A Novel Approach for Non-Invertible Cryptographic Key Generation from Cancellable Fingerprint Template

K.Siva Rama Krishna, G.Raghavendra, M.Shiva, P.Srinivasa Rao, K.Surya Narayana, K.Veerraju Chowdary

Abstract: - Generation of cancellable cryptographic key from cancellable fingerprint template is an encryption technique used to authenticate confidential systems. This method gives us the advantage that we can regenerate the key from same finger print even if the key is compromised from a different template generated by a different random algorithm.

This method can be employed in banking and financial applications, defense sector and other confidential matters.We implemented Pre-processing block as given in literature and developed proficient algorithms for ridge filling after edge detection block. This slightly improves the efficiency of the system over the existing algorithms. We implemented efficient algorithm for false minutiae and multiple minutiae removal. By removing the false features, we can reduce the rate of false identification and from increased efficiency we can reduce the rate of false rejection. Even the efficiency is better compared to the older algorithms, the result still depends on the quality of the acquisition devices. If we use a poor acquisition device, that may lead to either false identification of false rejection.

INTRODUCTION:

Personal identification is to associate a particular individual with an identity. It plays a critical role in our society, in which questions related to identity of an individual such as “Is this the person who he or she claims to be?”, “Has this applicant been here before?”, “Should this individual be given access to our system?” “Does this employee have authorization to perform this transaction?” etc. are asked millions of times every day by hundreds of thousands of organizations in financial services, health care, electronic commerce, telecommunication, government, etc. With the rapid evolution of information technology, people are becoming even more and more electronically connected. As a result, the ability to achieve highly accurate automatic personal identification is becoming more critical.

Traditionally, passwords(knowledge-based security) and ID cards (token-based security) have been used to restrict access to systems. The major advantages of this traditional personal identification are that they are very simple and they can be easily integrated into different systems with a low cost.

However these approaches are not based on any inherent attributes of an individual to make a personal identification thus having number of disadvantages like tokens may be lost, stolen, forgotten, or misplaced; PIN may be forgotten or guessed by impostors. Security can be easily breached in these systems when a password is divulged to an unauthorized user or a card is stolen by an impostor; further, simple passwords are easy to guess (by an impostor) and difficult passwords may be hard to recall (by a legitimate user).Therefore they are unable to satisfy the security requirements of our electronically interconnected information society. The emergence of biometrics has addressed the problems that plague traditional verification.

A growing number of biometric technologies have been proposed over the past several years, but only in the past 5 years have the leading ones become more widely deployed. Some technologies are better suited to specific applications than others, and some are more acceptable to users. We describe seven leading biometric technologies:

- Facial Recognition
- Fingerprint Recognition
- Hand Geometry
- Iris Recognition
- Signature Recognition
- Speaker Recognition

Fingerprint is one of the most well-know and publicized biometrics for personal identification, because it is unique. Different people have different fingerprints. Fingerprint has been used as an identification approach for a long time. Fingerprint recognition means provide an automated method of verifying a match between two human fingerprints. Today, Fingerprint recognition is widely used in human life. For example, in security identify a human for accessing a building or accessing a system. Some personal laptop provides a fingerprint recognition function to allow a user login.

Dactyloscopy is the study of fingerprint identification. Police investigators are experts in collecting “dactylograms”, otherwise known as fingerprints.

There are three types of fingerprint patterns as per the classification

i) Arches
ii) Loops

iii) Whorls

Fingerprint Factoid Approximately 60% of people have loops, 35% have whorls, and 5% have arches.

Fingerprints are widely used and well accepted biometric. A lot of research and processing has been done on fingerprints. In the field of crime investigation fingerprint plays a vital role. Extracting fingerprints also an easy way. Latent prints are processed and taken to investigate crimes.

