_i - ES_i + c, \delta) + ES_i - c \rangle$

   $\Leftrightarrow$ { definition 3 $\wedge |AS_i - ES_i| \le \delta$ }

   $\langle \forall i \in [1,n] : ES'_i = c + ES_i - c = ES_i \rangle$

2. $\langle \forall i \in [1,n] : ES'_i = ES_i \Rightarrow |AS_i - ES_i| \le \delta \rangle$

   This clause is equivalent to the clause $\langle \forall i \in [1,n] : |AS_i - ES_i| > \delta \Rightarrow ES'_i \ne ES_i \rangle$

   According to (*), we have

   $\langle \forall i \in [1,n] : ES'_i = \mathrm{Map}(AS_i - ES_i + c, \delta) + ES_i - c \rangle \Leftrightarrow$

   { definition 3 $\wedge |AS_i - ES_i| > \delta$ }

   $\langle \forall i \in [1,n] : ES'_i = c' + ES_i - c \ne ES_i \rangle \; (c' \ne c)$

From the lemma 2, to recover an enrolled sample, information need to be utilized is the trained ANN, *Q*, δ and *D* which are publicly stored as helper data. However, our proposed model is designed to eliminate the demand for storing the error tolerance δ. The purpose is to strengthen the security of the system while still guaranteeing the recognition performance. The remaining parts of this section will analyze in details the two cases of how to recognize a genuine as well as the safety of *ES* against imposters when δ is not supplied. Let's name the above two cases are the *enroller case* and the *imposter case*. Let δ and δ' be the error tolerance capacity at the enrolment and authentication stage respectively. In both cases, the only way to recover *ES* is to scan the value δ' in $\left[\max|d_i|, \min\left(10^{q_i}/2\right) - 1\right]$. The lower bound $\max|d_i|$ is used because $d_i \in [-\delta, \delta] \forall i \in [1,n]$, and the upper bound $\left(\min\left(10^{q_i}/2\right) - 1\right)$ is chosen to guarantee that there are at least two codewords in a codebook.

- *The enroller case*: $\langle \forall i \in [1,n] : |AS_i - ES_i| \le \delta \rangle$

  The scheme scans the tried δ' with the beginning value $\max|d_i|$. When δ' equals δ, the *ES*' will certainly match the *ES* (follow the lemma 2). The number of trying is small because the δ is shared among all dimensions. Therefore, the secret key *ES* could be easily recovered without the awareness of δ.

- *The impostor case*: $\langle \exists i \in [1,n] : |AS_i - ES_i| > \delta \rangle$
- Lemma 3. Given an *ES* and δ, the first necessary condition for an imposer with a δ' value could recover to *ES* is $\delta' > \delta$.
- *Proof.*

$\langle \forall i \in [1,n] : ES'_i = ES_i \rangle$

$\Leftrightarrow$ { (5) $\wedge$ (8) $\wedge$ arithmetic}

$\langle \forall i \in [1,n] : \mathrm{Map}(AS_i - d_i, \delta') = \mathrm{Map}(ES_i, \delta) \rangle$

$\Rightarrow$ { only one codeword of $cb_{\delta'}$ in range $[AS_i - d_i - \delta', AS_i - d_i + \delta']$ }

$\langle \forall i \in [1,n] : AS_i - d_i - \delta' \le \mathrm{Map}(ES_i, \delta) \le AS_i - d_i + \delta' \rangle \Leftrightarrow$ { (5) $\wedge$ arithmetic }

$\langle \forall i \in [1,n] : |AS_i - ES_i| \le \delta' \rangle$

$\Rightarrow$ { condition of impostor case }

$\delta' > \delta$

- *Lemma 4*. Given an *ES* and δ, the second necessary condition for an imposer with a δ' value ($\delta' \ne \delta$) could recover the *ES* is

  $\langle \forall i \in [1,n] : \mathrm{Map}(ES_i, \delta) \, mod \, (2\delta' + 1) = 0 \rangle$.

