Biometric Encryption System for Increased Security

Ranjith JAYAPAL and Pramod GOVINDAN
Electrical Engineering, University of North Florida, Jacksonville, Florida 32224, USA

ABSTRACT

In this highly-interconnected world, most of our daily activities are computer based, and the data transactions are protected by passwords. These passwords identify various entities such as bank accounts, mobile phones, etc. People might reuse the same password, or passwords related to an individual that can lead to attacks. Indeed, remembering several passwords can become a tedious task. Biometrics is a science that measures an individual's physical characteristics in a unique way. Biometrics serve as a method to replace the cumbersome use of complex passwords. By using a Biometric Encryption method, one can personalize the biometric to encode a PIN, a password, or an alphanumeric string, for a multitude of applications such as, bank ATMs, building access, and computer terminal access. Moreover, the database only needs to store the biometrically encrypted PIN or password, not the large biometric sample.

Keywords: Fingerprint, Image Enhancement, Image Binarization, Minutiae Extraction and Cryptography.

1. INTRODUCTION

Cryptography is the science of secret writing which is very helpful in communicating over the networks such as the internet. Combining cryptography and biometrics together is known as biometric cryptosystem. In this method, cryptography will provide the high security level and biometrics will help to avoid remembering passwords. In addition, the cryptographic keys are generated from the user's biometric templates. Unless the same person participates again, the system will not reveal the previously stored keys for verification.

There are several types of cryptosystems available for biometric applications such as key release, key binding and key generation cryptosystems [1]. In the key release cryptosystems, the key will be released after the given biometric sample is verified. In the key binding cryptosystems, the biometric data and the cryptographic keys are combined. Therefore, the key will not be generated unless the same person is involved in the system. In the key generation cryptosystems, the secret key will be generated by a special algorithm for given biometrically extracted points.

This research attempts to provide a biometric encryption system based on fingerprint-based identification using minutiae extraction. The various steps involved in this process are explained and demonstrated with the implementation results. Even though this study focuses on fingerprint-based identification, this system can be modified or enhanced to experiment on diverse types of biometrics such as iris, face recognition etc. to suit the requirements of various applications. An overview of the biometric encryption system with the advantages and disadvantages are described in section II. Section III provides the details of various algorithm implementation steps used in this study for fingerprint identification using minutiae extraction, and matching performance results. Section IV demonstrates the biometric key generation algorithm implemented in this study with the results. The details described in this paper are based on [2].

2. BIOMETRIC ENCRYPTION OVERVIEW

A typical block diagram of biometric encryption process is shown in Figure 1. The fingerprint images from relevant individuals are scanned by the sensor and these images are bound with a random encryption key, and the resultant images (called BE templates) are stored in a template database (smart card, laptop, cellphone, etc.). Both the original image and the key are destroyed for security [3]. When an individual requires access, that individual's fingerprint is scanned by the sensor and the image is applied to a key retrieval process, by matching the images from the template database. Only if there is a close match with one of the images from the template database, the user is given access.

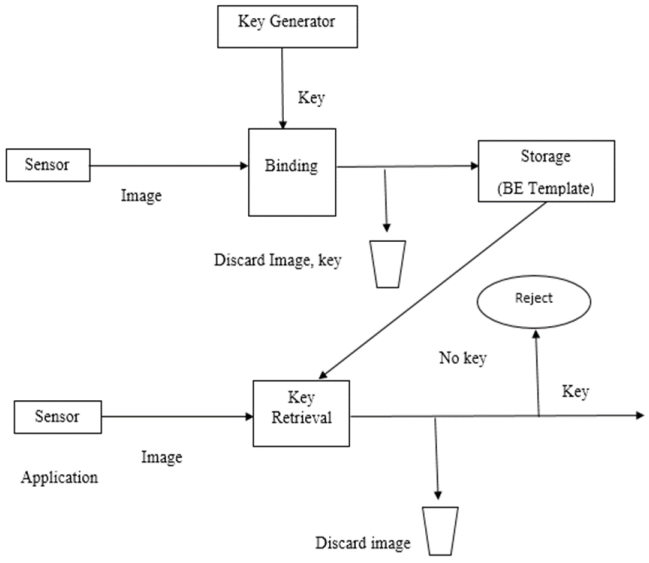

Figure 1. High-Level Diagram of a Biometric Encryption Process.

