JOURNAL OF COLLEGE OF EDUCATION FOR PURE SCIENCES
Vol. 8, No. 3, September 2018
Website: jceps.utq.edu.iq Email: jceps@eps.utq.edu.iq

Hiding Fingerprint Minutiae in FaceFeatures

Ali H. Jazi[1]and Kadhim H. Kuban Alibraheemi[2]

muhah307@gmail.com , khalibraheemi@utq.edu.iq

[1] Department of Computer Science, College of Education for Pure Sciences, Thi-Qar University, Thi-Qar,Iraq.

[2] Department of Computer Science, College of Education for Pure Sciences, Thi-Qar University, Thi-Qar, Iraq

Abstract

In this paper, a new scheme for biometric data security has been proposed.Fingerprint minutiae have been hidden in face templatesat feature level rather than image level.Facial templates areconstructed by extracting small block of coefficients matrix resulting from applying DCT on face images. First, we propose a new encryption algorithm based on arithmetic coding to encrypt the fingerprint minutiae points. Then, minutiae codes are embedded in facial feature vectors using a new embedding method based on XOR to produce a *stego-template*. The influence of hiding process on the performance of face recognition system was tested. The experiments showed that the degradation was not raised in system accuracy. So, the proposed scheme can be implementedto secure the sensitive data in a multibiometric system that combines face and fingerprint to authenticate individuals.

Keywords: Arithmetic Coding, DCT, Stego-template, Fingerprint Minutiae, RSAC, Steganography.

1. INTRODUCTION

A biometric authentication system is employed by acquiring the biometric data of a subject (e.g., face image), extracting a feature set from the data (e.g., eigen-coefficients), and comparing the feature set against the templates stored in a database for the purpose of authenticating person's identity[1].

Multimodal biometric systems, which employ several biometric attributes rather than one, has become an important research area to overcome several weaknesses in the unimodal system, such as the unacceptability of performance in large-scale applications[2,3].

Another issue encountered when using biometrics which is how to protect the biometric data(stored in the database or transmitted across an unsecure channel) from unauthorized accesses. However, if biometric data have been exposed, a serious problem about confidentiality and integrity of the biometric data can be raised.

In recent years, significant attempts have been focused on developing specific protection techniques for biometric data. Cryptography and steganography are among these techniques [3,4].Despite the relative high security provided by cryptographic systems, theprotected template produced by cryptography schemesare noise in appearance, and may lead the attacker's interest in violating the privacy of protected templates.For steganography, however, it is possible to maintain the main features of the biometric. The protected template with hidden secretis difficult to be distinguished from the original templates, and can be used directly for the biometric comparison [5].

At present, the use of cryptography and steganography together significantly improves information security. Since steganography data influences the recognition accuracy of the biometric system due to image quality degradation, its effect must be taken into account in the implementation of steganography techniques.[3,6].

Fingerprint and face are widely used in authentication systems. In this paper, we have focused ondeveloping data hiding technique to protect the privacy and secrecy offace andfingerprint templates.In fact, most of the face recognition systems only store face features for the purpose of storage and speed. It is necessary and important to develop proper data hiding techniques in the feature level of face template, instead of working on image level of the face. We simply alter the coefficients of facial template to embed fingerprint minutiae. The alteration should not reduce system accuracy.

To the best of our knowledge, except the study introduced bySheng Li et. al. (2017) as presented in[5], we have not noticedany literature that is developed for data hiding inbiometric templates in the feature level. In [5], the study was designed to hiding a secret datain fingerprint template, but in cost of reducing the accuracy. Therefore, any study involving data hiding in facial feature level was not

investigated.

However, a scheme for embedding the watermark payload in the compressed stream of fingerprint images is proposed by Ratha et al (2000). The embedding operates in the compressed stream generated by the wavelet scalar quantizer (WSQ) standard. The WSQ coefficients are altered in a way to account for possible image degradation [7].