In recent times, researchers have focused on incorporating biometrics with cryptography as a possible way to enhance overall security by purging the necessity of key storage via passwords. Biometric cryptosystems, or crypto-biometric systems, unite cryptographic security with biometric authentication. In the cryptographic technique the original data is encoded by using any key so that it is not in an understandable format for the attacker [1]. The original data can be obtained by decoding the encoded data using the same key. Thus the privacy is well protected in this cryptographic approach. Several cryptographic techniques like DES, AES and public key architectures like RSA are widely used for the authentication purpose.

In this report, corrupted fingerprint image is firstly subjected to pre-processing and then it is made such that prominent futures in the fingerprint are clearly intelligible. Then pre-processed fingerprint template is given to a thinning algorithm for further processing. The features in the fingerprint (minutiae) are extracted from thinned image.

Finally, A subset of minutiae points are given to cancellable fingerprint template generation algorithm. And further template is given to crypto-system to give non-recoverable key.

The rest of the report is organized as follows. Our proposed algorithm and block diagram is presented in Section II. The Pre-processing of the fingerprint image is presented in Section III. The Extraction of minutiae points from enhanced fingerprint image is presented in Section IV. The transformation of the minutiae points and the generation of the cancellable fingerprint template from the transformed minutiae points are discussed in Section V. Generation of cryptographic key using Data Encryption Standard (DES) discussed in Section VI. The Conclusions and future work are summed up in Section VII.

CHAPTER 2

PROPOSED ALGORITHM

Cancellable fingerprint generation from user given fingerprint includes four main steps:

Pre-processing

Feature Extraction

Cancellable Fingerprint Generation

Cryptographic Key Generation

We implemented Pre-processing block as given literature and developed proficient algorithms for ridge filling after edge detection block. We implemented efficient algorithm for false minutiae and multiple minutiae removal.

CHAPTER 3

PREPROCESSING [2]

A critical step in cancellable fingerprint template generation is to automatically and reliably extract minutiae from the input fingerprint images for further processing. The fingerprint image is not suitable for minutiae extraction after we get the image from fingerprint device. However, the performance of a minutiae extraction algorithm relies heavily on the quality of the input fingerprint images. In order to ensure that the performance of minutiae extraction module will be robust with respect to the quality of input fingerprint images, it is essential to incorporate a fingerprint enhancement algorithm before the minutiae extraction module.

We present a fast & efficient fingerprint enhancement algorithm, which can adaptively improve the clarity of ridge and valley structures of input fingerprint images based on the estimated local ridge orientation and frequency of normalized image. We have evaluated the performance of the image enhancement algorithm using subjective analysis and the goodness index of the extracted minutiae and the accuracy of an online fingerprint verification system.

The Pre-processing stage includes normalization, Orientation estimation, segmentation, ridge extraction and binarization.

3.1 Normalization [3]

Normalization is a pixel-wise operation. It does not change the clarity of the ridge and valley structures. The main purpose of normalization is to reduce the variations in gray-level values along ridges and valleys, which facilitates the subsequent processing steps. The normalized image is defined as the follows mathematics formula (Equation 1):

$$G(i,j)=\begin{cases} M_0+\sqrt{\dfrac{VAR_0(I(i,j)-M)^2}{VAR}} \\ M_0-\sqrt{\dfrac{VAR_0(I(i,j)-M)^2}{VAR}} \end{cases}$$

Equation 1: Normalization mathematical form

In this formula, I(i,j) means the gray level of point (i, j). M_0 and VAR_0 are the desired mean and variance values. Below figure shows an example of image normalization.

3.2 Orientation Estimation [3]

An orientation field represents the directionality of ridges in the fingerprint image. It is a very important role in fingerprint image analysis. This step is a basic step for minutiae extraction. It also prepare for image segmentation. *“Fingerprint image is typically divided into a number of non-overlapping blocks (e.g. 32x32 pixels) and an orientation representative of the ridges in the block is assigned to the block based on an analysis of grayscale gradients in the block. The block orientation could be determined from the pixel gradient orientations based on, say, averaging, voting, or optimization”.*

The following steps show the processing of orientation estimation:

