ElGamel Encryption for Biometric Database Protection

M. Mani Roja
Research Scholar
Sant Gadge Baba Amravati University
Asso.Prof, TSEC, Mumbai

Sudhir Sawarkar, PhD.
Principal
Datta Meghe College of Engineering Mumbai

ABSTRACT
The critical issues in biometric systems are protecting the template of a user which is stored in a data base. An attack against the stored templates constitutes a major security and privacy threat in a biometric system. Proper use of cryptography greatly reduces the threats in biometric system as the attackers have to find the decryption key and template. This paper proposes an approach for biometric database protection using ElGamel encryption technique. The algorithm was successfully tested on a binary image, gray image and a colour image.

General Terms
Biometric Security, Image Encryption, Template Protection

Keywords
Primitive roots;PN sequence;ElGamel Encryption

1. INTRODUCTION
The security analysis of traditional password-based authentication schemes can be based on simple parameters such as

- Minimum length of passwords.
- The password change period.
- Inclusion of special characters.

The security of card/token based system can be analyzed based on the parameters

- Illegal utilization of the token if the token is lost and found by intruder.
- Ability to generate a token.
- Ability to forge a token.

2. ISSUES IN BIOMETRIC SECURITY
Biometric systems are more complicated than the conventional authentication schemes using password and token because of the following reasons. During every acquisition of the biometric data, there is a minor variation of the biometric is possible. Biometrics may need very good image enhancement schemes if the quality of the captured biometric sample is poor. Unlike passwords and tokens, compromised biometric templates cannot be revoked and reissued [1]. Due to intra user variability in the acquired biometric traits, ensuring the security of the template while maintaining the recognition performance is a challenging task [2].

2.1 Attacks on Biometric Templates
The biometric templates can undergo four possible vulnerabilities. They are

- An intruder who wants to gain unauthorized access can replace the existing template
- Unauthorized access can be gained using a physical spoof
- The template can be stolen and can be replayed later to gain access
- The templates can be used for verification across different databases to track a person without his permission.

Due to these reasons, the raw biometric images should not be stored in plaintext form and fool proof techniques are required to securely store the templates such that both the security of the application and the users' privacy are not compromised by adversary attacks.

3. RELATED WORK
Almost all the commercial biometric systems secure the stored templates by encrypting those using standard cryptographic techniques. Either a public key cryptosystem like RSA or a symmetric key cipher like AES is commonly used for template encryption. Since the above cryptosystems are generic, they can be directly applied to any biometric template and the encrypted templates are secure as long as the decryption key is secure. However, encryption is not a good solution for biometric template protection due to two main reasons. Firstly, encryption is not a smooth function and a small difference in the values of the feature sets extracted from the raw biometric data would lead to a very large difference in the resulting encrypted features. Due to this reason, one cannot store a biometric template in an encrypted form and then perform matching in the encrypted domain [3]. Hence, for every authentication attempt,

- The template is decrypted,
- Matching is performed between the query and decrypted template and
- The decrypted template is then removed from memory.

Thus, the template gets exposed during every authentication attempt. Secondly, the security of the encryption scheme depends on the decryption key. Hence, the decryption key needs to be securely stored in the system and if the key is compromised, the template is no longer secure. Because of these two reasons, standard encryption algorithms alone are not adequate for securing biometric templates and techniques that are designed to specifically account for the intra user variability in the biometric data are needed. Adler [4] used a "Hill Climbing Attack" to generate a face image from a face template. Feng et al. [5] have also proposed a similar technique by modeling a fingerprint image as a 2D Frequency Modulation (FM) signal whose phase consists of the continuous part and the spiral part, which corresponds to minutiae. Vetro et al. [6] have discussed the application of distributed source coding techniques to biometric security, by using a Slepian Wolf coding system to provide a secure means of storing biometric data that provides robust biometric authentication for genuine users and guards against attacks

from imposters. Wang et al. [7] have presented a theoretical framework for the analysis of privacy and security tradeoffs in secure biometric authentication systems.