Yongwha et al. (2005) hided biometric data with watermarking techniques, they implemented two scenarios. In the Scenario 1, they use a fingerprint image as a cover and hide facial features into it. On the contrary, they hide fingerprint features into a facial image in the Scenario 2. Based on the experimental results, they confirm that the Scenario 2 is superior to the Scenario 1 in terms of the verification accuracy of the watermarked image [3].

N. Chaudhary et al. (2013) suggested enhancing security of multimodal biometric authentication system by implementingwatermarking. The fingerprint and face images are initially transformed using Discrete Wavelet and Discrete Cosine Transformation and then watermarked using Singular Value Decomposition.

The paper is organized as follows: Section 2 describes the use of DCT as means of feature extraction and the concept of arithmetic coding. Section 3. Introduces the proposed system, 4. Experimental results are represented in Section 3. Finally, some conclusions are made in Section 5.

2. **PRELIMINARIES**

In this section, we introduce some concepts related to discrete cosine transform (DCT), fingerprint minutiae, and arithmetic coding.

2.1 DCT coefficients as facial features

An image is typically locally correlated in spatial domain in which one can predict the neighbor pixels of a particular spatial location with high sureness. However, this is not the case when the image is transformed by orthogonal transform such as DCT. The transformed or decorrelated feature provides little information about the correlation between its neighbors [8]. This fact can be exploited to exclude the statistical attack typically used to detect hidden information in spatial domain of image.

The basic ideahere is to use the discrete cosine transform (DCT) as a means of feature extraction forface recognition. The DCT is computed for the face image, and a small subset of the DCT coefficients are obtained as a feature vector describing that face [9].

Given that$I(r,c)$ represent an image with the size of N*N, $C(u,v)$ be the coefficients obtained from the two-dimenion DCT of the image, then:

$$C(u.v) = \sum_{r=0}^{N-1}\sum_{c=0}^{N-1}\left\{I(r,c)\alpha(u)\alpha(v) \times \cos\left[\frac{(2r+1)u\pi}{2N}\right] \times \cos\left[\frac{(2c+1)v\pi}{2N}\right]\right\}$$

Where,

$$\alpha(u) = \left\{\begin{matrix}\sqrt{1/N}u = 0 \\ \sqrt{2/N}u = 1,2,\cdots,N-1\end{matrix}\right\}$$

$$\alpha(v) = \left\{\begin{matrix}\sqrt{1/N}v = 0 \\ \sqrt{2/N}v = 1,2,\cdots,N-1\end{matrix}\right\} \quad \ldots\ldots(1)$$

The larger coefficients are located in the low frequency area (the upper left cornerof DCT matrix C) and represent the feature of the imageas shown in Figure (1).

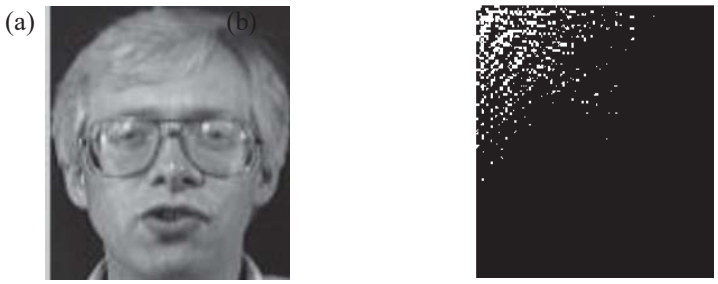

Figure 1. a- face image taken from AT&T database, b- DCT coefficient matrix of (a)

Therefore, facial features in face images are represented in this area of matrix C, for example, the most of facefeatures, such as eyes, nose, and mouth can be represented by this area[9].

An important question would be raised that is; how many coefficients should be selected? One may choose to use more DCT coefficients to represent faces. However, more coefficients do not necessarily imply better recognition performance, because by adding them, they may be representative of more irrelevant information. Experimentally, there is a slight decrease in recognition accuracy as we go to higher numbers of coefficients [10]. Optimal number of selected coefficients is varied according to different databases.

