Biometric Database Protection using Public Key Cryptography

M.ManiRoja[†] and Sudhir Sawarkar[††],

[†]Research Scholar, SGBA University, Asso.Prof. TSEC, Mumbai.
††Principal, DMCE, Airoli, Navi Mumbai, India.

Summary
With the increasing need for stringent security measures in recent times, biometric systems have assumed greater importance for information security systems. For biometric systems to offer reliable security, they themselves have to satisfy high security requirements to ensure invulnerability. This paper proposes two different approaches based on RSA and Elliptic curve cryptography for database protection of biometric authentication systems. Use of K-means algorithm is also proposed for the generation of encryption and decryption keys from the biometric templates. In this paper, implementation of public key algorithms has been realized for experimental purposes and the results thus obtained have been critically verified.

Key words:
Image encryption, Image cryptosystem, Biometric attacks, Database encryption

1. Introduction

Amongst the various types of biometric attacks [3], an attack which poses a significant threat and is potentially damaging in particular, is against the biometric templates stored in the system database [1,2]. Attacks on the template can lead to the following vulnerabilities:

- The stored reference template can be replaced by an impostor's template to gain unauthorized access.
- A physical spoof, essentially an imitation of the reference template, can be used to gain unauthorized access to the system.
- The stolen template can be replayed to the matcher to gain unauthorized access.

Traditional biometric authentication systems store biometric templates together with the data identifying an individual in a database for later comparison. In order to authenticate an individual the biometric data presented is looked up in the database. If a record is found with biometric data that is sufficiently close to the one presented, the person is identified and hence authenticated. However, the storage of biometric data leads to considerable risks for the authentication system and raises serious concerns regarding data protection. This way of storing biometric data is often criticized as a mass storage of privacy sensitive personal data that is potentially threatened by internal or external attacks on the database.

Therefore it would be of great value to protect biometric information by cryptographic means against not only external but also internal attacks.

A perfect image cryptosystem is required to be flexible in the security mechanism as well as be able to render high overall secure performance, and as such image security requires following characteristics [3]:

- The encryption system should be computationally secure. Cracking the system should require a reasonably long time thereby deterring the cracker to continue with the attack and thus barring the unauthorized user to read data.
- Encryption and decryption should be fast enough not to degrade system performance. The algorithm for encryption and decryption must be simple enough to be implemented by the user on a personal computer based platform.
- The security mechanism must be as widespread as possible.
- The security mechanism should be flexible.
- The scheme should not lead to a large expansion of encrypted image data. This is to avoid massive storage requirements.

The remainder of this paper is organized as follows. Current biometric templates protection schemes are introduced and reviewed briefly in section two. In section three, public key encryption schemes are introduced. Section four contains the proposed approaches for database protection. Experimental results and analysis are done in section five. Conclusions have been put forth in the final section.

II. Literature Review

Biometric system and the possible attack points are presented in fig.1. Ratha et al., [3] have identified eight attack points in this scheme. The UK biometric working group (UK-BWG) [4] lists several factors that can damage the integrity of the template as given below:

- Accidental template corruption due to a system malfunction such as a hardware failure.

Manuscript received May 5, 2013
Manuscript revised May 20, 2013

- Deliberate alteration of an enrolled template by an attacker.
- Substitution of a valid template with a bogus template for the purpose of deterring system functionality.

Adler [5] used a "Hill Climbing Attack" to generate a face image from a face template. Hill [7] describes a masquerade attack wherein the fingerprint structure is determined using the minutiae template alone. Ross et al., [8] propose another technique to elicit the fingerprint structure from the minutiae template. They use Gabor like filters suggested by Cappelli et al., [9] to generate fingerprints. Feng et al., [10] have also proposed a similar technique by modelling 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., [24] 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., [25] have presented a theoretical framework for the analysis of privacy and security tradeoffs in secure biometric authentication systems.

