Secure Fingerprint Authentication with Homomorphic Encryption

Wencheng Yang
Security Research Institute, School of Science, Edith Cowan University
Cyber Security Cooperative Research Centre, WA 6027, Australia
w.yang@ecu.edu.au

Song Wang
School of Engineering and Mathematical Sciences
La Trobe University
VIC 3086, Australia
song.wang@latrobe.edu.au

Kan Yu
School of Science
Edith Cowan University
WA 6027, Australia
k.yu@ecu.edu.au

James Jin Kang
School of Science
Edith Cowan University
WA 6027, Australia
james.kang@ecu.edu.au

Michael N. Johnstone
Security Research Institute, School of Science, Edith Cowan University
Cyber Security Cooperative Research Centre, WA 6027, Australia
m.johnstone@ecu.edu.au

***Abstract*—Biometric-based authentication has come into recent prevalence in competition to traditional password- and/or token-based authentication in many applications, both for user convenience and the stability/uniqueness of biometric traits. However, biometric template data, uniquely linking to a user's identity, are considered to be sensitive information. Therefore, it should be secured to prevent privacy leakage. In this paper, we propose a homomorphic encryption-based fingerprint authentication system to provide access control, while protecting sensitive biometric template data. Using homomorphic encryption, matching of biometric data can be performed in the encrypted domain, increasing the difficulty for attackers to obtain the original biometric template without knowing the private key. Moreover, the trade-off between the computational overload and authentication accuracy is studied and experimentally verified on a publicly available fingerprint database, FVC2002 DB2.**

***Keywords*—Fingerprint authentication, template protection, homomorphic encryption, biometrics.**

I. Introduction

Nowadays, biometric authentication systems, which use biometric traits, e.g., fingerprint, face and iris, have been widely deployed in numerous areas such as consumer devices, business, law enforcement and eHealth [1]. Compared to traditional authentication methods based on passwords or tokens, which can be forgotten or lost, the use of biometrics can prevent these from happening. However, while there are benefits to the use of biometrics, biometric templates as well as the user privacy of biometrics should be protected [2]. The exposure of biometric template data could leak a user's critical or sensitive information such as the identity of the user. In addition, biometric data are unchangeable. Once acquired by the attacker, they are compromised forever.

Existing biometric template protection schemes can be broadly divided into three major categories, namely, cancelable biometrics, biometric cryptosystems and homomorphic encryption [3]. Specifically, in cancelable biometrics, the original biometric template can be transformed into another version using a non-invertible transformation function depending on the transformation parameter, also called a key. In cancelable biometrics, the design of non-invertible transformation function plays a main role in striking a good trade-off between security and recognition performance [4, 5]. In biometric cryptosystems, a secret key is either bound with the biometric feature data [6] or directly generated from the biometric feature data [7]. The third category, homomorphic encryption, was first applied to biometrics in [8]. One benefit of using a homomorphic encryption technique, e.g., Paillier homomorphic encryption, in biometrics is that it allows biometric matching to be performed on the encrypted biometric data without degrading recognition accuracy.

All biometric template protection designs are expected to maintain recognition accuracy whilst meeting the following three main requirements [9].

(1) *Renewability*. Once it is found that the protected biometric templates and transformation parameters are leaked, new biometric templates can be re-issued by changing the transformation parameters. The new templates should be different from the old ones, even if they are generated from the same original one.
(2) *Irreversibility*. If the protected templates are obtained by the attacker, the attacker should not be able to reverse the protected templates to acquire the original template.
(3) *Unlinkability*. If multiple protected templates are transformed from the same original template, the attacker is incapable of discovering the linkage between these multiple protected templates.

In this paper, we propose a homomorphic encryption-based scheme to provide biometric template protection for a fingerprint authentication system. An overview of the proposed homomorphic encryption-based fingerprint authentication system is illustrated in Fig. 1. Specifically, the proposed system includes two stages, namely, enrollment and authentication. In the enrollment stage, a user registers his/her fingerprint at the client server (CS), in which a feature vector is extracted from the fingerprint image and encrypted using homomorphic encryption. The encrypted feature vector together with the user ID is sent to the database server (DS)