2.2 Fingerprint Minutiae

The fingerprint minutiae records the detailed features ofa fingerprint. Typically, a ridge can suddenly come to a certain end to bea ridge ending, or be splitted into two ridgesto produce a bifurcation. The fingerprint minutiae are themost popular features in fingerprint recognition. As illustrated in Figure (2) and Table (1), fingerprint minutiae can be represented by the triple :T U ,; where :T U is the coordinate (location) of each minutiae point (ridge ending or bifurcation) and is the corresponding

direction [5].

(a) (b)

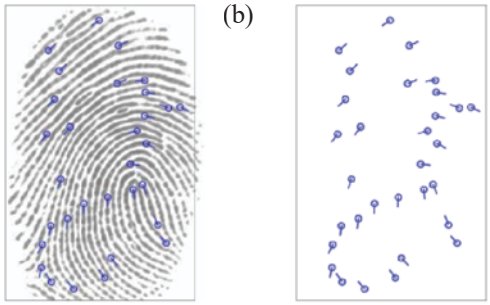

Figure 2. a- fingerprint with minutiae, b- minutiae points

x	y	θ
122	115	210
160	130	300
72	160	270
209	169	120
139	192	330
…	…	…

Table 1.Minutiae sample

2.3Arithmetic coding

Arithmetic coding (AC) was designed originally as an optimal data compression technique that requires accurate knowledge of the source statistics [11]. It has been adopted for Compression Standard, including JPEG2000 and H.264 [12]. The algorithm is also easily adaptable to changing source statistics and it is not necessary to order symbol probabilities in any particular order, and can encode an entire message into a single floating number[11]. Because of the powerful features that are introduced above, this technique has been selected in this study.

Arithmetic coding consists of two main parts: a *model* that estimate the probability distribution of the input data, and a *coder* which produces the output [12].

Good compression ratio can be obtained if the model is derived from the message that would be compressed. These probabilities can be determined by counting the symbol frequencies in representative samples of message to be transmitted. Such a *fixed model* is shared in advance to both encoder and decoder, after which it is used for many messages, so it may perform very poorly [12].

Alternatively, an *adaptive model* recalculates the probability of the symbols as the encoding progresses, and dynamically updates the model after a certain number of symbols are coming. Adaptive model does not scan the string in advance and does not need to send the symbol probabilities separately,but it requires an initial model in order to operate correctly. The first string of source symbols is encoded with the initial model, and then the statistics of this string are used to adapt the model to encode the next string of symbols[11].

The coder utilizes the model to introduce the output as floating point number between 0 and 1.The following example illustrates the basic idea of arithmetic coding:

Suppose the alphabet of a source is {a, b,c}, with frequencies and corresponding probabilitiesas detailed in Table (2), and the string to be encoded is *'aba'*. The range of initial interval is [0,1) and each

source symbol occupies a subinterval in that range according to its probability. The interval is successively subdivided when each new source symbol is encoded. The first step is to select the subinterval corresponding to the first symbo1. In this case, the first symbol is *'a'* and the corresponding interval is [0,0.5). This will be the new interval for the next iteration and is again divided into three subintervals, proportional to the symbol probabilities. The next input symbol to be encoded is *'b'*, and the corresponding interval is [0.25 , 0.375). This process is repeated until all stringsymbols have been encoded. Finally, the message *'aba'* could be assigned theinterval [0.25 , 0.3125) as shown in Fig. 3, and any decimalfraction within that interval can be used to denote themessage.

Table 2. The model of previous example

Symbol	Frequency	Probability	range
a	12	0.5	0 – 0.5
b	6	0.25	0.5 – 0.75
c	6	0.25	0.75 – 1

When the sequence becomes longer, the interval needed to represent it becomes smaller, and the number of bits needed to determine that interval grows.