Yeung and Pankanti [8] describe an invisible fragile watermarking technique to detect regions in a fingerprint image that have been tampered by an attacker. Jain and Uludag [9] suggest the use of steganography principles to hide biometric data (e.g., fingerprint minutiae) in host images (e.g., faces). Ferri et al. [10] propose an algorithm to embed dynamic signature features into face images present on ID cards. Ferri et al. report that any modification of the face image can be detected, thereby disallowing the use of fake ID cards. Uludag et al. [11] convert fingerprint templates (minutiae data) into point lists in 2D space, which implicitly hide a given secret (e.g., a 128-bit key). Mohapatra et al. [12] proposed a Biometric encryption method neither the key nor the original trait is stored, rather BE called biometric encrypted template is stored that contains the original template and as well as the key. Chander Kant et al. [1] presented a more secure system by use of steganography.

4. ELGAMEL ENCRYPTION

The security of public key crypto system is based on the difficulty of factoring. It is also possible to design a system whose security relies on the difficulty of computing discrete logarithms. This was done by ElGamel in 1985[13,14]. The implementation of this method needs a large prime number p and its primitive root α. In general, when p is a prime, primitive root mod p is a number 'a' whose powers yield every non zero class mod p. The importance of this notation is that if a is a primitive root of p, then its powers

$$a, a^2, a^3 \ldots a^{\phi(p)} \quad where\ \phi(p) = p-1$$

are distinct (mod p) and are all relative prime to p. Let us consider the prime number 7. The primitive roots of 7 are calculated in the following manner. From the table, we can find that the numbers 3 and 5 satisfies the conditions to be the primitive roots. Hence they are the primitive roots of prime number 7.

a	a2	a3	a4	a5	a6
2	4	1	2	4	1
3	2	6	4	5	1
4	2	1	4	2	1
5	4	6	3	1	5
6	1	6	1	6	1

Fig.1: Primitive roots of 7

4.1 Working Principle

Let p be a large prime number. The primitive root of p is found as α. The user has to select the private key and calculate the public key. Assume the private key to be a secret integer a, then the public key β is calculated as

$$\beta \equiv \alpha^a \pmod{p} \tag{1}$$

Choose a random integer k and then computes

$$r \equiv \alpha^k \pmod{p} \tag{2}$$

Now for a plain text M, the cipher text C is calculated as

$$t \equiv \beta^k M \pmod{p} \tag{3}$$

At the receiving side, the plain text is recalculated as

$$M \equiv tr^{-a} \pmod{p} \tag{4}$$

This works because

$$tr^{-a} = \beta^k M(\alpha^k)^{-a} \equiv (\alpha^a)^k M(\alpha^{-a})^{-k} \equiv M \tag{5}$$

4.2 Example for ElGamel Algorithm

Let us consider a prime number 11. This prime number has one of its primitive root as '3'. The next task is to decide the private key 'a' and then find the public key. Assume number 9 as the private key and then the associated public key is calculated as 4 using equation 1. Now let us consider a pixel from image say M=8. The encrypted pixel value is calculated as '6' using equation 3 for a random value 'k=10' with the help of the private key. Now for decryption, the value of r is calculated as '5' using the equation 2. Now the encrypted pixel value '6' is considered and decrypted using equation 4 to give the pixel value 8. Hence a pixel was successfully encrypted and decrypted using ElGamel algorithm

5. IMPLEMENTATION

The performance of the ElGamel encryption schemes was tested for three types of images namely binary image, gray scale image and a colour image.

5.1 Binary images

The procedure for encryption consists of the following steps

- select a large prime number p
- Find the primitive roots of p and select one primitive root.
- Select private key 'a' and calculate the associated public key using equation 1.
- Encrypt the pixels of biometric template using the formula equation 3 for a random number 'k'.
- Repeat the procedure for all images present in the database

The decryption of the template is done using the following steps

- Take the template to be decrypted
- Find the value of r using equation 2.
- Since, p and the associated primitive root are known, decrypt the pixel "t' using equation 4.

The results obtained for a binary image are shown in fig.2. It has been observed that even though the algorithm was doing encryption and decryption perfectly, the encrypted image was same as the original image. When we analyze, we found that , since the binary image is made up of only two values;1 and 0, even after encryption, the entire encrypted image was having pixels with two values , one corresponds to'0' and the other one corresponds to '1'. Hence the original image and the encrypted image produce the same histogram as shown in fig.2.