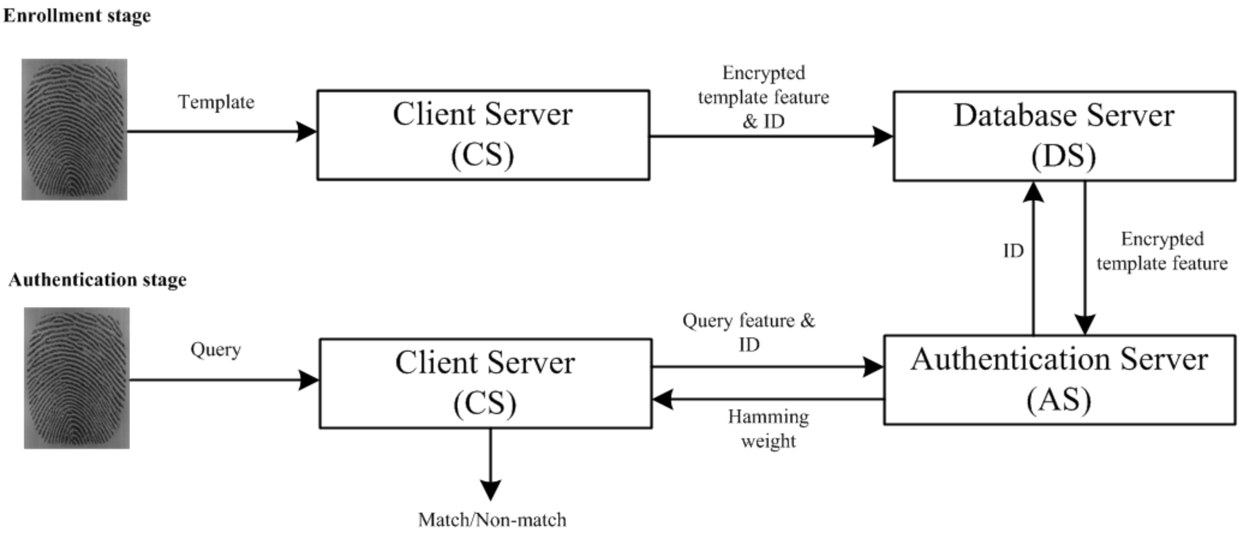

Figure 1. The overview of the proposed homomorphic encryption-based fingerprint authentication system.

and stored as a template. In the authentication stage, a query feature vector is extracted in the same way as the template feature vector at the client server (CS). This query feature vector, together with the user ID, is sent to the authentication server (AS). Using the user ID, the authentication server (AS) can retrieve the corresponding template feature vector from database server (DS) and calculate a Hamming weight (encrypted version) between the query and encrypted template feature vectors. Then, the encrypted Hamming weight is sent back to client server (CS), in which the encrypted Hamming weight is decrypted and treated as an input for calculating the similarity score. Based on the similarity score, a match or non-match report is given. Some major steps of the proposed system are detailed in Section III.

The rest of this paper is organized as follows. Related work of existing biometric template protection methods is presented in Section II. The proposed system is detailed in Section III. In Section IV, some experimental results are discussed. Finally, the conclusion is provided in Section V.

II. Related Work

In this section, some representative approaches of the three major categories of existing biometric template protection schemes are described.

A. *Cancelable Biometrics*

The generation of a cancelable biometric template usually depends on the design of a transformation function, which is expected to be computationally difficult in recovering the original biometric template. For example, Ratha et al. [10] proposed several transformation methods, e.g., Cartesian, polar, or surface folding transformation, to generate cancelable fingerprint templates. Tulyakov et al. [11] used symmetric hash functions to secure the local structure, e.g., triplet, composed by minutiae from a fingerprint image. As the encryption operation is performed on the local structure, the proposed scheme does not require pre-alignment between the template and query fingerprint images. Kho et al. [5] proposed a cancelable fingerprint template by applying a non-invertible transformation function, called Permutated Randomized Non-Negative Least Square (PR-NNLS) to a partial local structure-based minutiae descriptor. In the proposed scheme, the transformation is implemented on a generated dictionary rather than the biometric features, therefore, the performance degradation is not an issue in the proposed scheme.