Witten and Cleary tried to combine encryption with compression in one scheme using adaptive arithmetic coding. They believed that if the model is kept secret and known onlyto the authorizedparties, other recipients without the knowledge of the model cannot retrieve the encoded message and it is still secure since the adaptive nature of the model acts like a large size key that the intruder cannot guess[13,14].However, it was shown that the traditional implementation of AC, either using fixed model or adaptivemodel, cannot offer satisfactory level of security [12].

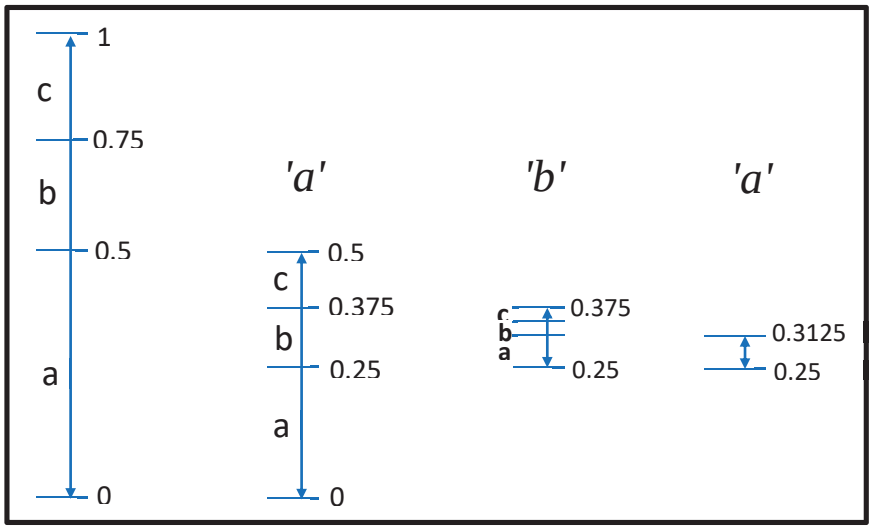

Figure 3. Graphical representation of encoding the string *'aba'* using arithmetic coding

KSAC is a compression and encryption scheme that splitsthe interval corresponding to a given symbol to subintervals, but keep the sum of subintervals equal to original interval using secret interval splitting key. Randomized arithmetic coding (RAC) is another approach achieves confidentiality by changing the order of the intervals corresponding to the symbols according to a secret key.It was used to encrypt JPEG2000 images.The purpose of interval swapping is to confuse the output of AC, and the scheme is believed tobe safe if a cryptographically secure keystream is used to control this swapping [15].

However, Raj S. Katti et al. proved that RAC is not secure. It certainly is not chosen-plaintext or chosen-ciphertext secure.[16].Jakimoski et al. [15] analyzed that KSACis vulnerable to known-plaintext attack and pointed out that the randomized arithmetic coding is not as secure as the authors claimed.

All of the above systems have beendevelopedunder a certain condition, namely preserving compression ratio for multimedia of real time application. This condition restricts the developers from introducing secure schemes sufficiently. In the next section, we propose a new encryption scheme using arithmetic coding appropriate to our purpose, freed of compression constraints. Then, we will utilize the proposed scheme to encrypt fingerprint minutiae.

3. PROPOSED SYSTEM

To ensure sufficient level of security in moltimodal systems, we have proposed two algorithms, each is complementaryto other; one toencrypt fingerprint minutiae, and the other toembed the encrypted minutiae in facial features extracted using DCT.

3.1Ramify–and-Shift Arithmetic Coding (RSAC)

In this paper, we have proposed a new encryption schemewithout consideration of the compression issue.This assumption will allow us to use large alphabets rather than small alphabets that have led to aneasy attack on previous encryption schemes.Large alphabet used here does not imply using more symbols than is required. Instead, 'large' means that each symbol in the model has multiple copies with distinict intervels.Intervals associated with each symbol are ramified, except for the larger interval,and copies of the corresponding symbols are assigned to each subinterval. Anysubintervalassociated with any copy of symbol to be encoded can be chosen.

The principle of the proposed algorithm is to encrypt each sequence of symbols with a different model, and the length of sequence should be relatively small.