- Divide the input fingerprint image into blocks of size WxW.
- Compute the gradients Gx and Gy at each pixel in each block.
- Estimate the local orientation at each pixel (i, j) using the equations below

$$V_x(i,j) = \sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} 2G_x(u,v)G_y(u,v),$$

$$V_y(i,j) = \sum_{u=i-w/2}^{i+w/2} \sum_{v=j-w/2}^{j+w/2} (G_x^2(u,v) - G_y^2(u,v)),$$

$$\theta(i,j) = \frac{1}{2}\tan^{-1}\frac{V_x(i,j)}{V_y(i,j)}$$

Equation 2: Formula for local orientation estimation at each pixel

Where W is the size of the local window; Gx and Gy are the gradient magnitudes in x and y directions, respectively.

- Compute the consistency level of the orientation field in the local neighbourhood of a block(i, j) with the following formula:

$$C(i,j) = \frac{1}{N}\sqrt{\sum_{(i',j')\in D} |\theta(i',j') - \theta(i,j)|^2},$$

- If the consistency level is above a certain threshold T_h, then the local orientations around this region are re-estimated at a lower resolution level until C(i, j) is below a certain level.

3.2.1 Reliability of Orientation Field [2]

It is a measure of reliability of the orientation measure. This is a value between 0 and 1. The value above 0.5 can be considered ‘reliable’. Reliability is used to find out non-recoverable regions even after enhancement.

3.2.2 Ridge Frequency

Ridge Frequency algorithm is to estimate the fingerprint ridge frequency across a fingerprint image. This is done by considering blocks of the image and determining a ridge count within each block.

3.3 Segmentation

Image segmentation is a basic way for fingerprint image enhancement. We cannot extract features from a fingerprint image without image enhancement, because without image segmentation, some important features will not present clearly, some unimportant features will present, some features maybe present twice. All these will lead to a false feature extraction. Segment is a way for keeping the useful image information and removes the un-useful image information. Image segmentation is typically used to locate objects and boundaries in images.

There are mainly two methods in Segmentation

3.3.1 Histogram-Based method

Histogram-based methods are very efficient when compared to other image segmentation methods because they typically require only one pass through the pixels. In this technique, a histogram is computed from all of the pixels in the image, and the peaks and valleys in the histogram are used to locate the clusters in the image.

In this method, image has been divided into several blocks. Using a grey level wavelet histogram to presents each block grey level, so that, we can determine how many blocks are useful (How many blocks are in the accepting grey level). With this method, we can keep the useful information part in the image, but it has a disadvantage. Its disadvantage is that it may be difficult to identify significant peaks and valleys in the image. [7]

3.3.2 Edge Detection method

Edge Detection algorithms are useful in fingerprint segmentation in identifying points in a digital image at which the image brightness changes sharply or more formally has discontinuities. [8]

Through edge detection, we can reduce amount of data and throw away information which is not used for fingerprint analysis. The idea underlying most edge-detection techniques is on the computation of a local derivative operator such as “Roberts”, “Prewitt”, “Canny” or “Sobel” operators.

3.3.2.1 Canny Edge Detection [9]

An edge in an image may point in a variety of directions, so the canny algorithm uses four filters to detect horizontal, vertical and diagonal edges in the blurred image. The edge detection operator (Roberts, Prewitt, and Sobel for example) returns a value for the first derivative in the horizontal direction (Gy) and the vertical direction (Gx). From this the edge gradient and direction can be determined as follows

$$\mathbf{G} = \sqrt{\mathbf{G}_x{}^2 + \mathbf{G}_y{}^2}$$

$$\mathbf{\Theta} = \arctan\left(\frac{\mathbf{G}_y}{\mathbf{G}_x}\right)$$

Equation 3: Formula of Canny Detector; determine the edge gradient and direction

The edge direction angle is rounded to one of four angles representing vertical, horizontal and the two diagonals.

3.3.2.2 Gabor Filter

Gabor filter is a linear filter used for edge detection. It has been found to be particularly appropriate for texture representation and discrimination.