B. *Biometric Cryptosystems*

In biometric cryptosystems, a secret key is either bound to or directly generated from the biometric feature vectors. Most existing biometric cryptosystems are based on two classic cryptographic preliminaries, called fuzzy commitment [12] and fuzzy vault [13]. Fuzzy commitment is applied to handle the Hamming distance, while the fuzzy vault is used to deal with the set difference, between the biometric template and query feature vectors. Many variants of fuzzy commitment and fuzzy vault schemes have been proposed to secure biometric templates. For instance, Sandhya et al. [14] applied fuzzy commitment to the binary representation of Delaunay neighbor structures extracted from a fingerprint minutiae set. The proposed scheme takes the benefit of stability of Delaunay neighbor structure under biometric uncertainty. Tams et al. [15] proposed an improved fuzzy vault scheme by implementing fused alignment-free features to thwart the correlation attack and possible information leakage caused by auxiliary alignment data. Li and Hu [6] proposed a pair-polar structure-based fuzzy vault scheme. Benefiting from the fine quantization of the pair-polar structure, discriminative information can be retained. Moreover, the correlation attack can be eliminated by transforming the template to different versions in different applications.

C. *Homomorphic Encryption*

One major benefit of using homomorphic encryption in biometric authentication is that it enables the matching of biometric data in the format of ciphertexts and the generated matching result is the same as the result calculated by using the biometric data in the format of plaintexts. In this way, confidentiality can be preserved. Several biometric authentication schemes with homomorphic encryption for template protection can be found in the literature. For example, Yasuda et al. [16] proposed an ideal lattice based homomorphic encryption approach to achieve fast calculation of hamming distance between the template and query biometric feature vectors. Experimental results show that the proposed scheme is significantly faster than many existing methods. Penn et al. [17] exploited Paillier cryptosystem to encrypt messages rather than one bit at a time so as to enhance computational efficiency. The proposed scheme is applied to the iris feature data to evaluate its performance. Morampudi et al. [18] proposed an iris-based biometric authentication system applying full homomorphic encryption to securing the confidentiality of the iris template. To reduce the computational time of encryption and decryption, a batching scheme is used. A related use of homomorphic encryption involves Proof of Retrievability (PoR), where a user can check the integrity of his/her encrypted biometric template(s) and confirm that the templates are both stored accurately and can be completely retrieved. A PoR scheme was implemented by Liu and Zic [19] which is able to prove the retrievability of homomorphically encrypted data by generating probabilistic and homomorphic message authenticators.

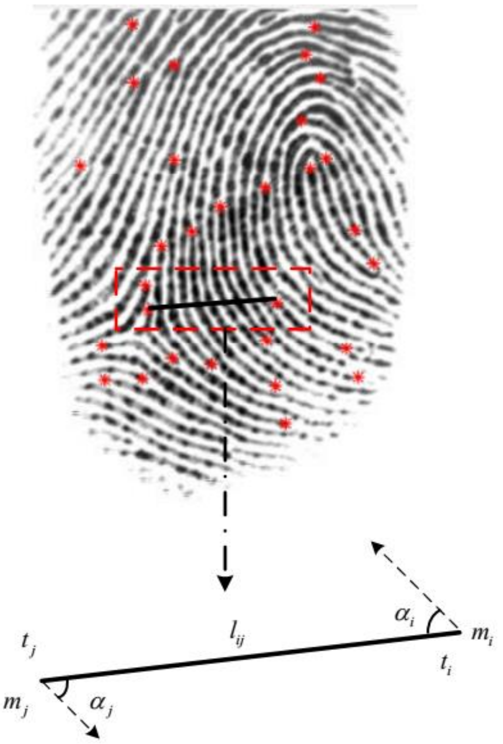

Figure 2. An example of the features defined on the minutiae-pair v_{ij} (adapted from [20]).

III. Proposed Method

Security and privacy concerns arise over unprotected biometric authentication systems, particularly the privacy of biometric templates stored in the databases. Protection should be provided to biometric templates to prevent the private information of users from being revealed, even if they are acquired by the attacker. The proposed homomorphic encryption-based fingerprint authentication system can achieve this goal. There are three major steps of the proposed system, namely, fingerprint feature extraction, homomorphic encryption-based fingerprint data protection, and encrypted matching, as detailed below.

A. *Fingerprint Feature Extraction*

Feature extraction is the first step of the proposed fingerprint authentication system. From the fingerprint image acquired by client server (CS), a set of minutiae $M = \{m_1, ..., m_n\}$ can be extracted from it and each minutia of M can be represented as $m_i = \{x_i, y_i, \theta_i, t_i\}$, where (x_i, y_i) is the location of the minutia in the fingerprint image under the Cartesian coordinate system. θ_i and t_i are the orientation and minutia type, respectively. Feature data extracted from single minutiae for matching can be negatively affected by biometric uncertainty, e.g., image rotation and distortion, which can lead to low recognition performance. To mitigate the negative effect of biometric uncertainty, several stable local structure-based features, such as Minutia Cylinder Code [21], minutiae-pair [22] [20], Delaunay triangulation [23], N-nearest minutiae structure [24], have been proposed in existing research. By balancing the performance and the complexity of applying these local structure-based features, we adopted the minutiae-pair [22] [20] in the proposed method.