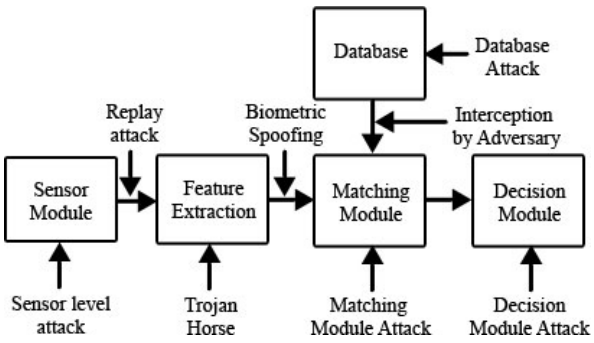

Fig.1 Possible Attack Points in Biometric System

In order to prevent the Hill climbing attack from successfully converging, Soutar [10] has suggested the use of coarsely quantized match scores by the matcher. However, Adler [11] demonstrated that it is still possible to estimate the unknown enrolled image although the number of iterations required to converge would be significantly higher. Yeung and Pankanti [12] describe an invisible fragile watermarking technique to detect regions in a fingerprint image that have been tampered by an attacker. Jain and Uludag [13] suggest the use of steganography principles to hide biometric data (e.g., fingerprint minutiae) in host images (e.g., faces). Ferri et al., [14] propose an algorithm to embed dynamic signature features into face images present on ID cards. Ferri et al., [14] report that any modification of the face image can be detected, thereby disallowing the use of fake ID cards. Ratha et al., [15] propose the use of distortion functions to generate biometric data that can be cancelled if necessary. Uludag et al., [17] 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., [18] 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., [19] presented a more secure system by use of steganography. Here the secret key (which is in the form of pixel intensities) will be merged in the picture itself while encoding, and at decoding end only the authentic user, who is aware of this specific, will be allowed to decode the encrypted image.

III. Biometric Database Protection

3.1 RSA Encryption

RSA is an algorithm for public key cryptography that is based on the presumed difficulty of factoring large integers, the factoring problem. The RSA algorithm was publicly described in 1978 by Rivest, Shamir and Adleman [21]. A user of RSA creates and then publishes the product of two large prime numbers, along with an auxiliary value, as their public key. The prime factors must be kept secret. The public key can be used to encrypt a message, however with currently published methods, only someone with knowledge of the prime factors can feasibly decode the message if the public key is large enough. The RSA encryption process is explained with a help of an example as given below.

Consider a scenario in which person A generates a public key and person B wants to use this public key to send a message to A. In this example, the message A sends to B is assumed to be entirely numeric. The steps are given by:

i. Person A selects two relatively large prime numbers.
ii. Person A multiplies p and q together to get $N = pq$ where N is the public key.
iii. Person A also chooses another number e which must be relatively prime to $(p-1)(q-1)$. 'e' is also part of the public key, so B also is told the value of e. At this stage, B has sufficient information to send an encoded pixel to A.
iv. B calculates the value of $C \equiv M^e \pmod N$ where M is the pixel to be encoded and C is the cipher pixel. The number C is the encoding that B sends to A.

v. Now A wants to decode C. To do so, A needs to find a number *d* such that

$$de \equiv 1(\mathrm{mod}(p-1)(q-1)) \quad (1)$$

vi. Now A calculates $M' \equiv C^d \pmod N$ where *M'* is the decrypted original pixel.

RSA cryptography works in conjunction with the following theorems:

1) Fermat's Little Theorem

If p is a prime number, and *a* is a positive integer not divisible by *p* then,

$$a^{p-1} \equiv 1(\mathrm{mod}\ p) \quad (2)$$

2) Euler's Theorem

Euler's theorem states that for every *a* and *n* that are relatively prime

$$a^{\varphi(n)} \equiv 1(\mathrm{mod}\ n) \quad (3)$$

where $\Phi(n)$ is the number of integers less than *m* that are relatively prime to *n*. The number *m* is not necessarily prime.

3) Chinese Remainder Theorem

Let *p* and *q* be two numbers that are relatively prime, then if $a \equiv b \pmod p$ and $a \equiv b \pmod q$, $a \equiv b \pmod{pq}$.