"This work was supported by the UNF- College of Computing, Engineering & Construction at University of North Florida, Jacksonville, FL".

Mr. Ranjith Jayapal is with the UNF- College of Computing, Engineering & Construction at University of North Florida, Jacksonville, FL. (e-mail: ranjith24490@gmail.com).

Dr. Pramod Govindan is with the UNF- College of Computing, Engineering & Construction at University of North Florida, Jacksonville, FL. (e-mail: Pramod.govindan@unf.edu).

During the verification process, when the user presents his or her biometric sample to the system, the key values are compared with the previously stored key or template image [4]. Then the key or image will be retrieved from the storage database to allow access to the person. At the end of the verification, the retrieved key or image is discarded again [5]. This algorithm is designed to accept a slight variation of the given input samples. On the contrary, if the sample keys are not matched with each other, the system will automatically reject the input.

Privacy and Security Problems Involving a Biometric System

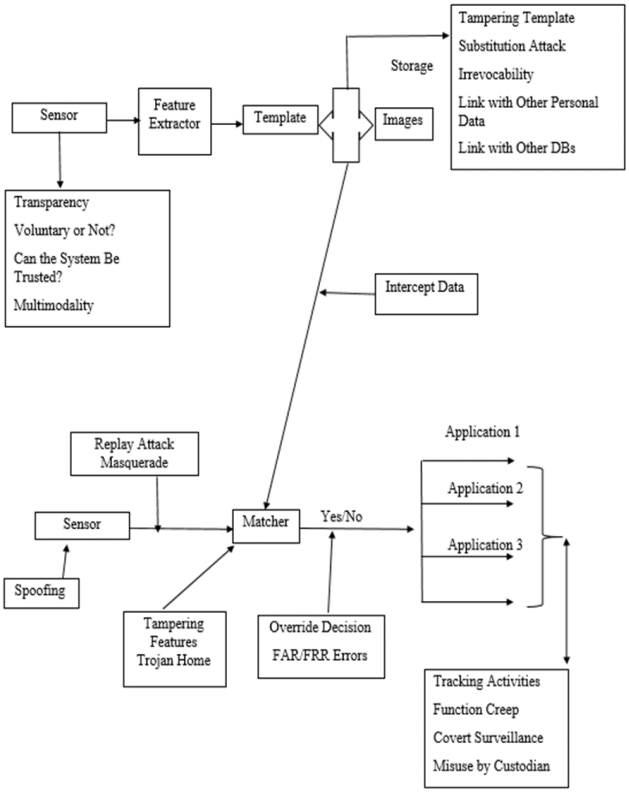

Figure 2. Privacy and Security Problems Involving A Biometric System.

The key problems involved in the biometric system are shown in Figure 2. This system explains how the template image has been stored and the data is kept in secret [6] [7] [8].

Spoofing: A biometric system sometimes can be fooled by applying fake fingerprints.

Replay attacks: A previously recorded image will be applied into the system, instead of giving an original one.

Masquerade attack: An artifact image can be drawn from the fingerprint template. Thus, whenever a person applies their fingerprint, the system will produce a match.

Tampering: An attacker will modify the templates to obtain a high verification score during the matching process. Thus, the system will be matched with all the given input data.

Trojan horse attacks: If the matcher is attacked by Trojan horse, all given inputs will result in a high verification score.

Substitution attack*:* Typically, the template is stored in the database, so the system must allow user verification. As an example, suppose an attacker were to get access to the template storage, he/she can modify the user's template to match with their own finger.

Overriding Yes/No response: The output of the system is always a binary Yes/No (i.e., match/no match) response.

Insufficient accuracy of many commercial biometric systems: High False Recognition Rates causes inconvenience for users to lower a verification threshold. This gives rise to False Acceptance Rate, which, in turn, lowers the security level of the system.