Specifically, each minutiae-pair v_{ij} is composed of two minutiae, $m_i = \{x_i, y_i, \theta_i, t_i\}$ and $m_j = \{x_j, y_j, \theta_j, t_j\}$. Similar to definitions in [20], three types of features are defined:

(1) l_{ij}, the edge length between minutiae m_i and m_j.

(2) α_i and α_j, the angle value between the edge l_{ij} and the orientation of minutia m_i and m_j in the direction of counter-clockwise, respectively.

(3) t_i and t_j, the minutiae type, which is represented by 0 or 1.

Each minutiae-pair can be represented by three types of features as $v_{ij} = (l_{ij}, \alpha_i, \alpha_j)$ demonstrated in Figure 2. These real-valued feature representations $(l_{ij}, \alpha_i, \alpha_j)$ are further quantized into binary-valued feature representations in the length of n_l, $n_{\alpha i}$ and $n_{\alpha j}$, respectively, with empirically selected quantization stepsizes for them. In this way, a short binary-valued feature vector λ_{ij} of length $n_v = n_l + n_{\alpha i} + n_{\alpha j}$ +2 is generated for each minutiae-pair vector v_{ij}. This short feature vector λ_{ij} together with all other resulted short feature vectors from the minutiae-pairs is further converted to '1's as elements of a longer binary-valued feature vector $\boldsymbol{\lambda}$ of length n_λ in a similar way to that of [20].

As introduced in the next section, the homomorphic encryption will be applied to the binary-valued template feature vector bit-by-bit and the encryption is a time-consuming process. A long length of n_λ of binary-valued vector $\boldsymbol{\lambda}$ slows the encryption process. To address this issue, we employ a method similar to bio-hashing [25]. Specifically, the binary-valued vector $\boldsymbol{\lambda}$ of size $1 \times n_\lambda$ is further processed by convolving it with a real-valued matrix $\mathbf{m}$ of size $n_\lambda \times n_b$, exprressed as $\boldsymbol{\lambda} \times \mathbf{m}$. The resulted vector is of size $1 \times n_b$ and each element of the resulted matrix is set to '1' if it is larger than zero; otherwise, it is set to '0'. By this means, a binary-valued vector $\mathbf{b} = \{\mathbf{b}[1], ..., \mathbf{b}[i], ..., \mathbf{b}[n_\mathbf{b}]\}_{i \in n_\mathbf{b}}$ of length n_b ($n_b \ll n_\lambda$) is generated. Although the shorter length n_b could cause lower authentication accuracy as less information is kept, it leads to less computational load in the encryption process. This is a trade-off between accuracy and computational time, which will be analyzed and compared in Section IV.

B. Homomorphic Encryption-based Fingerprint Data Protection

Homomorphic encryption is a special case of asymmetric cryptosystems. Similar to asymmetric systems, in homomorphic encryption, there is a public key that is used for encryption and a private key for decryption. Let E_{pub} and D_{pri} be the encryption and decryption functions with public key *pub* and private key *pri*. Given a plaintext *x*, its encrypted cyphertext is represented as $x^* = E_{pub}(x, s)$, where *s* is a random number. $E_{pub}(\bullet)$ is a one-way function, therefore, the attacker is unable to extract useful information about the plaintext *x* with the encrypted value *x** and public key *pub*. The attacker can only obtain the plaintext *x*, with the private key *pri* by decrypting the ciphertext *x** as $x = D_{pri}(x^*)$ [26].

In homomorphic encryption schemes, specific algebraic operations, such as additive or multiplicative operations, performed on a plaintext are equivalent to the algebraic operations performed on the ciphertext. In the proposed method, the Paillier cryptosystem, one of the most commonly used homomorphic encryption schemes, is used to achieve these algebraic operations on encrypted biometric data. There are two properties of the Paillier cryptosystem [27] as described below.