A. Working of RSA Algorithm

Let *p* and *q* be two large prime numbers, let $0 \le M < pq$ be a secret message, let *d* be an integer (usually small) that is relatively prime to $(p-1)(q-1)$, and let *e* be a number such that,

$$de \equiv 1(\mathrm{mod}(p-1)(q-1)) \quad (4)$$

To find e, Euler's theorem can be used to get,

$$d^{\varphi((p-1)(q-1))} \equiv 1(\mathrm{mod}(p-1)(q-1)) \quad (5)$$

Hence $d^{\varphi((p-1)(q-1))} (\mathrm{mod}(p-1)(q-1))$ is a suitable value for e. Now the encrypted message is found using,

$$C \equiv M^e (\mathrm{mod}\ pq) \quad (6)$$

The need is to show that the decrypted message is given by,

$$M \equiv C^d (\mathrm{mod}\ pq) \quad (7)$$

Using equation 4, for some integer k, we get,

$$de \equiv 1 + k(p-1)(q-1) \quad (8)$$

Thus $C^d \equiv M^{ed} \equiv M^{1+k(p-1)(q-1)} \equiv M.M^{(p-1)(q-1)k}$ (9)

If *M* is relatively prime to p, then by the extension of Fermat's Theorem,

$$M^{p-1} \equiv 1(\mathrm{mod}\ p) \quad (10)$$

Hence

$$M^{de} \equiv M.(M^{(p-1)})^{k(q-1)} \equiv M.(1)^{k(q-1)} \equiv M(\mathrm{mod}\ p) \quad (11)$$

By exactly the same reasoning,

$$M^{de} = M.(M^{(q-1)})^{k(p-1)} = M.(1)^{k(q-1)} = M(\mathrm{mod}\, q) \quad (12)$$

If Chinese remainder theorem is applied to equations 11 and 12, the result is given by

$$M^{de} \equiv M(\mathrm{mod}\ pq) \quad (13)$$

Thus the decrypted pixel is same as the original plain pixel value *M*.

3.2 Elliptic Curve Cryptography

In the mid-1980s, Miller and Koblitz introduced elliptic curves into cryptography, and Lenstra showed how to use elliptic curves to factor integers. One of the principal advantages of using elliptic curve cryptosystems is that they offer a comparable level of security with reference to classical cryptosystems that use much larger key sizes in comparison. For instance, Blake et al. in [28] have estimated that certain conventional systems with a 4096-bit key size can be effectively replaced by 313-bit elliptic curve systems without a considerable loss of accuracy. This results in appreciable savings in hardware implementations, thereby reducing the hardware requirement owing to a smaller key size requirement in terms of the number of bits.

Elliptic curve cryptography (ECC) [23] is a public key cryptography technique, based on the concept of elliptic curves. It can be effectively deployed in environments such as pagers, PDAs, cellular phones and smart cards. Such systems have proven to be most useful in environments which have long-term security requirements, mainly due to the intense security provided by the underlying elliptic curve discrete logarithm problem (ECDLP) [22]. A user participating in public key cryptography will essentially have a pair of keys, i.e., a public key and a private key. Only the particular user is aware of the private key whereas the public keys are distributed to all users taking part in the communication. The general cubic equation of elliptic curves can be written as:

$$y^2 + axy + by = x^3 + cx^2 + dx + e \quad (14)$$

For all practical applications, the above equation can be limited to the form:

$$y^2 = x^3 + ax + b \quad (15)$$

where a, b, c, d, e are appropriately either rational numbers, real numbers or integers mod p.

Consider an elliptic curve E_p (a, b) consisting of points (x, y) which satisfy the above equation together with element at infinity O. A group can be defined based on the set E_p (a, b) for specific values of a and b .The relations of commutativity, associativity, existence of an identity element and existence of inverse hold good. Now, if P and Q are points on E_p (a, b), elliptic curve discrete logarithmic problem (ECDLP) can be stated as "it is very difficult to find a value k such that Q=kP where P and Q are known, but it is relatively easy to find Q where k and P are known."

A. Working of ECC Algorithm

Let m the message pixel to be encrypted. The first task in the system is to encode the plain pixel message m to be sent as the x-y point P_m. This point P_m will be encrypted as a cipher pixel and subsequently decrypted [28]. An encryption system requires a point G and an elliptic group Ep (a, b) as parameters. The encryption key is generated as

$$P_A = n_A \times G \tag{16}$$

Here n_A is the primitive root of the prime number p.

To encrypt and send a message P_m, a random positive integer 'k' is selected and the cipher text C_m is produced as a pair of points as given in equation

$$C_m = \{ kG, P_m + kP_A \} \tag{17}$$

To decrypt the cipher pixel, the first point in the received pair is multiplied with n_A and subtracts the value obtained from the second point as given below.

$$P_m + kP_A - n_A(k.G) = P_m + k(n_A.G) - n_A(k.G) = P_m$$

It should however be noted that not every elliptic curve offers strong security properties and for certain curves, the ECDLP may be solved efficiently. Since a poor choice of the curve can compromise security and result into a vulnerable system, standards organizations such as NIST [26] and SECG [27] have published a set of recommended curves with well understood security properties. The use of these curves is also recommended as a means of facilitating interoperability between different implementations of a security protocol.