To realize that, suppose a source generates symbols from the alphabet $\{x_1.x_2.\ldots.x_N\}$. The frequency tabel is initialized by assigning random occurrences of alphabetic symbols. After that, all frequencies,

except larger one, are ramified to smaller pieces, and each piece is labeled as same as their original symbols. Hence, the new table will contain multiple copies of all the symbols except a symbol that has biggest frequency. Now, permutation step must be done to rearrange the symbols in the table. The resulting table represents the initial table by which all other tables will be derived, one table for each (sub)sequence. This table must be secret and shared between the sender andreceiver. The derivation method is as follows:

Let M be the size of ramified frequency table. The parties agree on two keys, one declared and another secret. $(i.MC)$ is the declared key, where $i = 1.2. \ldots .2N - 1$ is shifting process index, andMC is the counter of the models that have been derived for shifting case$i$. The decoder sends the declared key along with the ciphertext. $(sc1.sc2.p)$ is the secret key, where $sc1$ can be calculated by:

$$sc1 = i * p\ (mod\ M) \qquad \ldots\ldots(2)$$

and $sc2$is any relatively small number,but not equal to $sc1$,p is a prime number larger than M. To reproduce a new model:

step1: shift the frequency column and the symbol column by amount $sc1(MC + 1)\ mod\ (M)$and$sc2(MC + 1)\ mod\ (M)$, respectively.

Step2: Add $sc1(MC + 1)$to all frequencies.

Step3: Replace all other copies of the symbol in which the largest frequency has stabilized by copies of the previous largest frequency symbol.

Step4 : Increase the models counterMC by one and calculate the probabilities.

The resulting model can be used to encode a sequence of an agreed length between the two parties, so that if the length exceeds the predetermined length, the sequence should be broken down to subsequences of the desired condition. This is done to prevent an attacker from encrypting long sequences using same model which may lead to a successful chosen-plaintext attack. The symbol that has largest frequency is added to the beginning of the sequence to be coded.

Models counter MC for ithshifting case should be retained if we desire to derive another model for this shifting case.

To demonstrate the idea, suppose that we have the model represented in Table (2).After ramifying the frequencies, frequency of (a) remains{12}, frequencies of (b) are {1.2.3}, and frequencies of (c) are {2.4}. After permutation step, the frequency table is represented in Table (3). Thenwe select $i = 2.\ sc2 = 1.\ p = 11, M = 6, sc1 = 2 * 11\ (Mod\ 6) = 4$, and $MC = 0$.The derived model according to scheme above, is representedin Table (4).

Symbol	Frequency
c	4
b	2
a	12
b	1
c	2
b	3

Table 3. Frequency table

Symbol	Frequency	Probability	Range
b	13	0.433	0 – 0.433
c	2	0.067	0.433 – 0.5
a	3	0.1	0.5 – 0.6
a	4	0.133	0.6 – 0.733
a	5	0.167	0.733 – 0.9
c	3	0.1	0.9 – 1

Table 4. Derived model from frequency table in Table3

Now, before encoding the sequence *'aba'* , the symbol *'b'* should be added to the beginning so thatfinal sequence becomes*'baba'*.

3.2 Hiding Algorithm

Suppose that$I_i$is a facial image, where $i = 0,1, \dots, V-1$, V is the number of images belong to same person.First, we extract facial features using DCT and put them in one vector. Next, we encrypt minutiae points using RSAC to get minutiae code(s). Finally,we embed minutiae code(s) in the facial feature vectors using a new embedding scheme.

A. Feature Extraction

Step1: Resize all facial imagesI_i to the size $n \times n$.

Step2: For every I_i, configureDCT coefficients matrix C_i using DCT to whole image.

Step3: Extractfirst block of size$k \times k$ from coefficients matrixC_i, and reshape the extracted block to one dimension array to construct a feature vector of the face image.

B. Embedding Minutiae Data

Step1: Combine all face feature vectors in one vector F of size $1 \times (k^2 * V)$.