Property One: The product of two cyphertexts, x_1^* and x_2^* can be decrypted to the sum of their corresponding plaintexts,

$$D_{pri}(x_1^* \cdot x_2^* \bmod n^2) = x_1 + x_2 \bmod n \qquad (1)$$

Property Two: An encrypted plaintext with another plaintext being its exponent is decrypted to the product of the two plaintexts,

$$D_{pri}((x_1^*)^{x_2} \bmod n^2) = x_1 \cdot x_2 \bmod n \qquad (2)$$

In the enrollment stage, given a binary-valued template feature vector $\mathbf{b} = \{\mathbf{b}[1], ..., \mathbf{b}[i], ..., \mathbf{b}[n_\mathbf{b}]\}_{i \in n_\mathbf{b}}$, each element $\mathbf{b}[i]$ is encrypted by the Paillier encryption function with the public key *pub*, bit-by-bit, resulting in an encrypted version $E(\mathbf{b}) = \{E(\mathbf{b}[1]), ..., E(\mathbf{b}[i]), ..., E(\mathbf{b}[n_\mathbf{b}])\}_{i \in n_\mathbf{b}}$. The encrypted template vector $E(\mathbf{b})$ is sent and stored in the database server (DS) as a template and will be retrieved for verification purposes in the authentication stage.

C. Encrypted Matching

In the authentication stage, a query vector $\mathbf{b}^Q$ is extracted in the same way as the extraction of the template vector $\mathbf{b}^T$ from a query fingerprint image at the client server (CS). The query vector $\mathbf{b}^Q$ together with its user ID is sent to the authentication server (AS), where it is matched with the encrypted template vector $E(\mathbf{b}^T)$ that is retrieved from the database server (DS) using the user ID. Here, Q represents the Query and T represents the Template.

Hamming distance (HD) is a metric to measure the difference between encrypted template vectors $E(\mathbf{b}^T)$ and the query $\mathbf{b}^Q$. Hamming distance (HD) can be easily computed by Equation (4) with Hamming weight *h* as input. The Hamming weight *h* between $E(\mathbf{b}^T)$ and $\mathbf{b}^Q$ can be calculated by performing xor operation as $\mathbf{b}^* = E(\mathbf{b}^T) \otimes \mathbf{b}^Q$ and counting the '1' bits (summing the values of all the elements) in vector $\mathbf{b}^*$. The xor operation of each corresponding element of $E(\mathbf{b}^T)$ and $\mathbf{b}^Q$ can be expressed as [17],

$$\mathbf{b}^*[i] = E(\mathbf{b}^T[i]) \otimes \mathbf{b}^Q[i] = E(\mathbf{b}^T[i]) + \mathbf{b}^Q[i] - 2E(\mathbf{b}^T[i])\mathbf{b}^Q[i] \qquad (3)$$

where $i = 1, .., n_\mathbf{b}$. Note that the xor operation between an encrypted value $E(\mathbf{b}^T[i])$ and an unencrypted value $\mathbf{b}[i]^Q$ still results in an encrypted vector $\mathbf{b}^*[i]$.

After obtaining the encrypted Hamming weight $E(h)$, the authentication server (AS) sends $E(h)$ back to the client server (CS). $E(h)$ is then decrypted using the user's private key *pri* as $h = D_{pri}(E(h))$ and treated as the input of Equation (4) to calculate the similarity score S_{score} between the query feature vector and template feature vector as

$$S_{score} = 1 - {}^{h}\!/_{n_\mathbf{b}} \qquad (4)$$

If the similarity score S_{score} is equal or larger than the threshold $S_{threshold}$, then the authentication is success and a match report is given, and vice versa

IV. Experimental Results

In this section, the authentication performance of the proposed system is evaluated over a public fingerprint database FVC2002 DB2 [28]. This fingerprint database contains 800 fingerprint images collected from 100 fingers and each finger contributes 8 images. The size of each image is 296×560 pixels. The commercial software Verifinger SKD [29] is used for minutiae extraction in the experiment. The 1VS1 matching protocol used in [5] is also utilized in the experiment. Specifically, in 1VS1 matching protocol, the 1st fingerprint image is compared with the 2nd fingerprint image of the same finger as a genuine comparison. The 1st fingerprint image of each finger is compared with the 1st fingerprint image of remaining fingers as imposter comparison. To avoid duplicate comparison, if a fingerprint image *a* is selected as the template and compared with image *b*, then image *a* is not compared with image *b* again when image *b* is selected as the template. In this way, there are a total of 100 genuine comparisons and 4950 (=100×99/2) imposter comparisons in our experiment to calculate the false acceptance rate (FAR), false rejection rate (FRR) and equal error rate (EER) [30], which are commonly used performance indices in biometric research. The implementation of the homomorphic encryption function is based on the freely available Python Paillier [31]. The experiment is conducted on a desktop with AMD processor AMD FX-8370 Eight-Core Processor 4.01GHz, RAM of 24GB.