- *Proof.*

$\langle \forall i \in [1,n] : ES'_i = ES_i \rangle$

$\Leftrightarrow$ { (5) $\wedge$ (8) $\wedge$ arithmetic}

$\langle \forall i \in [1,n] : \mathrm{Map}(AS_i - d_i, \delta') = \mathrm{Map}(ES_i, \delta) \rangle$

$\Leftrightarrow$ { definition 1 $\wedge$ definition 3 $\wedge$ Let $I_i^{\delta'}$ be the index of the codeword in the $cb_{\delta'}$ which $(AS_i - d_i)$ is mapped to }

$\langle \forall i \in [1,n] \wedge \exists I_i^{\delta'} : \mathrm{Map}(ES_i, \delta) = (2\delta' + 1) I_i^{\delta'} \rangle$

$\Rightarrow$ { arithmetic }

$\langle \forall i \in [1,n] : \mathrm{Map}(ES_i, \delta) \, mod \, (2\delta' + 1) = 0 \rangle$

Because the quantization range of the i^{th} dimension is [0, U_i], with $U_i = 10^{q_i}$, the set of codewords for i^{th} dimension based on δ value is $cb_\delta^{U_i + \delta}$ (according to lemma 1). The necessary condition mentioned in lemma 4 means that every codeword $\mathrm{Map}(ES_i, \delta)$ of *ES* must concurrently belong to two corresponding codebooks: $cb_\delta^{U_i + \delta}$ and $cb_{\delta'}^{U_i + \delta'}$. These shared codewords are common multiples of $(2\delta + 1)$ and

$(2\delta'+1)$. Without loss of generality, the least common multiple can be expressed as $(2\delta+1)\beta_{\delta'}$ with $\beta_{\delta'}=\gamma_1.\gamma_2\ldots\gamma_l$, $l\geq 1$ and γ_l are prime numbers greater than 2 because $(2\delta+1)$ and $(2\delta'+1)$ are odd numbers.

- *Lemma 5.* The probability *P* of the necessary condition mentioned in lemma 4 is less than

$$\left(\frac{1}{\beta_{\delta''}}+\frac{2\delta+1}{\min\left(10^{q_i}\right)+\delta}\right)^n$$

- *Proof.* Let P_i be the probability that the second necessary condition is satisfied for the i^{th} dimension of vector ES. Following the lemma 1, the codebook for i^{th} dimension with respect to δ is $cb_\delta^{U_i+\delta}$ ($U_i = 10^{q_i}$) and contains $\aleph_i=10^{q_i}+\delta/2\delta+1+1$ codewords. The number of codewords belonging to both codebooks: $cb_\delta^{U_i+\delta}$ and $cb_{\delta'}^{U_i+\delta'}$ for the i^{th} dimension is $\eta_i=\frac{\aleph_i(2\delta+1)}{(2\delta+1)\beta_{\delta'}}+1=\frac{\aleph_i}{\beta_{\delta'}}+1$.

  The P_i is estimated as follows.

$$P_i=\frac{\eta_i}{\aleph_i}=\frac{\frac{\aleph_i}{\beta_{\delta'}}+1}{\aleph_i}\leq\frac{\frac{\aleph_i}{\beta_{\delta'}}+1}{\aleph_i}=\frac{1}{\beta_{\delta'}}+\frac{1}{\aleph_i}=\frac{1}{\beta_{\delta'}}+\frac{1}{\frac{10^{q_i}+\delta}{2\delta+1}+1}$$

$$<\frac{1}{\beta_{\delta'}}+\frac{1}{\frac{10^{q_i}+\delta}{2\delta+1}}=\frac{1}{\beta_{\delta'}}+\frac{2\delta+1}{10^{q_i}+\delta}<\frac{1}{\beta_{\delta'}}+\frac{2\delta+1}{\min\left(10^{q_i}\right)+\delta}$$

$$P=\prod_{i=1}^{n}P_i<\left(\frac{1}{\beta_{\delta'}}+\frac{2\delta+1}{\min\left(10^{q_i}\right)+\delta}\right)^n$$

- *Definition 4*: Given an error tolerance δ, an *ES* when mapping to cb_δ is called "δ-quality" on the range $[lower,upper]$ if the second necessary condition mentioned in lemma 4 does not satisfy for every $\delta' \in [lower,upper]$, $\delta \neq \delta'$. Conversely, it is called "δ-unquality" on the above range.