Step2: Encrypt minutiae points using RSAC to get minutiae code(s)MS.

Step3: Select a binary password Psuch that has $(k^2 * V)$ bits.

Step4: The password P is scanned, and for each bit $P_j = 1$,binary representation of fraction of feature coefficient$F_j$is selected and normalized to unified length, say u.

Step5: All the selected $F_j$fractions are XORed, the result is XORed with one minutiae code MS, and that result is XORed with oneof the selected $F_j's$, say F_a. The result is stored instead of one of the unselectedF_j's, say F_t

$$F_t \leftarrow \left(\bigoplus_{P_j=1} F_j\right) \oplus MS \quad \oplus F_a \quad \dots\dots (3)$$

Step6: Repeat (Step5) for all minutiae codes MS.

Any reader familiar with steganographicfile system will know that the previous procedure is adjustment to the basic method of steganographicfile system that used to hide a file in a file system (for details see [17]). The adjustment is applied to increase the embedding capacity and to be suitable for our goal. The *stego-template* resulting from this procedurecan be transferred over an insecure channel or stored in system database.

Big coefficients (that its position ($modk^2$) equal to 0) should be avoided in (*Step5*). In other word,F_t should not be one of the big coefficients. The reason that it is usually have shortfraction.

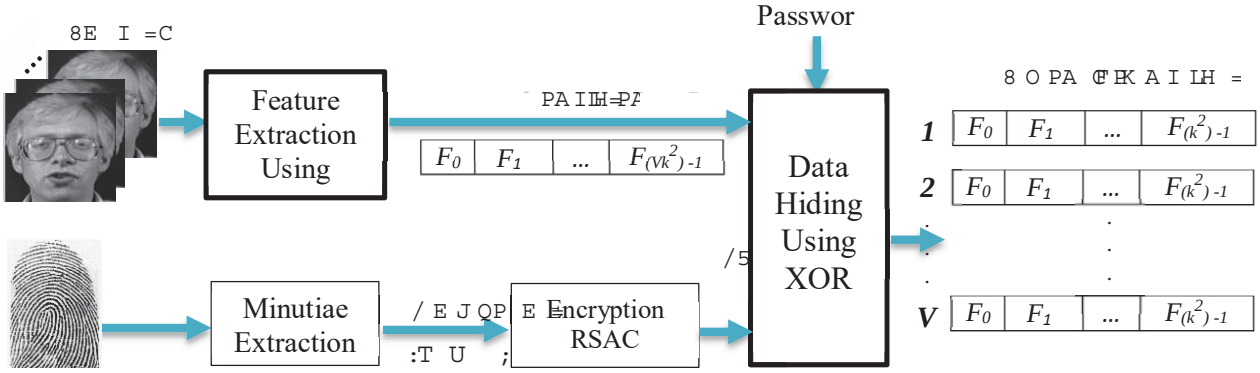

Figure 4. Embedding scheme

C. Extraction Procedure

Step1: combine all *V* stego-templates in one vector of size (k^2*V) coefficients

Step2: Scan the password *P* and for each bit $P_j = 1$, binary representation of fraction of feature coefficient F_j is selected.

Step3: All the selected F_j fractions, except F_a, are XORed, the result is XORed withF_t to obtain minutiae code(s)MS.

$$MS \leftarrow \left(\bigoplus_{P_j=1} F_j\right) \oplus F_t \ , \quad j \neq a \qquad \ldots\ldots\ldots (4)$$

Step4: Decrypt MS using RSAC decoder to get minutiae matrix.