In the proposed system, the encryption is performed bit-by-bit to the template feature vector $\mathbf{b}=\{\mathbf{b}[1],...,\mathbf{b}[i],...,\mathbf{b}[n_\mathbf{b}]\}_{i\in n_\mathbf{b}}$. A larger value $n_\mathbf{b}$ means more information is preserved and may lead to better authentication accuracy. However, the homomorphic encryption and decryption are the most time-consuming operations of the whole authentication procedure. Therefore, a larger value $n_\mathbf{b}$ means longer computational time. The effects of different parameter settings of $n_\mathbf{b}$ on the system performance in terms of authentication accuracy and the computational time are explored in the following.

The receiver operating characteristic (ROC) curves in terms of FAR, FRR, and EER of the proposed system under different parameter settings, $n_\mathbf{b}$ = 300 and $n_\mathbf{b}$ = 600, are illustrated in Figure 3 and Figure 4, respectively. It can be seen that with the parameter setting $n_\mathbf{b}$ = 600, the authentication accuracy (EER=8.25%) of the system is better than that (EER=13.94%) when the parameter setting is $n_\mathbf{b}$ = 300.

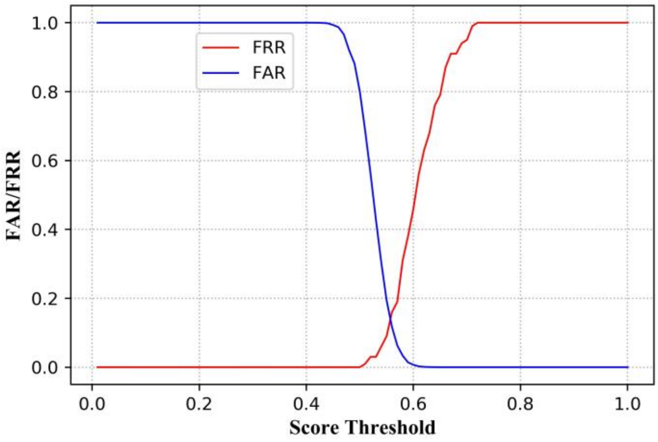

Figure 3. The ROC curve of the proposed system when the feature length $n_\mathbf{b}$ =300.

Moreover, the computational time of the proposed system is demonstrated in Table I. It takes about 1.7 seconds to generate the public key *pub* and private key *pri*. Both keys are stored on the client server (CS). In the enrollment stage, the binary-valued feature vector $\mathbf{b}^T$ as a template is encrypted with the generated public key *pub* and stores the encrypted template feature vector $E(\mathbf{b}^T)$ in the database server (DS). It can be seen that the encryption takes about 248376 milliseconds (ms) with $n_\mathbf{b}$ = 600, while it takes about 123537 milliseconds (ms) with $n_\mathbf{b}$ = 300.

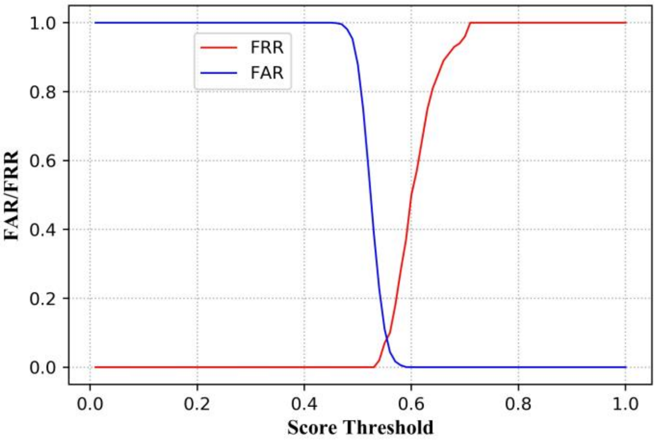

Figure 4. The ROC curve of the proposed system when the feature length $n_\mathbf{b}$ =600.

In the authentication stage, the query feature vector $\mathbf{b}^Q$ is xor-ed with the encrypted template feature vector $E(\mathbf{b}^T)$ and the encrypted Hamming weight $E(h)$ is calculated at the authentication server (AS). $E(h)$ is then sent back to the client server (CS), where it is decrypted and treated as an input to calculate the similarity score S_{score} between the template and query feature vectors. The process of calculating the Hamming weight $E(h)$ and similarity score S_{score} takes about 3.0 seconds. It can be seen that the authentication process is much faster than the enrollment process. This is because the encryption operation is carried out $n_\mathbf{b}$ times as the encryption of the feature vector is bit-by-bit, while in the authentication process, the decryption operation is only performed once on the encrypted Hamming weight $E(h)$.