Fig.2: Binary Image Encryption & Decryption Process

Hence to overcome this difficulty, PN sequenced based encryption has been cascaded with ElGamel Encryption as shown in fig.3. PN sequences look like a random noise but they are not purely random in nature. When the PN sequence is multiplied with the signature template, the resulting image resembles noise. However, during decryption, this noise image can be used to exactly reconstruct the original image by multiplying it by the same pseudorandom sequence. Using this scheme, the initial seed state which is nothing but the key is only needed to generate exactly the same sequence of length.

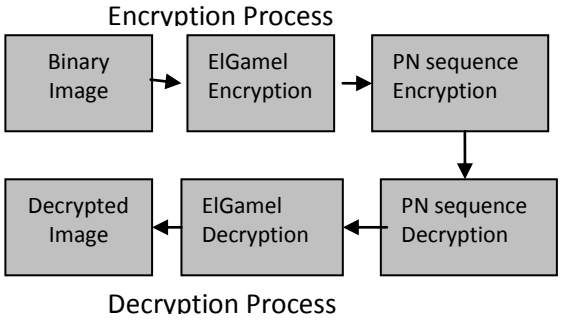

Fig.3: Proposed Setup for Binary Image Encryption Process

The results using PN sequence based encryption process is shown in fig.4. and the system was found to work satisfactorily since the histogram of the encrypted template is completely different from the original image histogram.

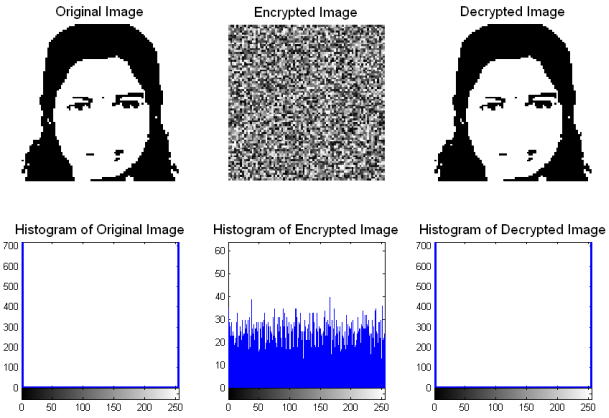

Fig.4: PNS based Binary Image Encryption & Decryption

5.2 Gray Images

For gray images, since the pixels are continuously varying, it was possible to get satisfactory results only using ElGamel encryption as shown in fig.5.

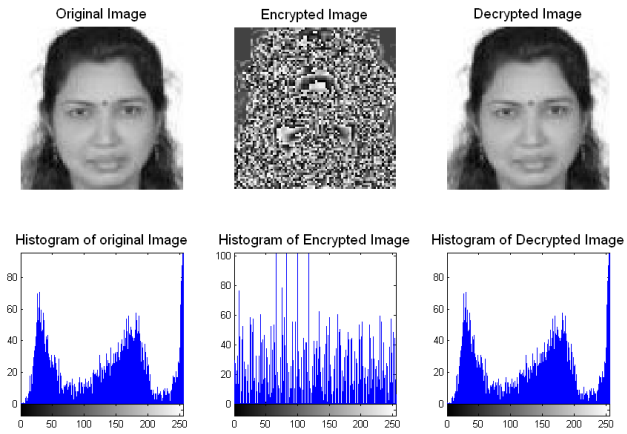

Fig.5: Gray Image Encryption & Decryption

5.3 Colour Images

A colour image is combination of red, green and blue components. When the RGB components are extracted from the colour image, every component is associated with a gray image. Hence, it was decided to extract the RGB components and then gray image encryption procedure was implemented on each component. The implementation steps are discussed below.