IV. Implementation

RSA involves a public key and a private key. The public key can be known to everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted using the private key. The keys for the RSA algorithm are generated the following way:

4.1 K-means Algorithm for Direct Key Generation

Direct key generation from biometrics is an appealing template protection approach which can also be very useful in cryptographic applications. A distinct advantage of using direct key generation is that the keys need not be remembered since the keys are derived from the templates. K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume *k* clusters) fixed a priori. The main idea is to define *k* centroids, one for each cluster. Since the end result is a function of the centroid location, the location of the centroids is of utmost importance. Hence, a variation in the location of the centroids may give rise to a different result. So, the better choice is to place them as much as possible far away from each other.

The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate *k* new centroids as barycenters of the clusters resulting from the previous step. After we have these *k* new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the *k* centroids change their location step by step until no more changes can be done.

The generation of biometric key is implemented using following steps:

i. Place *k* points into the space represented by the objects that are being clustered. These points represent initial group centroids.
ii. Assign each object to the group that has the closest centroid.
iii. When all objects have been assigned, recalculate the positions of the *k* centroids
iv. Arrange the centroids as $N_1N_2N_3\ldots N_k$. If there are 8 clusters, this approach gives an 8 digit number.
v. Now find out the next two prime numbers after this number which will be used as *p* and *q* for RSA encryption.

4.2 RSA Implementation

A. Encryption Procedure

i. Calculate *n*, *d* and *e* from *p* and *q*
ii. The combination of *e* and *n* will be used for encrypting the database images.
iii. The biometric template pixels are treated as message (M) and encrypted pixel by pixel using the equation
$C = M^e \pmod{pq}$.
iv. Now the encrypted image is stored in the database

B. Procedure for Decryption

i. The combination of *d* and *n* will be used for decryption.
ii. The encrypted image pixels are decrypted as
$M' = C^d \pmod{N}$.
Now the decrypted image is ready for comparison with the query image.

C. Need for two levels of Encryption using RSA

Upon implementing RSA encryption for both facial images and fingerprint biometric samples, the encrypted images were still observed to visually resemble the original images. The RSA encrypted images were found to be a shade of the original images as shown in fig. 2.

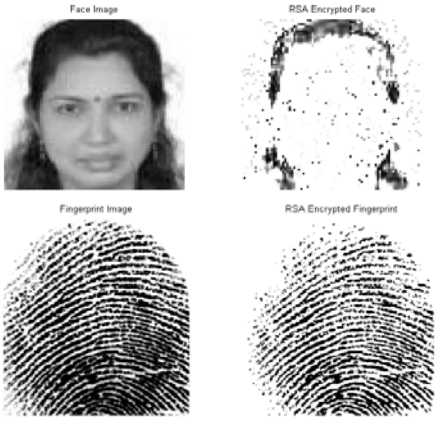

Fig. 2 Result of RSA Encryption

Due to the RSA encrypted samples retaining the pictorial characteristics of the original images, two levels of encryption have been considered to improve the security. After level 1 encryption which is obtained by RSA encryption, level 2 encryption is deployed, wherein a pseudo noise sequence [29] is used. The results of the above discussed procedure are shown in fig. 3

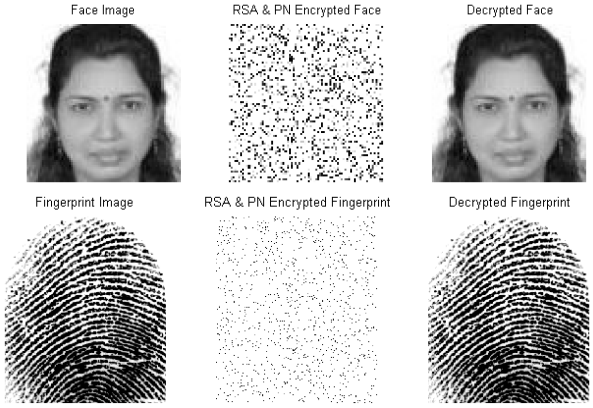

Fig. 3 Result of RSA & PN Encryption

The histograms of the images before and after encryption are shown in fig.4.