According to lemma 3 and 4, the *ES* will not be recovered in the *imposter case* when the second necessary condition is not satisfied for every $\delta' \in \left[\delta+1,\min\left(10^{q_i}/2\right)-1\right]$, which means that the vector *ES* is "δ-quality" on the range $\left[\delta+1,\min\left(10^{q_i}/2\right)-1\right]$. Fortunately, generating a "δ-quality" *ES* is simply because the probability for an occurrence of "δ-unquality" case, called *TP,* is extremely small. According to the lemma 3 and 5, the *TP* is evaluated as follows.

$$TP<\sum_{\delta+1}^{\min\left(\frac{10^{q_i}}{2}\right)-1}\left(\frac{1}{\beta_{\delta'}}+\frac{2\delta+1}{\min\left(10^{q_i}\right)+\delta}\right)^n$$

Let $R=2\delta+1/\min\left(10^{q_i}\right)+\delta$. The series prime numbers of distinct $\beta_{\delta'}$ values are sorted in ascending order: 3, 5, 7, 3*3, 11, 13, 3*5, 17, 19, 3*7, 23 … One of many upper bounds could be used to estimate the value of *TP*.

$$TP<\left(\frac{1}{3}+R\right)^n+\left(\frac{1}{5}+R\right)^n+\left(\frac{1}{7}+R\right)^n+\left(\min\left(\frac{10^{q_i}}{2}\right)-\delta-4\right)\left(\frac{1}{9}+R\right)^n$$

An example with $\min(q_i)=3$, δ = 10, $n=40$ then $TP < 9.25302*10^{-19} + 5.746*10^{-27} + 3.603*10^{-32} + 3.159*10^{-33} < 9.2531*10^{-19}$. That concludes the probability of a "δ-unquality" case is extremely small, when *n* is large enough. Therefore, it is much more effective and easier to design a testing algorithm than a generating algorithm for a "δ-quality" vector *ES*. If a vector *ES* is checked as "δ-unquality", simply another ES will be generated. The testing algorithm, called QUALITY_TEST, is as follows.

Algorithm 1: QUALITY_TEST

```
Input:
  δ: error tolerance capacity (ETC)
  [lower_δ, upper_δ]: range of ETC
  ES = (ES1 ES2 ... ESn)
Output: True or False
begin
  ci = Map( ESi , δ) , i =(1...n)
  for δ' = lower_δ to upper_δ
    v = 0
    for i = 1 to n
      if ci mod (2δ'+1) ≠ 0 then v = v+1 endIf
    endFor
    if v = 0 then return False endIf
  endFor
  return True
end
```

The δ value must be determined before the enrolment stage finishes. After the register, the enroller will be at a certain degree of performance and security. When the δ value rises, the performance will increase while the security will decrease and vice versa. The proposed scheme can adjust the δ value to satisfy the tradeoff between security and performance without retraining the ANN. The adjustment of δ is hardly unsuccessful due to the extremely small value *TP*. The algorithm δ-ADJUSTMENT for changing the δ value is as follows.

Algorithm 2: δ-ADJUSTMENT

1. *Firstly, authenticate the enroller successfully. As a result, the ES and δ values are recovered.*
2. *Increase or decrease the δ value.*
3. *If (QUALITY_TEST=False), this new δ value is omitted. Turn back to step 2 to try another δ.*
4. *Else, recalculate the new vector D. The new δ value is successfully updated.*

4. Security Analysis

As mentioned in Figure 1, the helper data comprise the trained ANN, a quantization vector *Q* and a distance vector *D*. However, the δ value is not stored due to previous analysis. This section assesses the security of our proposed model based on examining the two aspects of exhausting and neighbourhood search space.