- Extract the RGB components from the colour image
- On each component apply ElGamel encryption
- Combine the encrypted RGB components to generate the Encrypted colour image

The results thus obtained are shown in fig.6

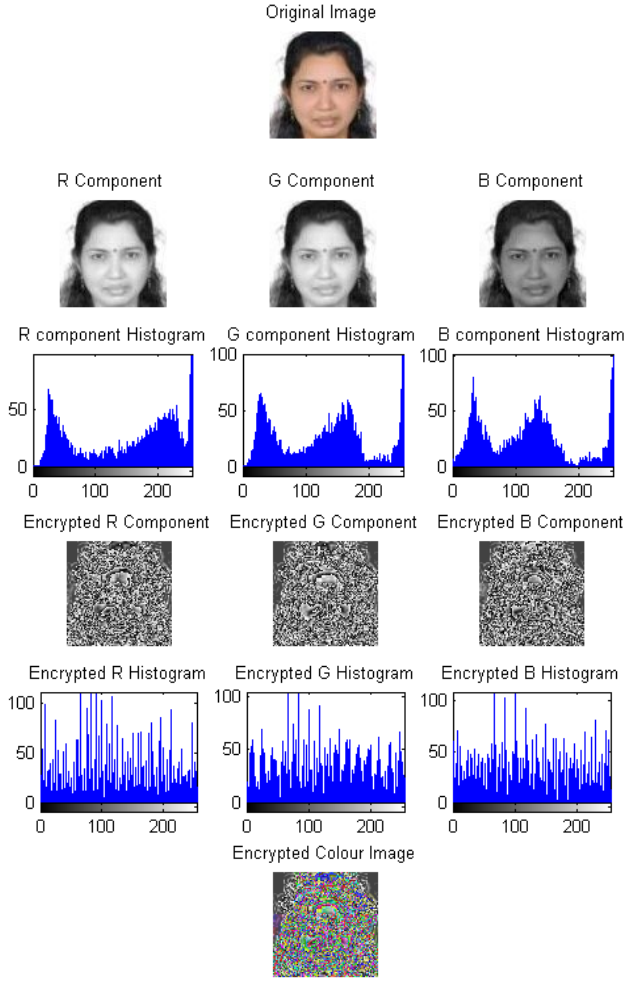

Fig.6: Colour Image Encryption

To decrypt the template the steps are as discussed below and process is shown in fig. 7.

- Take the template to be decrypted
- Extract the RGB components from the encrypted colour image
- On each component apply, ElGamel decryption process.
- Combine the decrypted RGB components to get the complete decrypted colour image

6. Results and Analysis

To evaluate the performance of the proposed method, we have the mean square error between the original face template and the decrypted face template has been calculated. The results are listed in table 1.

Table 1.

Nature of Image	Encryption Time in Sec	Decryption Time in Sec	MSE
Binary	8.45	0.31	0
Gray	8.17	0.30	0
Colour	9.30	0.4	0

Since binary image encryption was implemented as a combination of ElGamel scheme and PN sequence, it took more time than gray image for encryption and decryption process. Color image encryption process took more time since RGB components were considered. All three images gave mean square error of zero.