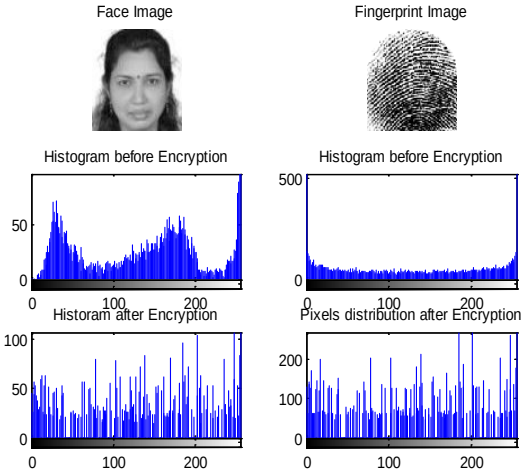

Fig. 4 Comparison of Histograms in RSA Encryption

4.3 ECC Implementation

A. Encryption Procedure

The following steps were adopted to implement encryption using ECC:

i. The implementation begins with the selection of a suitable password. We choose to work with an image having 8 bits per pixel representation, in which case the maximum possible value for the gray level of a pixel, say *m*, would be 255.

ii. Let *k* be any integer which would be used to find a square so that the failure rate to find a square and hence fail to associate the grey level to a point on an elliptic curve is $\frac{1}{2^k}$. Let us consider $k = 20$.

iii. Next, a prime number *p* is selected as:

$$mk + 1 < p \tag{18}$$

iv. Selection of *p* is done so that we obtain an equivalent point whose x co-ordinate is above or equal to mk + 1. Since the maximum gray level value is 255, *p* should be greater than 5001. Let us select $p = 6563$.

v. Choose any elliptic curve $E_p(1, 1)$ having *N* points on it, with the equation of the elliptic curve as:

$$y^2 = x^3 + ax + b$$

vi. For each gray level *m*, find *x* as *mk + 1* and solve for y, until you obtain y as in the following equation:

$$y^{\frac{(p-1)}{2}} = 1 (\text{mod } p) \tag{19}$$

vii. Compute the square root of modulo *p* of *y* as y_j.

viii. Consequently, the point P_m = (x_j, y_j) is the representation of the gray level of the image on the chosen elliptic curve. For a gray level value of 55 you get the point on elliptic curve as P_m = (1102,214)

ix. Compute the primitive root n_A of p. For 6563, the primitive root is found as 5.

x. Consider a point G on the chosen elliptic curve E_p (1, 1). As an example, let us consider G as (3, 5116).

xi. Calculate the public key used for encryption using equation 16 and we get P_A as (6121,5800) for the above example

xii. Choose a random positive integer k as 8 and compute:

$$P_1 = k.G \text{ as } (2966, 4895)$$

xiii. Find P_2 as: $P_2 = P_m + k(n_A.G)$ and we get

[1102, 214]+8[6121, 5800] = [5233, 2125].

xiv. Find the Cipher text as:

$$C = (P_1, P_2) = [(2966,4895),(5233,2125)]$$

xv. This procedure is repeated throughout the image and encryption is thus accomplished for all the pixels present in the image.

B. ECC Decryption Procedure

To recover the original information back from the data encrypted data using ECC, the following steps were adopted:

i. Upon multiplying P_1 by n_A, we get

$$5(2966,4895) = [1340,2566]$$

ii. Subtract this value from P_2 such that:

$$M = P_m + k(n_A.G) - n_A(k.G) = P_2 - n_A(k.G) \quad (20)$$

In our example, we get

[(5233, 2125) - (1340 2566)]= [1102, 214]

The point P_m is the representation of the gray pixel as a point on the elliptic curve.

iii. Consider the first value of P_m as x_j'. Calculate the gray level m as:

$$m = \frac{x_j'}{k} = \frac{1102}{20} \approx 55 \text{ which is the point } P_m$$

iv. The entire procedure is repeated for the entire encrypted image and the original image is thus reproduced back.

C. Results of ECC Encryption

Using the above mentioned procedure, encryption of face and finger print templates have been done and the results are shown in fig. 5 and the histograms are shown in fig 6.