Its impulse response is defined by a harmonic function multiplied by a Gaussian function. Because of the multiplication-convolution property (Convolution theorem), the Fourier transform of a Gabor filter's impulse response is the convolution of the Fourier transform of the harmonic function and the Fourier transform of the Gaussian function. The filter has a real and an imaginary component representing orthogonal directions the two components may be formed into a complex number or used individually.

Real

$$g(x,y;\lambda,\theta,\psi,\sigma,\gamma) = \exp\left(-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right)\cos\left(2\pi\frac{x'}{\lambda}+\psi\right)$$

Equation 4: Gabor filter basic formula

Imaginary:

$$g(x,y;\lambda,\theta,\psi,\sigma,\gamma) = \exp\left(-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right)\sin\left(2\pi\frac{x'}{\lambda}+\psi\right)$$

Equation 5: Gabor filter imaginary part

Where,

$$x' = x\cos\theta + y\sin\theta$$

$$y' = -x\sin\theta + y\cos\theta$$

In this equation, λ represents the wavelength of the sinusoidal factor, θ represents the orientation of the normal to the parallel stripes of a Gabor function, ψ is the phase offset, σ is the sigma of the Gaussian envelope and γ is the spatial aspect ratio, and specifies the ellipticity of the support of the Gabor function.

These two algorithms can be used in edge detection. Canny edge detector is traditional way and Gabor filter seems more suitable for do the edge detection in the image which includes character only, but some research papers say edge detection using Gabor filter is accurate in fingerprint edge detection, so here we present both canny edge detection and then to use Gabor filter in our program.

3.4 Ridge Filling and Ridge Filter for smoothing:

After edge detection, we can get an image that: reduce amount of un-useful data. But still it is not sufficient for further processing because small portions of ridges may be broken after edge detection leading to false minutiae. Ridge filling algorithm, by using pixel adjacency can fill up the gaps in ridges to reduce false minutiae points.

After that, Ridge filter algorithm is used to smooth out the segmented image by using Orientation image and ridge frequency. Reliability of Orientation is used to remove non-recoverable regions after smoothing. Finally, we will get smoothed image for thinning process.

3.5 Ridge extraction [3]

The important property of the ridges in a fingerprint image is that grey level values on ridges. There is an algorithm was provided base on grey level threshold. This algorithm is image binarization [4]. The approaches to ridge extraction use either simple or adaptive threshold. With this way, we can separate the foreground (Fingerprint part) and the background. The theory of this algorithm is that: calculate a gray level threshold, compare this threshold to each pixel of the fingerprint image. Below equation is the formula of this algorithm.

$$iVal = \begin{cases} 1 & if\ Gmean[I] \geq Gmean[iVar] \\ 0 & Otherwise \end{cases}$$

Equation 6: Thresholding formula

Where iVar is point i, Gmean [I] is this point's gray level, Gmean[iVar] is the gray level threshold. With this way, the foreground's gray level will be set in 1, and background's gray level will be set in 0. The ridge can be extracted. After this, fingerprint image become a binary image. Below figure is an example of result image.

3.6 Thinning

After Ridge extraction, the image still cannot be used for minutiae extraction, because the ridge is too wide, it is not suitable for extract feature points. We have to make the ridge thin. Thinning is to representing the structural shape of a plane region is to reduce it to a graph. This reduction may be accomplished by obtaining the skeleton of the region via thinning.

CHAPTER 4

FEATURE EXTRACTION

After pre-process, a fingerprint image is prepared for minutiae extraction. A fingerprint is characterized by a pattern of interleaved ridges (dark lines) and valleys (bright lines). Generally, ridges and valleys run in parallel and sometimes they terminate or they bifurcate. At a global level, the fingerprint may present regions with patterns of high curvature; these regions are also called singularity. At the local level, other important feature called minutia can be found in the fingerprint patterns. Minutia mean small details and this refers to the behaviour of the ridges discontinuities such as termination, bifurcation and trifurcation or other features such as pores (small holes inside the ridges), lake (two closed bifurcations), dot (short ridges), etc. I prefer to match the minutiae for comparing, so that, I will only describe the minutiae extraction in local level. There are two algorithms for minutiae extraction.