Firstly, the exhausting search space of the secret key *ES* is examined. It is clear that if an attacker discovers the codeword vector that the secret key *ES* is mapped to, *ES* will be exactly obtained with the assistance from the vector *D*. Therefore, the search space of *ES* is degraded to the search space of codeword vectors that *ES* is mapped to. Based on the vector *Q*, the domain for i^{th} dimension of key *ES* is the range [0, 10^{q_i}]. Following the lemma 1, the codebook for the ES_i is $cb_{\delta}^{U_i+\delta}$ ($U_i = 10^{q_i}$) which contains $\left(10^{q_i}+\delta \big/ 2\delta+1^{+1}\right)$ codewords. Because each ES_i could be mapped to an arbitrary codeword in $cb_{\delta}^{U_i+\delta}$, the search space of *ES* is measured by Equation (9). With the example mentioned in sec. 3.6 when q_i = 3 for all 40 dimensions and δ = 10, the search space is 224 bits. In the case of the scanning value δ' is different from the δ ($\delta' \neq \delta$), even a brute-force attack still cannot reveal the secret *ES* because the attacker is exhausting on a different search space which is constructed from a $cb_{\delta'}$. As a result, the search space for the secret key *ES* is practically larger than the measurement in Equation (9) as our schema is designed to conceal the δ value.

$$SeachSpace^{\delta} = \prod_{i=1}^{n}\left(\frac{10^{q_i}+\delta}{2\delta+1}+1\right) \tag{9}$$

Based on the above exhausting search space, the brute force method is obviously not an effective attack. Another kind of striking, called neighbourhood attack may also occur. The neighbourhood attack bases on the intruder's authentication sample *AS* which is generated by the feature vector, the trained ANN and the vector Q to recover the secret key. The intruder will first explore on a neighbourhood space of the codeword vector of the *AS*. If the codeword vector of the *ES* does not exist in this neighbourhood space, the space needs to be expanded to obtain the *ES*.

With a given secret key *ES* and an authentication sample *AS* from an intruder, the Equation (10), which is measured by the number of codewords, indicates the distance between the intruder key and the secret key for each dimension *i*. In order to discover the secret key, the intruder must execute at least $\prod_1^n(DIS_i+1)$ trials; and the maximum is $\left(Max_i^n\left[DIS_i+1\right]\right)^n$ to guarantee that the *ES* will be discovered in *AS*'s neighbourhood search space. As a consequence, the search space, which is measured in bits, is evaluated through a lower bound (11) and an upper bound (12). This is called remaining entropy of the *ES* with respect to the *AS*.

$$DIS_i = \frac{\left|Map\left(ES_i,\delta\right)-Map\left(AS_i-d_i,\delta\right)\right|}{2\delta+1} \tag{10}$$

$$E\left(ES \mid AS\right)_{lower} = \sum_{i}^{n}\log\left(DIS_i+1\right) \tag{11}$$

$$E\left(ES \mid AS\right)_{upper} = n*Max_i^n\left[\log\left(DIS_i+1\right)\right] \tag{12}$$

The Equations (11) and (12) just indicate the remaining entropy of an *ES* with respect to an *AS* of an intruder. With a given secret key *ES* and a set of *N* authentication samples *AS* from a population, the Equations (13) and (14) respectively represent the lower bound and the upper bound of the average remaining entropy of the *ES*. The Equation (15) is the average remaining entropy of the i^{th} dimension.

$$\overline{E_{lower}} = \sum_{i}^{n}\overline{E_i} \tag{13}$$

$$\overline{E_{upper}} = n*Max_i^n\left[\overline{E_i}\right] \tag{14}$$

$$\overline{E_i} = \log\left(\sum_j^N\left(\frac{\left|Map\left(ES_i,\delta\right)-Map\left(AS_{ji}-d_i,\delta\right)\right|}{2\delta+1}+1\right)\Big/N\right) \tag{15}$$

Let *MDIS* be the maximum distance for all DIS_i (i=1..n). The intruder does not know how the *MDIS* is. In order to search the secret key, he supposes an upper bound neighbourhood search space delegated by a corresponding *MDIS*'. Sequentially, this space is explored. If the correct codeword vector is not found, the upper bound neighbourhood space must be expanded to *MDIS*'' which is larger than *MDIS*' and an exploration is performed on this new space (notice that the neighbourhood space with *MDIS*' is not a subspace of neighbourhood space with *MDIS*'').