3. FINGERPRINT BASED IDENTIFICATION

There are about 18 different models of biometrics in the recognition method such as fingerprint, face, iris, palm print, hand geometry, voice, and gait and so on. Humans have 10 fingerprints, more than five times the amount of other biometrics, like iris or facial recognition [9] [10]. Due to these benefits, this research focuses on developing a basic system for fingerprint identification for biometric encryption.

Fingerprint Recognition Using Standardized Fingerprint Model

This study implements a fingerprint recognition system as shown in Figure 3. First, the fingerprint image is preprocessed, and it converts the image into binary. Then the morphological operation is applied to get a thinning image. Finally, the minutia points are extracted, and their corresponding values are found [11]. MATLAB scripts were implemented for pre-processing, thinning and minutiae extraction. The fingerprint image used in this study is taken from the Fingerprint verification competition (FVC2004) Database 4 [12]. The various steps involved in this implementation are explained in Figure 3.

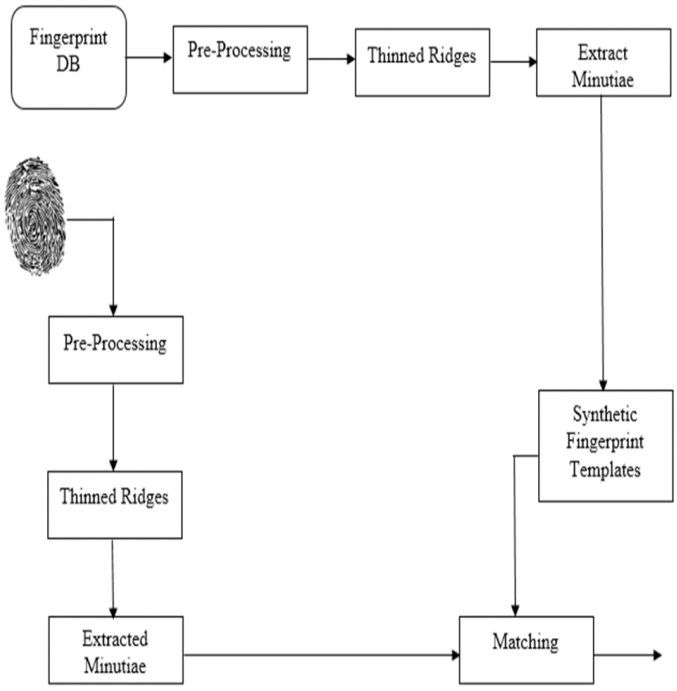

Figure 3. Fingerprint Recognition Using Standardized Fingerprint Model.

Fingerprint Recognition Steps

As shown in Figure 4, there are three steps involved in this model: (I) pre-processing, (II) minutiae extraction and (III) post processing.

Pre-processing: During the pre-processing stage, the captured image is enhanced by a histogram technique. If there is any noise in the image, it can be removed by image filtering techniques such as Gaussian, Speckle, Salt & Pepper methods [8] [14]. Furthermore, the image is sharpened to find the edge detection of the enhanced image.

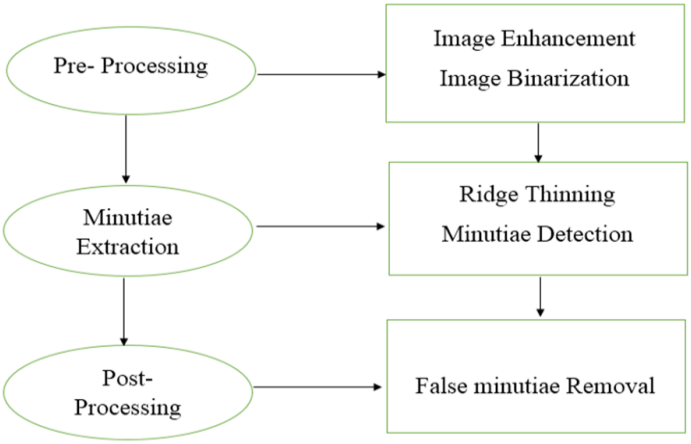

Figure 4. Fingerprint Recognition Steps.

Minutiae extraction: A thinning process is used to remove portions of foreground pixels in the binary image as shown in Figure 5(a). Thinning is a morphological operation and it is widely used in many applications [10] [15] [16]. It is very useful for skeletonizing. In this study, thinning is used for cleaning up the resultant output of edge detectors.