4.1 Using a 3X3 template mask on the binary thinned fingerprint image

Here, we use a 3X3 template mask to extract the minutiae. The main process of this algorithm is that:

Steps:

```
If the translation count is 4
Point(x, y) is a bifurcation minutia;
If the translation count is 2
Point(x, y) is an ending minutia;
Else,
Point(x, y) is not minutia;
```

4.2 Using 3X3 mask base-on neural network [5]

This algorithm is based on neural network algorithm. The minutiae are detected by using 3X3 masks. All this masks used for identifying the ridge ending and bifurcations point. Before minutiae extraction, we have to train the network will all the masks. Once the neural network is trained, next step is to input the prototype fingerprint image to extract the minutiae, using these masks to scan the fingerprint image.

4.3 False Minutiae Removal

After minutiae extraction, some of the minutiae points exists at the borders of the fingerprint image and some minutiae may be present because of loops existed in thinned image. These minutiae points may not be present in original fingerprint image. So, we have to remove those from final thinned image before feature extraction.

A novel approach of making mask image from normalized image in such a way that non-ridge regions in normalized image makes mask image which when applied to thinned image will remove false minutiae points at the borders of fingerprint image.

After minutiae extraction process, we may also get multiple bifurcation and multiple ridge ending points, we can remove those points by considering bifurcation point's image and applying multiple bifurcation removal algorithm we developed. Finally, we will get minutiae extracted image almost removing all false minutiae points for generation of cancellable fingerprint generation.

4.3. Algorithm for Removing Minutiae at borders

Steps followed:

```
❖ Divide normalized image into 30 blocks giving block size 10x10
❖ Binarize the block obtained using threshold 128 resulting binary block
❖ Find the sum of pixel values in binary block
❖ If sum of pixel values is greater than threshold (ex: 95 for 10x10 block)
  Make block as non-ridge region
❖ Else
  Make block as ridge region
```

4.3.2 Algorithm for Multiple Minutiae removal

Steps followed:

```
❖ Scan the bifurcation points from left to right and top to bottom
❖ Intensity value of pixel is I(x, y)
❖ If pixel value is 0
    Apply bifurcation mask at the pixel
    If number of 0's in mask is 2
    Point(x, y) is an ending minutiae
    If number of 0's in mask is 4
    Point(x, y) is bifurcation minutiae
   And remove m-connected bifurcation at point(x, y)
❖ Else
    Move to next pixel
```

CHAPTER 5

CANCELLABLE FINGERING TEMPLATE GENERATION [6] [10]

This module transforms the extracted minutiae points into transformed points and the generation of cancellable fingerprints. The extracted minutiae points are represented as

$$M_p = \{P_1, P_2, P_3, \ldots\ldots\ldots\ldots, P_n\}$$

And their equivalent x, y coordinates are specified as

$$M_{p1}(x_1, y_1), M_{p2}(x_2, y_2), M_{p3}(x_3, y_3), \ldots, M_{pn}(x_n, y_n)$$

These x, y co-ordinates are symbolized as a vector and transformed completely into another set of transformed points with the aid of the deterministic algorithm discussed. To begin with, the x, y co-ordinates of the minutiae points are stored in a vector V_C. For each element in the vector V_C the corresponding next prime number is obtained and placed in another vector V_P. Then, a discrete exponential function is applied on individual elements of V_C with their corresponding values in V_P. If the discrete exponential value E_D computed is prime, then the value is appended to a vector PD_E, else the corresponding next prime number is obtained and appended to PD_E.

$$C_v = [\, x_1 \;\; y_1 \;\; x_2 \;\; y_2 \ldots x_n \;\; y_n]$$

$$P_v = [\, x_1' \quad y_1' \quad x_2' \quad y_2' \ldots . x_n' \qquad y_n']$$

$$D_E = 2^{C_{v(i)}} mod\, P_{v(i)}; 1 \le i \le n$$

$$PD_E \ll \begin{cases} D_{E(i)}; & if\ (D_{E(i)} = prime) \\ nextprime\, ; otherwise \end{cases}$$

$$PD_E = [P_{x_1}\, P_{y_1}\, P_{x_2}\, P_{y_2} \ldots\, P_{x_n}\, P_{y_n}\,]$$

The following steps are involved in the formation of the transformed points from the vector PD_E:

- Random pair selection [9]: The indexes for random selection of pairs from PD_E are computed by the below mathematical operation. The random pairs selected are removed from PD_E and the process is repeated until PD_E is empty.

$$rand(\,)\ mod(|PD_E| - k)\ ;$$

$$where\ k = 0,2,4,6, \ldots , |PD_E|$$

- Prime factoring: The pair of values selected is prime numbers and represented as (R_1, R_2). The values in each pair are multiplied to obtain the transformed points. The pairs taken out from PD_E are represented as

$$R_p = \{(R_{11}\ , R_{12}), (R_{21}\ , R_{22}), (R_{31}\ , R_{32}), \ldots (R_{n1}\ , R_{n2})\}$$

- The transformed points are denoted as

$$T_p = (P_1\ , P_2, P_3, \ldots , P_n)$$

$$Where\ P_i = (R_{i1} * R_{i2}), 1 \le i \le n$$

- As the two values R_{i1} and R_{i2} are prime numbers, the multiplication results in a value that is almost infeasible to factorize. The utilization of prime number factoring and discrete exponential guarantees that, obtaining minutiae points' co-ordinates from the transformed points is extremely complex. Subsequently, the distance between each point with respect to the other points is computed

$$Distance(P_i, P_j) = \sqrt{(P_i - P_j)^2}$$

- After the calculation of the respective distances of each point, the values are sorted in a separate array and unique values are taken out. The array is represented as:

$$\begin{Bmatrix} D_1 \\ D_2 \\ D_3 \\ \vdots \\ D_n \end{Bmatrix} = \begin{Bmatrix} (d_{11}, d_{12}, \cdots , d_{1m}) \\ (d_{21}, d_{22}, \cdots , d_{2m}) \\ (d_{31}, d_{32}, \cdots , d_{3m}) \\ \vdots \\ (d_{n1}, d_{n2}, \cdots , d_{nm}) \end{Bmatrix}$$

And the values obtained are denoted as

$$D = [\, D_1\, D_2\, D_3 \ldots\, D_n\,]$$

Sorted array is represented as S_D=Sort $(D)_{Asc}$

Whereas the unique values are represented as

$$U_D = \cup\, S_D = [u_{D1}\, u_{D2} \ldots\, u_{Dn}\,]$$

- The U_D thus created is termed as the "cancellable fingerprint template" [10]. The cancellable template U_D is employed in the generation of non-invertible cryptographic key.

CHAPTER 6

DES ALGORITHM

This block is for getting a Final Secured Cryptographic key having binary bits 0's and 1's from cancellable fingerprint template. Key will be represented as

$$K = [\, K_1\, K_2\, K_3 \ldots\, K_n\,]$$

The Data Encryption Standard (DES) is a symmetric-key block cipher published by the national Institute of Standard and Technology (NIST). It has been the widely used symmetric –key block cipher since its publication. The algorithm transforms 64-bit input in a series of steps into a 64-bit output. The same steps, with the same key, are used to reverse the encryption. At the encryption site, DES takes a 64-bit plaintext and creates a 64-bit cipher text. At the decryption site, DES takes a 64-bit cipher text and creates a 64-bit block of plain text.

6.1 DES Encryption:

The overall scheme for DES encryption is illustrated in Figure 18. As with any encryption scheme, there are two inputs to the encryption function: the plaintext to be encrypted and the key. In this case, the plaintext must be 64 bits in length and the key is 56 bits in length.