TABLE I. THE TIME PERFORMANCE IN DIFFERENT OPERATIONS OF THE PROPOSED SYSTEM

Operations	Time
Enrollment - Key generation	≈ 1681 ms (1.7 seconds)
Enrollment - Encryption ($n_\mathbf{b}$ = 300)	≈ 123537 ms (2.1 minutes)
Enrollment - Encryption ($n_\mathbf{b}$ = 600)	≈ 248376 ms (4.2 minutes)
Authentication - Hamming weight and similarity score calculation	≈ 3028 ms (3.0 seconds)

V. Conclusion

In this paper, we implemented a cryptographic technique, called homomorphic encryption, to secure the template data of fingerprint authentication systems. By using homomorphic encryption, biometric matching can be performed in the format of ciphertexts and achieve the same result as the matching with plaintext. The trade-off between the computational time and authentication accuracy is studied in this paper. From the experimental results, it can be seen that the encryption with homomorphic encryption is time-consuming. Efficient homomorphic encryption algorithms should be explored as future work to reduce the computational time, so as to accelerate the deployment of biometric authentication using homomorphic encryption in some real-applications such as mobile health (mHealth) [32, 33].

Acknowledgment

The work has been supported by the Cyber Security Research Centre Limited whose activities are partially funded by the Australian Government Cooperative Research Centres (CRC) Program.

References

[1] J. J. Kang, T. H. Luan, and H. Larkin, "Inference system of body sensors for health and internet of things networks," in *Proceedings of the 14th International Conference on Advances in Mobile Computing and Multi Media*, 2016, pp. 94-98.

[2] M. Barni, T. Bianchi, D. Catalano, M. Di Raimondo, R. D. Labati, P. Failla, *et al.*, "A privacy-compliant fingerprint recognition system based on homomorphic encryption and fingercode templates," in *2010 Fourth IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, 2010, pp. 1-7.

[3] M. Sandhya and M. V. Prasad, "Biometric Template Protection: A Systematic Literature Review of Approaches and Modalities," in *Biometric Security and Privacy*, ed: Springer, 2017, pp. 323-370.

[4] S. Wang, W. Yang, and J. Hu, "Design of Alignment-Free Cancelable Fingerprint Templates with Zoned Minutia Pairs," *Pattern Recognition,* vol. 66, pp. 295-301, 2017.

[5] J. B. Kho, J. Kim, I.-J. Kim, and A. B. Teoh, "Cancelable Fingerprint Template Design with Randomized Non-Negative Least Squares," *Pattern Recognition,* 2019.

[6] C. Li and J. Hu, "A security-enhanced alignment-free fuzzy vault-based fingerprint cryptosystem using pair-polar minutiae structures," *IEEE Transactions on Information Forensics and Security,* vol. 11, pp. 543-555, 2016.

[7] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *SIAM Journal on Computing,* vol. 38, pp. 97-139, 2008.

[8] Y. Luo, S. C. Sen-ching, and S. Ye, "Anonymous biometric access control based on homomorphic encryption," in *2009 IEEE International Conference on Multimedia and Expo*, 2009, pp. 1046-1049.

[9] A. Nautsch, S. Isadskiy, J. Kolberg, M. Gomez-Barrero, and C. Busch, "Homomorphic encryption for speaker recognition: Protection of biometric templates and vendor model parameters," *arXiv preprint arXiv:1803.03559,* 2018.

[10] N. K. Ratha, S. Chikkerur, J. H. Connell, and R. M. Bolle, "Generating cancelable fingerprint templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 29, pp. 561-572, 2007.

[11] S. Tulyakov, F. Farooq, P. Mansukhani, and V. Govindaraju, "Symmetric hash functions for secure fingerprint biometric systems," *Pattern Recognition Letters,* vol. 28, pp. 2427-2436, 2007.

[12] A. Juels and M. Wattenberg, "A fuzzy commitment scheme," in *Proceedings of the 6th ACM conference on Computer and communications security*, 1999, pp. 28-36.

[13] A. Juels and M. Sudan, "A fuzzy vault scheme," *Designs, Codes and Cryptography,* vol. 38, pp. 237-257, 2006.