The Equations from (9) to (15) are evaluated when the δ value is published. However, the δ value is not known by our proposed scheme. The QUALITY_TEST assures at least one element in the codeword vector of *ES* when mapping to cb_{δ} does not belong to the scanning $cb_{\delta'}$. Therefore, the codeword vector of *ES* does not belong to any neighbourhood space of the *AS*'s codeword vector. This means the intruder cannot discover the secret key *ES* when $\delta' \neq \delta$. Because the intruder is vague about the δ value, he must exhaustingly search on a neighbourhood space with a large enough *MDIS* to able to believe that the current δ' is incorrect. Therefore, the actual security level of the scheme is significantly higher than the measurements in Equations (11), (12), (13) and (14).

5. Experiments

In our experiments, the database [19] with 152 different people is used. Each person is represented by

20 facial pictures, each of which is extracted to a 51-dimension feature vector. Fifty two people are randomly chosen as enrollers, and the remaining 100 individuals play as the intruders. For a single test, a 40-dimension random vector R is generated and three images of an enroller are chosen for the training/enrolling phase. The enroller's 17 remaining images and intruders' 2000 images are used for the authentication. For each of 52 enrollers, we performed three single tests and the metrics are measured in average.

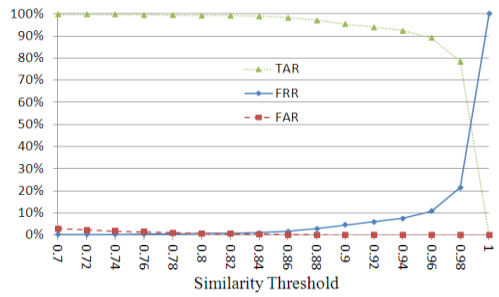

Figure 3. EER on original data according to similarity threshold.

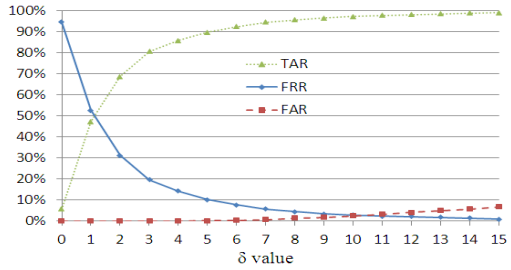

Figure 4. EER on the generated keys according to δ value.

To measure the performance, the Equal Error Rate metric (EER) which is the crossing point between the False Accept Rate (FAR) and the False Reject Rate (FRR) is regularly utilized. The Figure 3 shows the average EER on the original feature vectors through cosine similarity while the Figure 4 presents the same metric on the generated keys according to δ value. The EER achieves about 0.83% at similarity threshold 0.8 for original feature vector and 2.8% at δ=10 for generated keys. A slight reduction on EER occurs after the scheme is applied to compensate for the template protection, revocability and cryptographic ability, which is completely reasonable and acceptable.

The EER reflects a situation when a scheme tolerates with both FRR and FAR concurrently. When a more secure degree is demanded, the point with FAR=0% and the maximum value True Accept Rate (TAR) could gain becomes significant. The value of TAR at this point, called border, actually determines the performance of a scheme. With original feature vector, the average TAR is about 94% at the border (with cosine similarity 0.92) while the average TAR achieves 80% at the border (with δ=3) for generated keys. The reason for the reduction in TAR is that the δ_{border} in each single test is different even the same enroller. Therefore, the δ_{border} in the average case is the smallest δ_{border} from all single tests (the δ_{border} is the value δ at the border). The reason for different δ_{border} values is that the trained ANN in each single test stays at a different degree of over-fitting. In general, the more over-fitting the trained ANN gains, the larger the δ_{border} value is. In each single test, the TAR is much higher at the border.