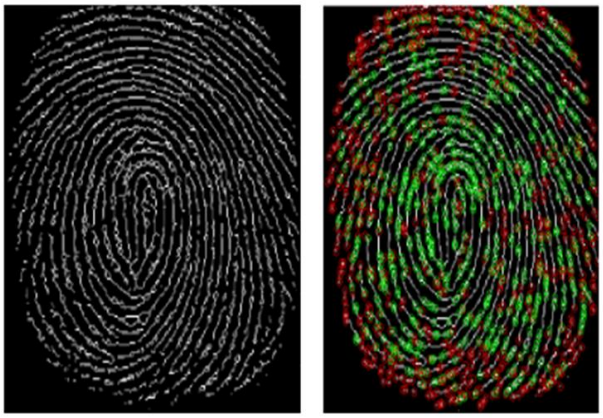

Figure 5. (a) Ridge Thinning (b) Minutiae Extraction.

In this study, the correlation method is used for minutiae extraction model as shown in Figure 5(b). A skeleton image contains eight neighborhoods of its ridge patterns, each ridge pixel is scanned by 3×3 window operation [8]. The crossing numbers are computed by half the sum of the 8 neighbor's pairs, as shown in Table 1, Table 2 and Eq. (1) The ridge pixels are classified as ridge ending, and bifurcation points with their corresponding values. For instance, the crossing number value "1" represents the ridge ending point and the crossing number value "3" represents the bifurcation point.

Table 1. Crossing Number Properties.

Property	Crossing Number
Isolated Point	0
Ridge Ending Point	1
Continuing Ridge Point	2
Bifurcation Point	3
Crossing Point	4

$$CN = 0.5\sum_{i=1}^{9}|P_i - (P_{i+1})|,\ P_9 = P_1 \qquad \text{Eq. (1)}$$

Where Pi is the pixel value (where possible values are 0 and 1) about P. For a pixel, P, its eight neighboring pixels are represented in an anticlockwise direction as follows in Table 2.

Table 2. (a) Eight Neighboring Pixels (b) Ridge Ending (c) Bifurcation Point.

P_4	P_3	P_2
P_5	P	P_1
P_6	P_7	P_8

0	0	1
0	1	0
0	0	0

0	1	0
0	1	0
1	0	1

Post-processing (False Minutiae removal): Some of the minutiae points extracted during the preprocessing stage are incorrect [17]. For example, as shown in Figure 6(a) false ridge breaks due to insufficient amount of ink and ridge cross-connections due to over inking are not eliminated. Figure 6(b) shows the minutiae points after the false minutiae removal.

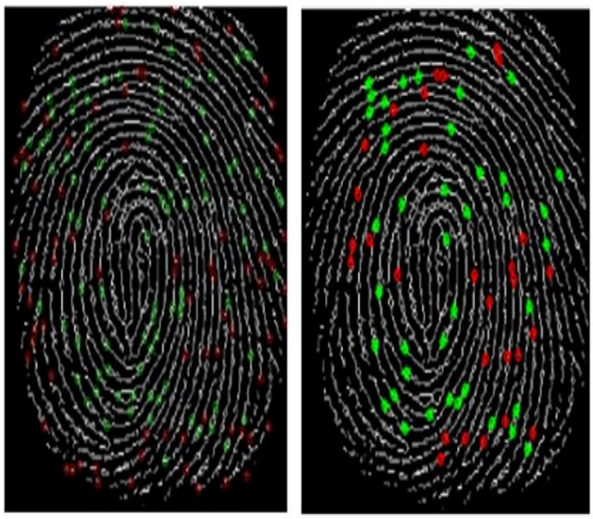

Figure 6. (a) Minutiae Detection Points (b) After False Minutiae Removal.

Matching process

The Veri-Finger 9.0/MegaMatcher9.0 algorithm was used for minutiae extraction and fingerprint matching [17]. The minutiae templates of 2,000 file fingerprints from NIST SD4 were used for fingerprint identification experiments. The minutiae template includes the coordinates (x, y), and the direction (θ) for each minutia values [8] [13]. The algorithm was implemented in MATLAB and run on a machine with Intel Core i5 2430M 2.40GHz, 4GB RAM and 64-bit Windows 7 operating system. The output result values are shown in Figure 7 and Figure 8 for 1:N matching (identification) and 1:1 matching (verification) respectively.