An example to explain the embedding and extraction algorithms is given as:

$P = 01110, u = 4$,F_0 =1010, (=0011, (=1110, (=0111, and (=0100 , /5 L s rri=2 , and t=0;

(=((Ø (Ø) Ø /5Ø (

(=(0011 Ø1110 Ø0111) Ø1001 Ø1110=1101

/5 (Ø (Ø (= 1101 Ø0011 Ø0111=1001

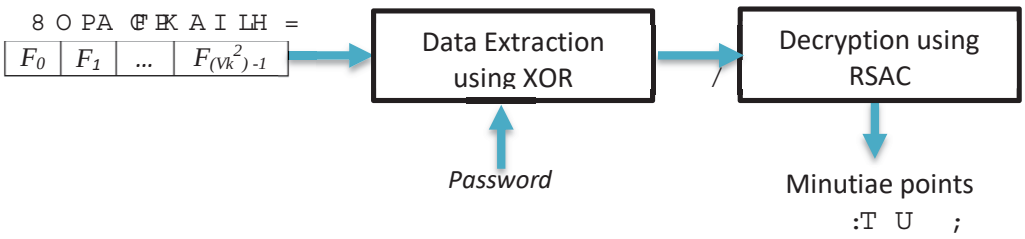

Figure (4): Extraction scheme

4. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed system, the stego-template should be compared directly with the incoming testing template without any degradation in face recognition performance. Three face databases were used to verify this condition. For each person, we select number of his images as training images and the others as testing images. For example,if 10 images per person was contained in a database, then 7 images were used as a training set, the remaining 3 images was used as a testing set. It is expected that the recognition accuracy of a face recognition system wouldbe improved as the number of faceimages per person increases.

In this system, the number of training image is determined not only by the requirements of face recognition system, but also by the security level required for hiding fingerprint minutiae. That is, the number of training images will determine the space of the password P. More images mean large space and high security level. The password space is also determined by the number of coefficients of feature vector.There is a slight decrease in recognition accuracy as we go to higher numbers of coefficients.Therefore, a tradeoff between increasing security level and maintaining the requirements of face recognition system will arise. This tradeoff can be solved through resizing training images according to their number. As the number of training images become large, the size should be small, and so on.

The experiments wererun two times, one before hiding minutiae data, and the other when the hiding process is done.The fingerprint minutiae were28 in average. Face recognition system simply works through comparing the templates stored in database,that generated using the procedureexplained in (3.2.A) in previous section, with the feature vector extracted from test images using the same procedure. The comparison was carried outusing Euclidian distance.Details about the databases used in these experimentsare listed in Table (5).

Table 5. Details about the databases used in the experiments

DB name	Image size	Resized to	Features block size	Training set		Testing images	
				Images / person	total	Genuin	Imposter
AT&T	112×92	100×100	5×5	7	210	90	100
Grimace	200×180	200×200	9×9	3	36	204	120
Faces94	200×180	200×200	9×9	3	180	1020	1059

Performance measuresare FAR, FRR, Accuracy. The results are explained in Table (6) and Table (7) before and after hiding data, respectively.

FAR L——————————————— :w;

L——————————————— :x;

L sr F l E t p :y;

DB Name	FAR%	FRR%	Accuracy%
AT@T	1	12.2	93.4
Grimace	0.83	0	99.58
Faces94	1.86	1.61	98.26

Tabel 6. FRR, FAR, and accuracy of face recognitison system before embeding minutiae data.

DB Name	FAR%	FRR%	Accuracy%
AT@T	1	12.2	93.4
Grimace	0.83	0	99.58
Faces94	1.86	1.61	98.26

Tabel 7. FRR, FAR, and accuracy of face recognition system after embeding minutiae data.

This result can be improved using the multimodal system described in this paper. That is, the rejected and the accepted individuals could be verified again using their fingerprint minutia, thereby reducing FRR and FAR.

5. CONCLUSION AND FUTURE WORK

The ability of hiding a secret data in the feature level, without any degradation in the performance of biometric system, has been tested and proved. The stego-template can be used directly in face recognition system without needing to retrievethe original cover-template. The number of selected coefficients was very small compared to images size, but was enough for ensuring high security level using the hiding scheme proposed by the author. The encryption algorithm RSAC is a new algorithm proposed to add additional defense line to sensitive biometric data against potential attacks.