Looking at the left-hand side of the figure, we can see that the processing of the plaintext proceeds in three phases. First, the 64-bit plaintext passes through an initial permutation (IP) that rearranges the bits to produce the permuted input. This is followed by a phase consisting of sixteen rounds of the same function, which involves both permutation and substitution functions. The output of the last (sixteenth) round consists of 64 bits that are a function of the input plaintext and the key. The left and right halves of the output are swapped to produce the pre-output. Finally, the pre-output is passed through a permutation that is the inverse of the initial permutation function, to produce the 64-bit cipher text. With the exception of the initial and final permutations, DES has the exact structure of a Feistel cipher, as shown in Figure above.

The right-hand portion of Figure below shows the way in which the 56-bit key is used. Initially, the key is passed through a permutation function. Then, for each of the sixteen rounds, a sub key (K_i) is produced by the combination of a left circular shift and a permutation. The permutation function is the same for each round, but a different sub-key is produced because of the repeated shifts of the key bits.

Here in this project we used DES algorithm mat-lab code for encryption to generate Output secure Cryptography Key.

6.2 Strengths of DES Key Size

- 56-bit keys have $2^{56} = 7.2 \times 10^{16}$ values
- brute force search looks hard
- still must be able to recognize plaintext

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

While fingerprint biometrics presents obvious advantages over password and token-based security, the difficulty in assuring the integrity of the key is one of the most important problems associated with cryptosystems. Generating cryptographic keys from cancellable biometrics has received considerable attention in recent years. We outlined several advances that originated both from the cryptographic encryption of cancellable fingerprint biometrics to address problem of compromised fingerprint database. In particular, we outlined the advantages of cancellable biometrics over other approaches and presented a proficient approach for enhancing uneven fingerprint images and proposed an efficient algorithms for filling up the gaps in ridge structures after segmentation and to remove multiple minutiae from thinned image.

Pre-processing time for the proposed algorithm is 3.45 sec, and time taken by DES key generation algorithm is 1.75 sec. Average processing and comparison is 5.2 sec. For future work, we intend to develop an algorithm for unique cancellable fingerprint template generation irrespective of the orientation of user given fingerprint image. We would also present different proficient algorithms for generation of cryptographic key from generated cancellable fingerprint.

References

[1] N. Lalithamani, Dr. K.P. Soman, AMRITA Vishwa Vidyapeetham , An Efficient Approach For Non-Invertible Cryptographic Key Generation From Cancellable Fingerprint Biometrics, 2009 International Conference on Advances in Recent Technologies in Communication and Computing

[2] Yifei Wan, and Anil Jain , Fingerprint Image Enhancement: Algorithm and Performance Evaluation Lin Hong, IEEE transactions on pattern analysis and machine intelligence, vol. 20, no. 8, August 1998.

[3] Nigel Whyte and Dayu Chen, Research Manual Fingerprint Recognition, institue of technology Carlow.

[4] Wenzhou Liu, Xiangping Meng, Linna Li and Quande Yuan (2008), A kind of Effective Fingerprint Recognition Algorithm and Application In Examinee I dentity recognition available: ftp://ftp.computer.org/press/outgoing/proceedings/csse08/data/3336d035.pdf

[5] Manvjeet Kaur, Mukhwinder Singh, Akshay Girdhar, Parvinder S. Sanhu (2008).Fingerprint verification system using minutiae extraction technique available: http://www.waset.org/journals/waset/v46/v46-85.pdf

[6] N. Lalithamani, K.P. Soman, "An Effective Scheme for Generating Irrevocable Cryptographic Key from Cancelable Fingerprint Templates", International Journal of Computer Science and Network Security, Vol.9, No.3, pp: 183- 193, 2009.

[7] Wikipedia. (2010). **Segmentation**, available:http://en.wikipedia.org/wiki/Segmentation (image processing).

[8] Wikipedia. (2010). **Edge detection**, available: http://en.wikipedia.org/wiki/Edge_detection.

[9] Feature Detector – Canny edge detector http://homepages.inf.ed.ac.uk/rbf/HIPR2/canny.htm#1

[10] “RSA Factoring Challenge from http://en.wikipedia.org/wiki/RSA_Factoring_Challenge.