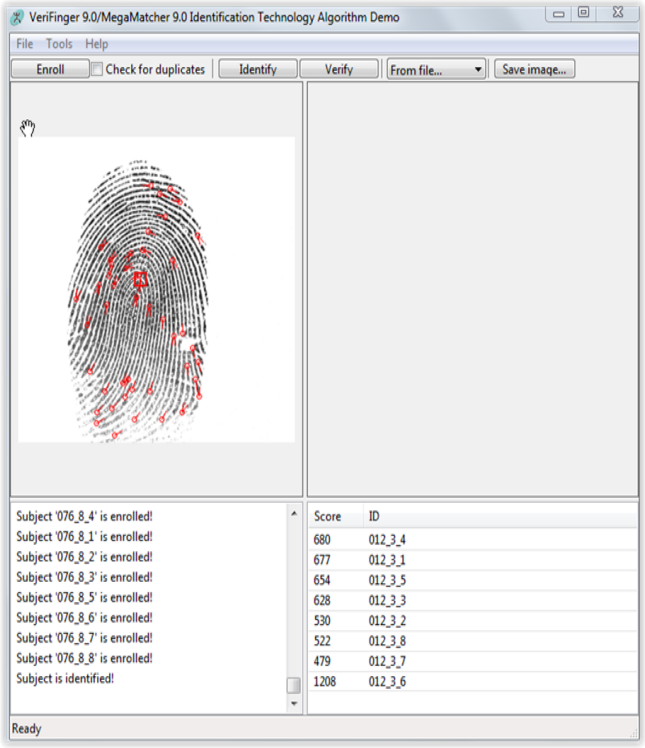

Figure 7. Enrollment/Identification Process (1:N) Matching

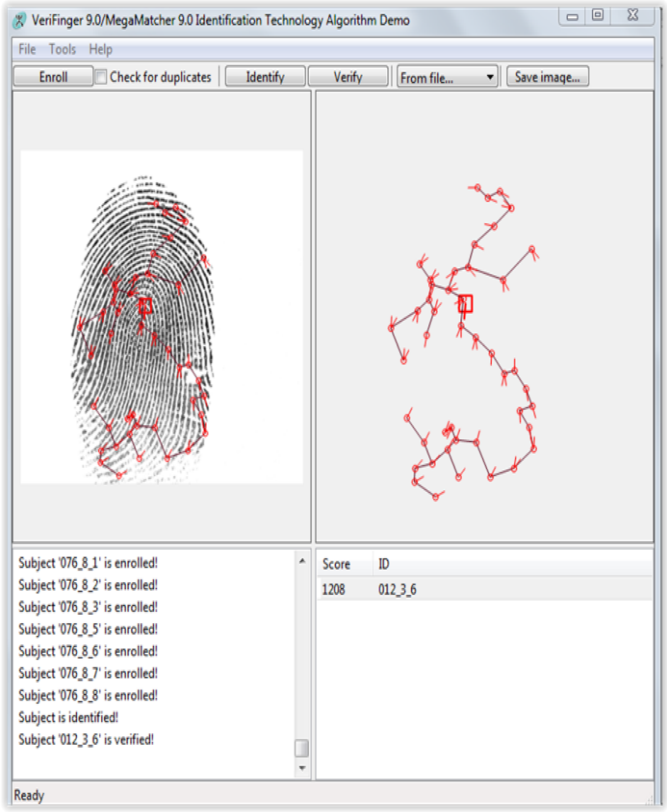

Figure 8. Verification Process (1:1) Matching

By applying a varying score threshold to the similarity score, pairs of False Acceptance Rate (FAR) and False Rejection Rate (FRR) are calculated as shown in Table 3.