[14] M. Sandhya and M. V. Prasad, "A bio-cryptosystem for fingerprints using Delaunay neighbor structures (dns) and fuzzy commitment scheme," in *Advances in Signal Processing and Intelligent Recognition Systems*, ed: Springer, 2016, pp. 159-171.

[15] B. Tams, J. Merkle, C. Rathgeb, J. Wagner, U. Korte, and C. Busch, "Improved fuzzy vault scheme for alignment-free fingerprint features," in *Biometrics Special Interest Group (BIOSIG), 2015 International Conference of the*, 2015, pp. 1-12.

[16] M. Yasuda, T. Shimoyama, J. Kogure, K. Yokoyama, and T. Koshiba, "Packed homomorphic encryption based on ideal lattices and its application to biometrics," in *International Conference on Availability, Reliability, and Security*, 2013, pp. 55-74.

[17] G. M. Penn, G. Pötzelsberger, M. Rohde, and A. Uhl, "Customisation of Paillier homomorphic encryption for efficient binary biometric feature vector matching," in *2014 International Conference of the Biometrics Special Interest Group (BIOSIG)*, 2014, pp. 1-6.

[18] M. K. Morampudi, M. V. Prasad, and U. Raju, "Privacy-preserving iris authentication using fully homomorphic encryption," *Multimedia Tools and Applications,* pp. 1-23, 2020.

[19] D. Liu and J. Zic, "Proofs of encrypted data retrievability with probabilistic and homomorphic message authenticators," in *2015 IEEE Trustcom/BigDataSE/ISPA*, 2015, pp. 897-904.

[20] W. Yang, S. Wang, G. Zheng, J. Chaudhry, and C. Valli, "ECB4CI: an enhanced cancelable biometric system for securing critical infrastructures," *The Journal of Supercomputing,* vol. 74, pp. 4893-4909, 2018.

[21] R. Cappelli, M. Ferrara, and D. Maltoni, "Minutia Cylinder-Code: A New Representation and Matching Technique for Fingerprint Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 32, pp. 2128-2141, 2010.

[22] S. Wang and J. Hu, "Alignment-free cancellable fingerprint template design: a densely infinite-to-one mapping (DITOM) approach," *Pattern Recognition,* vol. 45, pp. 4129-4137, 2012.

[23] W. Yang, J. Hu, and M. Stojmenovic, "NDTC: A novel topology-based fingerprint matching algorithm using N-layer Delaunay triangulation net check," in *Industrial Electronics and Applications (ICIEA), 2012 7th IEEE Conference on*, 2012, pp. 866-870.

[24] E. Liu, H. Zhao, J. Liang, L. Pang, M. Xie, H. Chen, *et al.*, "A key binding system based on n-nearest minutiae structure of fingerprint," *Pattern Recognition Letters,* vol. 32, pp. 666-675, 2011.

[25] A. T. B. Jin, D. N. C. Ling, and A. Goh, "Biohashing: two factor authentication featuring fingerprint data and tokenised random number," *Pattern Recognition,* vol. 37, pp. 2245-2255, 2004.

[26] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *International conference on the theory and applications of cryptographic techniques*, 1999, pp. 223-238.

[27] M. Gomez-Barrero, J. Fierrez, J. Galbally, E. Maiorana, and P. Campisi, "Implementation of fixed-length template protection based on homomorphic encryption with application to signature biometrics," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 191-198.

[28] *Fingerprint Verification Competition 2002*. Available: http://bias.csr.unibo.it/fvc2002

[29] *VeriFinger, S. D. K. Neuro Technology*. Available: http://www.neurotechnology.com/verifinger.html

[30] W. Yang, S. Wang, J. Hu, G. Zheng, J. Chaudhry, E. Adi, *et al.*, "Securing Mobile Healthcare Data: A Smart Card based Cancelable Finger-vein Bio-Cryptosystem," *IEEE Access,* vol. 6, pp. 36939-36947, 2018.

[31] (Access date: 2020-06-07). *Python Paillier*. Available: https://github.com/data61/python-paillier

[32] J. Kang and S. Adibi, "A review of security protocols in mHealth wireless body area networks (WBAN)," in *International Conference on Future Network Systems and Security*, 2015, pp. 61-83.

[33] J. J. W. Kang, "An inference system framework for personal sensor devices in mobile health and internet of things networks," Doctor of Philosophy, School of IT, Deakin University, 2017.