For future work, iris featureswill be hidden in face features instead of fingerprint minutiae.

REFERENCES

[1] Arun Ross and Anil K. Jain, "Human Recognition Using Biometrics: An Overview", Annals of Telecommunications, Vol. 62, No. 1/2, pp. 11-55, Jan/Feb 2007, USA.

[2] Arun Ross and Asem Othman, "Visual Cryptography for Biometric Privacy", IEEE transactions on information forensics and security, Vol. 6, NO. 1, March 2011, USA.

[3] Yongwha Chung, Daesung Moon, Kiyoung Moon R. , and Sungbum Pan, "Hiding Biometric Data for Secure Transmission", Springer-Verlag Berlin Heidelberg, LNAI 3683, pp. 1049–1057, 2005, Korea.

[4] Fabien A. P. Petitcolas, Ross J. Anderson and Markus G. Kuhn, "Information Hiding-A Survey", IEEE, special issue on protection of multimedia content, 87(7):1062{1078, July 1999, UK.

[5] Sheng Li, Xin Chen, Zhigao Zheng, Zhenxin Qian, Guorui Feng, Xinpeng Zhang, "Data hiding in Fingerprint Minutiae Template for Privacy Protection", arXiv:1710.00516v1 [cs.CR] 2 Oct 2017.

[6] Shashikala Channalli, Ajay Jadhav, "Steganography an Art of Hiding Data", International Journal on Computer Science and Engineering, Vol.1(3), pp. 137-141, 2009, Pune.

[7] N. K. Ratha, J. H. Connell, and R. M. Bolle, “Secure data hiding in wavelet compressed fingerprint images,”in Proc. ACM Multimedia, pp. 127–130, Oct. 2000.

[8] Cong Jie Ng, Andrew Beng Jin Teoh, and Cheng Yaw Low, "Dct Based Region Log-Tiedrank Covariance Matrices for Face Recognition", IEEE 978-1-4799-9988-0/16, ICASSP 2016, South Korea.

[9] Bin Jiang, Guo-Sheng Yang, And Huan-Long Zhang, "Comparative Study Of Dimension Reduction And Recognition Algorithms Of DCT And 2DPCA", International Conference on Machine Learning and Cybernetics, Kunming, IEEE 978-1-4244-2096-4/08 12-15 July 2008, China.

[10] Ziad m. Hafed and martin d. Levine, "Face Recognition Using the Discrete Cosine Transform", International Journal of Computer Vision 43(3), 167–188, 2001, Kluwer Academic Publishers,2001, Canada.

[11] David Kruger, "Adaptive Homophonic Coding Techniques for Enhanced E-Commerce Security", University of Pretoria, 2002.

[12] Jiantao Zhou, Oscar C. Au, and Peter Hon-Wah Wong, "Adaptive Chosen-Ciphertext Attack on Secure Arithmetic Coding", IEEE Transactions On Signal Processing, Vol. 57, No. 5, May 2009, Hong Kong.

[13] Takeyuki Uehara, Reihaneh Safavi-Naini, "Attack On Lid /Farrell/Boyd Arithmetic Coding Encryption Scheme", Springer Science+Business Media Dordrecht 1999, Australia.

[14] Abbie Barbir, Ph.D., "A Methodology For Performing Secure Data Compression", IEEE, 0-8186-7873-9/97, 1997.

[15] G. Jakimoski and K. P. Subbalakshmi, "Cryptanalysis of Some Multimedia Encryption Schemes", Stevens Institute of Technology, Hoboken, NJ 07030, USA.

[16] Raj S. Katti, Sudarshan K. Srinivasan, and Aida Vosoughi, "On the Security of Randomized Arithmetic Codes Against Ciphertext-Only Attacks", IEEE Transactions On Information Forensics and Security, Vol. 6, No. 1, March 2011, USA.

[17] David Salomon, "Data Privacy and Security", Library of Congress Cataloging-in-Publication Data, Springer-Verlag New York, 2003.