Table 3. FAR and FRR Performance Results

Percentage Match	False Acceptance Rate (FAR)	False Rejection Rate (FRR)
25%	0.693	0.113
30%	0.680	0.117
35%	0.677	0.121
45%	0.593	0.121
50%	0.549	0.122

The FAR and FRR were calculated for five different values of percentage matching parameters. The matching performance of the proposed system can be seen in Table 3 and Figure 9. The percentage match parameter is shown alongside FAR and FRR to illustrate the tradeoff between security and matching. For higher security (a higher percentage match), FAR is low whereas FRR is high. As shown in Figure 9, with a percentage match setting of 50 and above, the FAR went down. However, that also generated many false rejections (false positives). Therefore, this study recommends a value of around 35 for the percentage matching parameter.

There are two common ways of plotting performance evaluation results such as Detection Error Trade-off (DET) graph and Receiver Operating Characteristic (ROC) graph. In this study, DET curve is used, because it is a most commonly used and far better at highlighting areas of interest as shown in Figure 9.

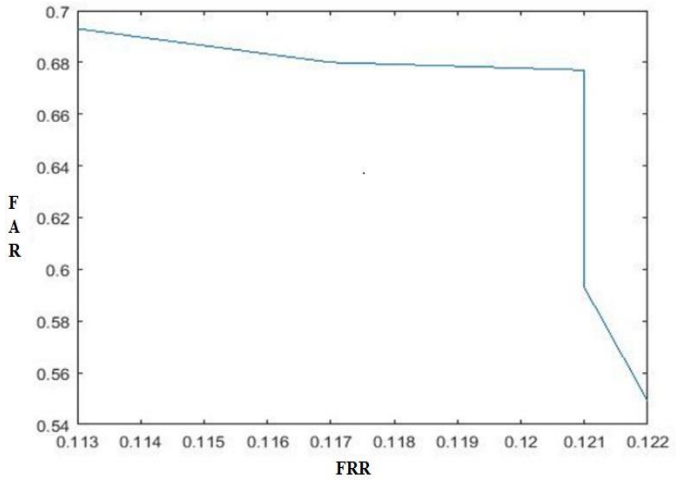

Figure 9. DET Curve for The Experiment

4. CRYPTOGRAPHNIC KEY GENERATION FROM BIOMETRIC

In this study, the key generation algorithm produces a 64 bit key, which can be used in Data Encryption Standard Algorithm [18]. As shown in Figure 10, the cryptographic key is generated by using the minutiae coordinates and angles. Figure 11 shows the minutiae sets extracted in this study, where the Ridge Ending (RIG) is marked in red, and Bifurcation (BIF) is marked in green. According to the values given in Table 1, the minutiae coordinates, and angles are determined (see Table 4) for the image shown in Figure 11.

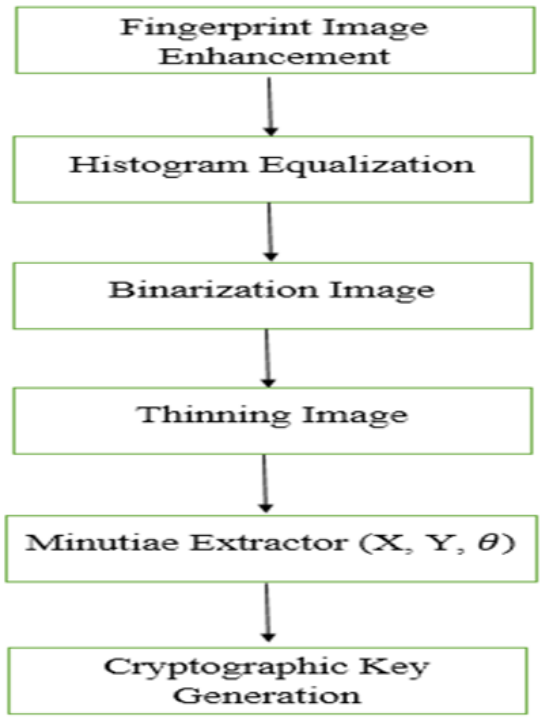

Figure 10. Block Diagram of Cryptographic Key Generation.