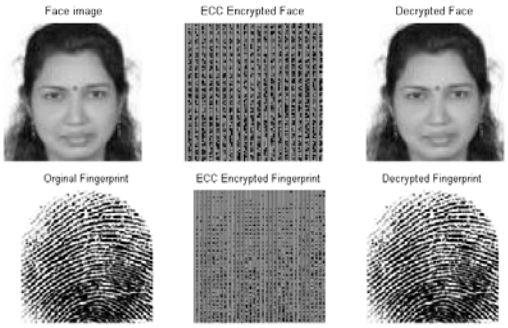

Fig.5 Results of ECC Encryption

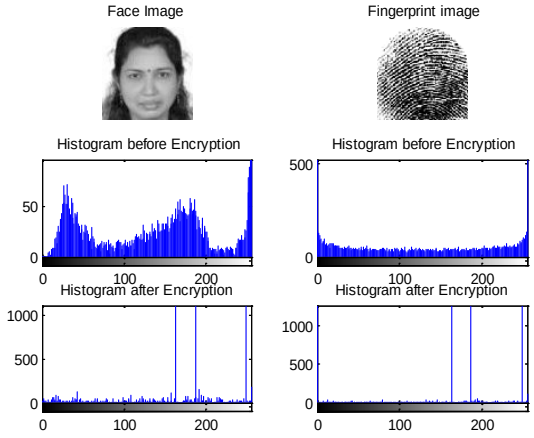

Fig. 6: Comparison of Histograms in ECC Encryption

V. Performance Analysis

Moment, contrast and entropy are the parameters considered for the performance analysis of the proposed encryption schemes. Analysis has been performed on a facial image to benchmark RSA and ECC encryption approaches.

5.1 Moment

In image processing, computer vision and related fields, an image moment is a certain particular weighted average (moment) of the image pixels' intensities, or a function of such moments, usually chosen to have some attractive property or interpretation. For a gray scale image with pixel intensities I(x, y), raw image moments M_{ij} are calculated by:

$$M_{ij} = \sum_x \sum_y x^i y^j I(x,y) \quad (21)$$

5.2 Contrast

Contrast is the difference in visual properties that makes an object distinguishable from other objects and the background. In visual perception of the real world, contrast

is determined by the difference in the color and brightness of the object and other objects within the same field of view. Because the human visual system is more sensitive to contrast than absolute luminance, we can perceive the world similarly regardless of the huge changes in illumination over the day or from place to place.

Contrast has multiple definitions in image processing. For our practical purposes, we have used Root Mean Square (RMS) contrast. RMS contrast does not depend on the spatial frequency content or the spatial distribution of contrast in the image. RMS contrast is defined as the standard deviation of the pixel intensities as

$$\sqrt{\frac{1}{MN}\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}(I_{ij}-\bar{I})^2} \quad (22)$$

where intensities I_{ij} are the ij[th] element of the two dimensional image of size M by N. $\bar{I}$ is the average intensity of all pixel values in the image. The image I is assumed to have its pixel intensities normalized in the range [0, 1].

5.3 Entropy

Entropy is a measure of disorder, or more precisely unpredictability. The entropy *H* of a discrete random variable *X* with possible values $\{x_1, ..., x_n\}$ is

$$H(x) = E(I(x)) \quad (23)$$

Here *E* is the expected value, and *I* is the information content of *X*. *I(X)* is itself a random variable. If *p* denotes the probability mass function of *X* then the entropy can be explicitly written as:

$$H(x) = \sum_{i=1}^{n} p(x_i)I(x_i) = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i) \quad (24)$$

The above values are calculated using both encryption schemes and the results are listed in table 1. From table 1, it is clear that in case of both RSA and ECC based approaches, the decrypted image is having same parameters as the original image.

Table I. Performance Evaluation

Parameter	*Original template*	*RSA Encryption*	*ECC Encryption*
Contrast	0.2353 x 10^8	0.2353x10^8	0.2353x 10^8
Entropy	0.0453x 10^8	0.0453x 10^8	0.0453x 10^8
Moment	1. 6085 x 10^8	1.6085x 10^8	1.6085 x 10^8

5.4 Noise Analysis for Remote Application

On considering an AWGN channel to judge the performance of the approaches for remote authentication applications, it is observed out that RSA encrypted image was beyond decryption at any amount of signal to noise ratio. This result is as shown in Fig. 7.