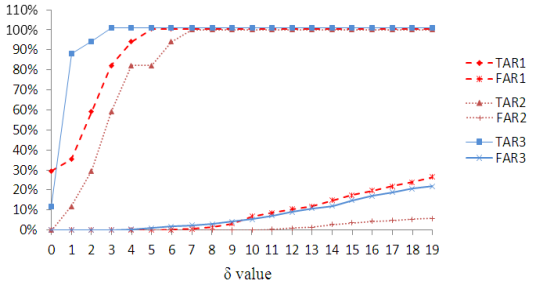

Figure 5. Performance in single test of our method according δ value.

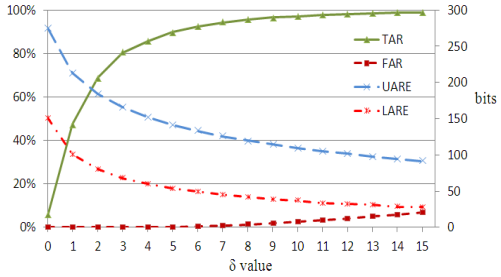

Figure 6. Performance Security trade-off according to δ value.

The Figure 5 shows the typical performance of three single tests from three enrollers. Each single test is depicted by a pair of TARi and FARi lines. All the TARi values in the three single tests achieve 100% at corresponding borders, but the δ_{border} values are 5, 7 and 3 respectively for single tests from 1 to 3. These experiments indicate that this scheme should begin with a small δ for secure reason, and the δ value is then adjusted gradually for better TAR metric.

The Figure 6 shows the trade-off between the performance and the security of our proposed method. At the border with δ_{border} = 3, the TAR is 80%; and the corresponding Average Remaining Entropy (ARE) [LARE, UARE] is about [68, 155] bits. The system achieves higher performance and lower average remaining entropy when error tolerance capacity δ increases, and vice versa. The experiment shows that at the border, high performance and high security could be concurrently achieved.

Finally, the Figure 7 visualizes the diversity of generated keys of a single test through the well-known Parallel Coordinates method. Each coordinate presents a dimension of 40-dimension keys. These keys include the secret key (red bold line) of the corresponding enroller and 2000 other keys (blue line) of intruders. Since different δ values generate different secret keys, performance and security, the error tolerance capacity δ is set at the δ_{border} value. It could be perceptive from

the Figure 7 that the intruders' keys are significantly variant with the enroller's secret key. These generated keys' space is estimated about 104.5 bits. While ARE reflects the density of generated keys, the keys' space demonstrates how diverse the generated keys are.

6. Discussion

In this section, the template protection, diversity and revocability properties stated in [15] are discussed. Let's examine the ability an original feature *F* is revealed when the secret key *ES* has been compromised. The inversive process from *ES* to *F* is obviously based on the public trained ANN with *m* input nodes, *n* output nodes and *k* hidden nodes. The trained ANN is generally considered as a linear equations system with *m* variables and *n* equations. Systems with fewer equations than unknown variables are considered as undetermined linear equation systems which are consistent with infinitely many solutions. Finding the target solution in the solution space is NP-hard [3, 7, 12] that is suitable for the purpose of data security [12]. In summary, our scheme guarantees the template protection property because of the following reasons.

- Our proposed ANN is applied with $m > n$ and $k = \sqrt{m*n}$ (following (1)) which leads to $m > k > n$. Therefore, the trained ANN is a combination of the two sequential undetermined linear systems whose output is the true output vector *O*.
- There are no any evidences to discover the vector *O* of the trained ANN from the compromised *ES* which is just a rounding version of *O*.
- With the type of the transform function P is $[-\infty,+\infty] \rightarrow [0,1]$, a small variance on value *x* may cause a large difference on $P^{-1}(x)$. Therefore, the accumulated fault through the two systems is considerable.