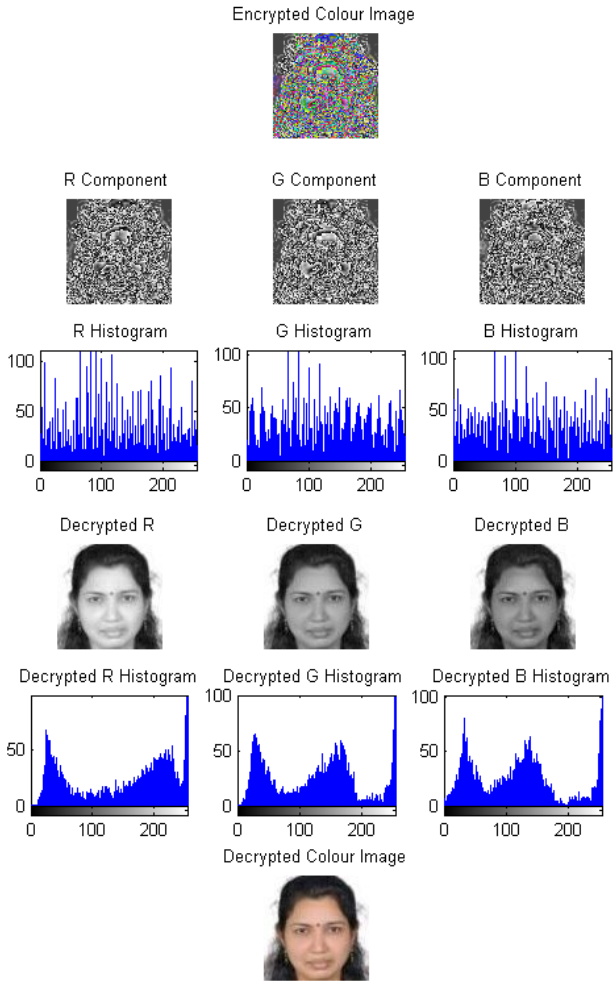

Fig.7: Colour Image Decryption

7. CONCLUSION

In this paper, an ElGamel encryption based approach is proposed for the protection of templates present in the biometric database. The algorithm was tested on a binary facial image, gray facial image and a colour image. While binary image needed a two stage procedure for encryption, gray and colour images needed a one step procedure. After decryption, all the three images were able to give a mean square error of zero. The entire system was implemented using MATLAB 7.1 software on an i3 processor.

8. REFERENCES

[1] Chander Kant, Ranjender Nath & Sheetal Chaudhary, "Biometrics Security using Steganography", International Journal of Security, vol. 2 , no.1,pp.1-5,2008.

[2] Manvjeet Kaur, Deepak Saraswat, Dr. Sanjeev Sofat, "Template and Database Security in Biometrics Systems: A Challenging Task", International Journal of Computer Applications Vol 4, no.5, pp. 1-5, July 2010.

[3] Anil K. Jain, Karthik Nandakumar, and Abhishek Nagar, "Biometric Template Security", EURASIP Journal on Advances in Signal Processing, Volume 2008, pp. 1-7, Article ID 579416.

[4] A. Adler, "Can Images be Regenerated from Biometric Templates?", in Biometrics Consortium Conference, (Arlington, VA), September 2003, pp.1-2, http:// www. sce. carleton. Ca / faculty /adler/publications/2003/adler-2003-biometrics-conf-regenerate-templates.pdf, dated June 2012.

[5] J. Feng, and A. K. Jain, "FM Model Based Fingerprint Reconstruction from Minutiae Template", Proc. International Conference on Biometrics (ICB), pp. 544-553, June 2009.

[6] A. Vetro, S. C. Draper, S. Rane and J. Yedidia, "Distributed Source Coding: Theory, Algorithms, and Applications", P. L. Dragotti and M. Gastpar (editors), Academic Press, pp. 293-324, 2009.

[7] Y. Wang, S. Rane, S. C. Draper and P. Ishwar, "A Theoretical Analysis of Authentication, Privacy and Resuability across Secure Biometric Systems", IEEE Trans. Inform. Forensics and Security, vol.7, no.6, pp.1825-1840,Dec 2012.

[8] M. Yeung and S. Pankanti, "Verification Watermarks on Fingerprint Recognition and Retrieval", in Proc. SPIE, Security and Watermarking of Multimedia Contents, vol. 3657, pp. 66–78, (San Jose, USA), January 1999.

[9] U. Uludag and A. K. Jain, "Attacks on Biometric Systems: A Case Study in Fingerprints", in Proc. SPIE, Security, Seganography and Watermarking of Multimedia Contents VI, vol. 5306, pp. 622–633, (San Jose, CA), January 2004.

[10] L. C. Ferri, A. Mayerhofer, M. Frank, C. Vielhauer, and R. Steinmetz, "Biometric Authentication for ID cards with Hologram Watermarks", in Proc. SPIE, Security and Watermarking of Multimedia Contents IV, vol. 4675, pp. 629–640, (Bellingham, WA), January 2002.

[11] A. K. Jain and U. Uludag, "Hiding Biometric Data", IEEE Trans. Pattern Anal. Mach. Intelligence, Vol. 25, No. 11, pp. 1493–1498, 2003.

[12] A.K.Mohapatra, Madhvi Sandhu, "Biometric Template Encryption", Published in International Journal of Advanced Engineering & Application, pp.282-284, Jan. 2010.

[13] William Stallings, "Cryptography and Network Security", Pearson Education, 5th edition, 2011

[14] Trappe, Washington, "Introduction to Cryptography with coding Theory", Pearson Education, 2nd edition, 2011.