Fig.7 Result of RSA Encryption under noise effect

But under ECC encryption scheme, it is found out that the decryption works perfectly with 20 dB SNR. At various values of SNR values, the decrypted image is shown in Fig.8.

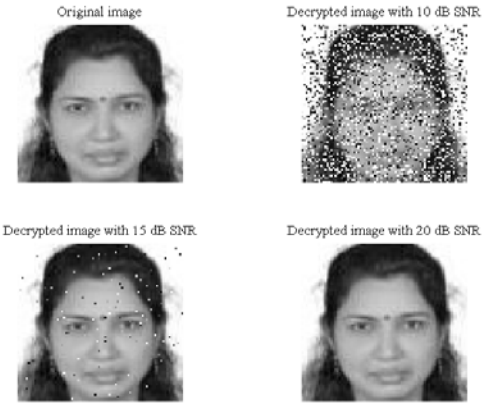

Fig.8 Result of ECC Encryption under noise effect

The Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR) are calculated and compared with the original template and the values are listed in table II.

Table II. Noise Analysis

SNR in dB	*MSE*	*PSNR in dB*
10	77.07	10
15	16.669	23
20	0.0415	75

5.5 Comparison of RSA with ECC encryption

The proposed algorithms were implemented using MATLAB 2010a on an Intel core i3 based platform. The average encryption time for a face image of size 256 × 256 is 10 seconds with the RSA scheme and 30 seconds using the ECC scheme. In terms of performance, the calculated values of contrast, entropy and moment are found to be the same for both the schemes. The RSA based approach required two levels of encryption while a single level of encryption was found to have yielded sufficient results for ECC encryption. In the event of using an AWGN channel for transmission, images encrypted using ECC scheme

were possible to decrypt but RSA encrypted images gave a relatively poor response.

VI. Conclusion

In this paper, the use of RSA and ECC based encryption schemes have been proposed for biometric template protection. The keys used for the encryption schemes were derived from the biometric template itself using the K-means algorithm. Even though RSA based encryption has a faster time response, ECC based encryption outperforms RSA based encryption under noise analysis and hence it is useful for remote authentication applications.

Acknowledgements

The authors would like to thank Aman Chadha, Graduate Student, University of Wisconsin-Madison, USA, for evaluating, testing and analysis of the proposed RSA and ECC based approaches.

References

[1] A. K. Jain, A. Ross and U. Uludag, "Biometric Template Security: Challenges and Solutions" Proceedings of European Signal Processing Conference (EUSIPCO), (Antalya, Turkey), September 2005

[2] A. K. Jain, K. Nandakumar, and A. Nagar, "Biometric Template Security", EURASIP Journal on Advances in Signal Processing Volume 2008, Article ID 579416, 17 pages

[3] N. Ratha, J. H. Connell, and R. M. Bolle, "An analysis of minutiae matching strength," in Proc. Audio and Video-based Biometric Person Authentication (AVBPA), pp. 223–228, (Halmstad, Sweden), June 2001.

[4] U. K. Biometric Working Group, "Biometric security concerns," Technical Report, CESG, September 2003, http://www. cesg.gov.uk/site/ast/biometrics/media/ Biometric Security Concerns.

[5] A. Adler, "Can images be regenerated from biometric templates?" in Biometrics Consortium Conference, (Arlington, VA), September 2003.

[6] 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.

[7] C. J. Hill, "Risk of masquerade arising from the storage of biometrics," B.S. Thesis, Australian National University, November 2001, http://chris.fornax.net/biometrics.html.

[8] A. Ross, J. Shah, and A. K. Jain, "Towards reconstructing fingerprints from minutiae points," in Proc. SPIE, Biometric Technology for Human Identification II, Vol. 5779, pp. 68–80, (Orlando, FL), March 2005.

[9] R. Cappelli, R. Erol, D. Maio, and D. Maltoni, "Synthetic fingerprint-image generation," in Proc. Int'l Conf. Pattern Recognition (ICPR), Vol. 3, pp. 475–478, (Barcelona, Spain), September 2000.

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

[11] A. Adler, "Images can be regenerated from quantized biometric match score data," in Proc. Canadian Conf. Electrical Computer Eng., pp. 469–472, (Niagara Falls, Canada), May 2004.