Whenever a secret key *ES* is compromised, a new key is easily generated by applying the scheme with a new random vector *R*. Since the key *ES* is generated easily, randomly and independently with the biometric features, the scheme achieves the diversity and revocability properties.

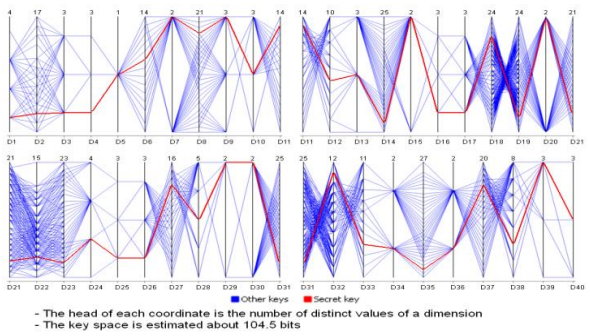

Figure 7. The diversity of generated keys of a single test.

7. Conclusions

In this paper, our main contributions are twofold. First, we propose a hybrid scheme which fuses two main approaches: non-invertible transformation and key generation. The scheme is a flexible combination of ANN for mapping feature vectors to a random vector and a construction of the Secure Sketch concept to generate strong key. Second, we analyze the security based on two aspects of the average remaining entropy and the diversity of generated keys.

The scheme assures the properties of revocability, diversity and template protection. By flexibly adapting the ANN for the purpose of non-invertible transformation, the intruder is incapable to retrieve the original feature information in case the biometric key is cracked. Based on the number of nodes of the ANN, the inversion transform is an undetermined linear system and finding a target solution is NP-hard. Therefore, the template protection property is fully assured. The proposed non-invertible transformation is superior to traditional Feature Transform categories when our scheme eliminates a key/password requirement as a mandatory argument for transformation function.

In addition, the generated secret key is random and renewable. The constructed key space is high diversity with average remaining entropy bounded in [68, 155]. Concurrently, a good recognition performance is maintained with TAR=80% and the corresponding FAR=0%, and EER=2.8% which is approximate to the performance in the case of original feature vectors. The intruder with a wrong δ is incapable to recover the secret key even with the exhausting attack on the neighbourhood search spaces. Therefore, the security of the proposed scheme is significantly enhanced.

Acknowledgements

This research is funded by Vietnam National University-Ho Chi Minh City (VNU-HCM) under grant number B2013-20-02.

References

[1] Almaadeed N., Aggoun A., and Amira A., "Speaker Identification Using Multimodal Neural Networks and Wavelet Analysis," *IET Biometrics*, vol. 4, no. 1, pp. 18-28, 2015.

[2] Arunachalam M. and Subramanian K., "AES Based Multimodal Biometric Authentication using Cryptographic Level Fusion with Fingerprint and Finger Knuckle Print," *The International Arab Journal of Information Technology*, vol. 12, no. 5, pp. 431-440, 2015.

[3] Babaie-Zadeh M., Jutten C., and Mohimani H., "On the Error of Estimating the Sparsest Solution of Underdetermined Linear Systems,"

IEEE Transaction on Information Theory, vol. 57, no. 12, pp. 7840-7855, 2011.

[4] Chafia F., Salim C., and Farid B., "A Biometric Crypto-system for Authentication," *Machine and Web Intelligence*, Algiers, pp. 434-438, 2010.

[5] Chen H. and Chen H., "A hybrid Scheme for Securing Fingerprint Templates," *International Journal of Information Security*, vol. 9, no. 5, pp. 353-361, 2010.

[6] Dodis Y., Ostrovsky R., Reyzin L., and Smith A., "Fuzzy Extractors: How to Gener-ate Strong Keys from Biometrics and Other Noisy Data," *SIAMJ Journal on Computing*, vol. 38, no. 1, pp. 97-139, 2008.

[7] Donoho L., Tsaig Y., Drori I., and Starck J., "Sparse Solution of Underdetermined Systems of Linear Equations by Stage wise Orthogonal Matching Pursuit," *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 1094-1121, 2012.

[8] Han J. and Kamber M., *Data Mining: Concepts and Techniques* 2nd, Morgan Kaufman, 2006.

[9] Jain K., Nandakumar K., and Nagar A., "Biometric Template Security," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, no. 113, 2008.

[10] Juels A. and Sudan M., "A Fuzzy Vault Scheme," *Designs, Codes and Cryptography*, vol. 38, no. 2, pp. 237-257, 2006.

[11] Juels A. and Wattenberg M., "A Fuzzy Commitment Scheme," *in Proceedings of the 6th ACM Conference on Computer and Communications Security*, New York, pp. 28-36, 1999.

[12] Lai J., "On Sparse Solutions of Underdetermined Linear Systems," *Journal of Concrete & Applicable Mathematics*, vol. 8, no. 1, pp. 296, 2010

[13] Li Q., Sutcu Y., and Memon N., "Secure Sketch for Biometric Templates," *in Proceedings of the 12th International Conference on Theory and Application of Cryptology and Information Security*, Shanghai, pp. 99-113, 2006.

[14] Nagar A., Nandakumar K., and Jain K., "Securing Fingerprint Template: Fuzzy Vault with Minutiae Descriptors," *in Proceedings of 19th International Conference on Pattern Recognition*, Tampa, pp. 1-4, 2008.

[15] Nandakumar K. and Jain K., "Biometric Template Protection Schemes: Bridging the Performance Gap Between Theory and Practice," *IEEE Signal Processing Magazine*, vol. 32, no. 5, pp. 88-100, 2015.

[16] Nandakumar K., Nagar A., and Jain K., "Hardening Fingerprint Fuzzy Vault using Password," *in Proceedings of Conference on Biometrics*, Seoul, pp. 927-937, 2007.

[17] Russell S. and Norvig P., *Artificial Intelligence a Modern Approach 1st*, Alan Apt, 1995.

[18] Sharma P., Yadav N., and Arya V., "Pose-invariant Face Recognition using Curvelet Neural Network," *IET Biometrics*, vol. 3, no. 3, pp. 128-138, 2014.

[19] Spacek L., *Libor Spacek's Faces94 database*, http://cswww.essex.ac.uk/mv/allfaces/faces94.html, Last Visited 2014.

[20] Sutcu Y., Li Q., and Memon N., "How to Protect Biometric Templates," *in Proceedings of SPIE Security, Steganography, and Watermarking of Multimedia Contents*, vol. 6505, San jose, 2007.

[21] Turk M. and Pentland A., "Eigenfaces for Recognition," *Cognitive Neuroscience*, vol. 3, no. 1, pp. 71-86, 1991.

Tran-Khanh Dang got his PhD degree (Dr.techn.) in May 2003 at FAW Institute, University of Linz (Austria). Afterwards, he had been working as a lecturer and researcher at the School of Computing Science, Middlesex University in London (UK) since August 2003. Currently, he is an associate professor of computer science and a vice-dean at the Faculty of Computer Science and Engineering, HCMUT, Vietnam. Dr. Dang's research interests include database and information security, privacy protection in location based services, mobile data security, and big data management. He is also the founder of the Data SecuriTy Applied Research (D-STAR) Lab (http://www.dstar.edu.vn). He has published more than 140 scientific papers in international journals and conferences.

Van-Quoc-Phuong Huynh received his MSc degree in Computer Science from Vietnam National University - Ho Chi Minh City University of Technology (VNU-HCMUT). He is currently a lecturer of Information System department, Computer Science and Engineering faculty, HCMUT. His main research interests include biometrics-based and information security, data mining and privacy preserving data mining.

Hai Truong received his MSc degree in Advanced Computing Science from the School of Computer Science of The University of Nottingham, UK. He is currently a lecturer of the Information System department, Computer Sciene and Engineering faculty, HCMUT. His research interests include biometric template security, authentication and recognition, privacy preserving in social networks.