[12] 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.

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

[14] 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.

[15] N. Ratha, J. Connell, and R. bolle, "Enhancing security and privacy in biometrics-based authentication systems," IBM Systems Journal, Vol. 40, No. 3, pp. 614–634, 2001.

[16] U. Uludag, S. Pankanti, S. Prabhakar, and A. K. Jain, "Biometric cryptosystems: issues and challenges," Proceedings of the IEEE, Vol. 92, No. 6, pp. 948–960, 2004.

[17] U. Uludag, S. Pankanti, and A. K. Jain, "Fuzzy vault for fingerprints,", Proc. Audio and Video based Biometric Person Authentication (AVBPA), (RyeBrook, NY), July 2005.

[18] A.K.Mohapatra, Madhvi Sandhu, "Biometric Template Encryption", International Journal of Advanced Engineering & Application, Jan. 2010

[19] Chander Kant, Ranjender Nath & Sheetal Chaudhary, "Biometrics Security using Steganography", International Journal of Security, Volume (2) : Issue (1)

[20] Manvjeet Kaur, Sanjeev Sofat, Deepak Saraswat, "Template and Database Security in Biometrics Systems: A Challenging Task" International Journal of Computer Applications (0975 – 8887) Volume 4 – No.5, July 2010

[21] R. Rivest, A. Shamir; L. Adleman (1978). "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". Communications of the ACM 21 (2): 120–126.

[22] Padma Bh., D.Chandravathi. , P.Prapoorna Roja, "Encoding And Decoding of a Message in the Implementation of Elliptic Curve Cryptography using Koblitz's Method', International Journal on Computer Science and Engineering, Vol. 02, No. 05, 2010, pp. 1904-1907.

[23] Certicom, Standards for Efficient Cryptography, Sec 1: Elliptic Curve cryptography, version 1.0, September 2000, Available at http://www.secg.org/download/aid-385/sec1_final.pdf

[24] 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, 2009, pp. 293-324.

[25] Y. Wang, S. Rane, S. C. Draper and P. Ishwar, "A theoretical analysis of authentication, privacy and resuability across secure biometric systems," to appear in IEEE Trans. Inform. Forensics Security.

[26] Recommended Elliptic Curves for Federal Government Use, July 1999, Available at http://csrc.nist.gov /groups/ST/toolkit / documents /dss/ NISTReCur.pdf
[27] Certicom, Standards for Efficient Cryptography, Sec 2: Recommended Elliptic Curve Domain Parameters, Version 1.0, September 2000, Available at http://www.secg.org /download/aid-386/sec2_final.pdf
[28] I. Blake, G. Seroussi, N. Smart, "Elliptic Curves in Cryptography", Cambridge University Press, July 1999.
[29] R. Mutagi, "Pseudo noise sequences for engineers," Electronics & Communication Engineering Journal , Vol. 8, No. 2, pp. 79-87, Apr 1996.

M. Mani Roja was born in Tirunelveli (T.N.) in India on June 19, 1969. She has received B.E. in Electronics & Communication Engineering from GCE Tirunelveli, Madurai Kamraj University in 1990, and M.E. in Electronics from Mumbai University in 2002. Her employment experience includes 22 years as an educationist at Thadomal Shahani Engineering College (TSEC), Mumbai University. She holds the post of an Associate Professor in TSEC. Her special fields of interest include Image Processing and Data Encryption. Currently, she is pursuing her PhD from Sant Gadge Baba Amravati University. She has over 25 papers in National / International Conferences and Journals to her credit. Ms. M.Mani Roja is a member of IETE, ISTE, IACSIT and ACM.

Dr. Sudhir Sawarkar was born in Amravati, Maharashtra in India on 18th October, 1966. He received his BE (Electronics) and ME (Electronics) from Sant Gadge Baba Amravati University, India in 1988 and 1995 respectively. He received his PhD degree in 2007 from Dr. Babasaheb Ambedkar Technological University, Lonere, Maharashtra India. He is currently working as a Principal of Datta Meghe college of Engineering, Navi Mumbai. His special fields of interest include Image Processing and Neural networks. His employment experience includes 25 years in teaching. He has published more than 75 research papers in national / international journals /conferences. He has guided many ME dissertations. He is a recognized PhD supervisor in Amravati University and many other universities.