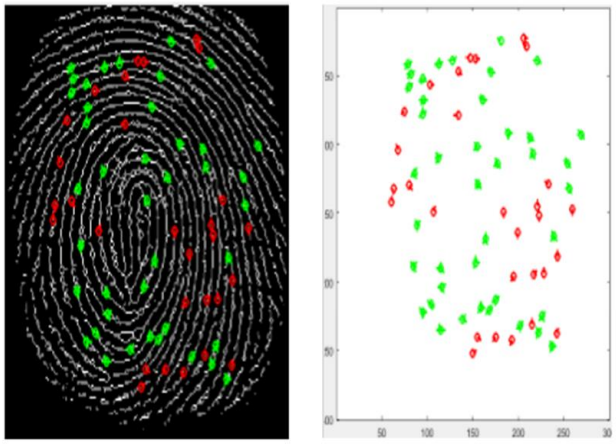

Figure 11. True Minutiae Sets.

The resultant outputs in Table 4 are in decimal for X and Y coordinates and radians for angle θ. These output values are converted into binary representation [17] [19], because during the preprocessing time the input image values are resized into 256x256 array for minutiae extraction. Then the average values of minutiae point X and Y coordinates and angle θ are calculated [20].

Finally, the binary values of the average of X, Y and θ are concatenated to get a single binary vector which is the private key corresponding to the original fingerprint image [17] [21].
The algorithm steps are given below:

- Find the average values X_{ave}, Y_{ave} and θ_{ave} from Table 4.
- Get the corresponding binary values XB (28bits), YB (28 bits), and θB (8 bits) of X_{ave}, Y_{ave} and θ_{ave} respectively.
- Concatenate all the binary values in the following order MB = {XB, YB, θB}.

The algorithm results for the original fingerprint image values (Table 4) are given below.

X_{ave} = 14320/65; XB = 11010010000110100000111011 00
Y_{ave} = 12171/65; YB = 10110010100100100110010010 01
θ_{ave} = 1027/65; θB = 10011110

The generated cryptographic key of length 64 for the biometric image in Figure 11 is as follows:

1101001000011010000011101100101100101001001001100100100110011110

The original image, the encrypted image and the output image respectively are shown in Figure 12 (a) (b) (c).

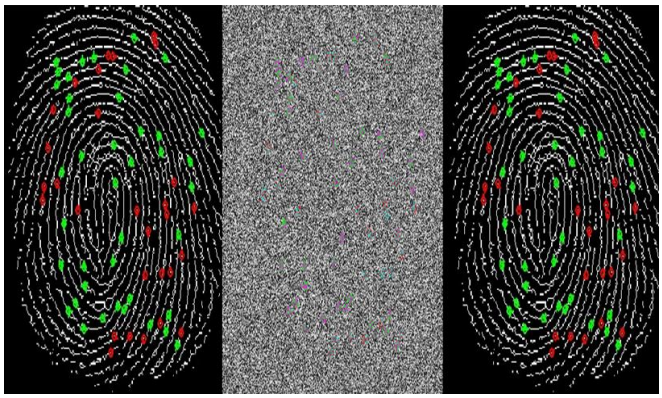

Figure 12. (a) Original Image (b) Encrypted Image (c) Decrypted Image

Table 4. Values of Minutiae Co-Ordinates and Angles Generated for the Image in Figure 11.

X	Y	θ	Minutiae Type
61	291	0	3
64	132	4	1
65	271	16	3
83	214	18	3
85	160	4	3
96	161	20	3
105	115	21	3
116	145	20	3
133	279	14	3
135	288	13	1
140	282	29	3
154	181	2	3
163	339	11	3
163	380	25	3
175	70	23	3
177	353	25	3
184	82	24	3
186	330	9	3
188	123	23	3
192	25	24	3
192	262	31	3
193	13	24	3
193	57	24	3
195	151	7	3
196	167	22	3
197	72	8	3
198	219	1	3
199	118	8	3
202	55	8	3
203	87	24	1
203	240	16	3
205	80	8	3
207	25	8	3
207	336	23	3
208	311	6	3
213	317	22	3
214	147	24	3
214	258	1	3
216	14	9	3
219	208	17	3
223	166	8	1
225	65	9	3
237	371	4	3
239	177	28	1
242	150	27	3
258	337	19	3
269	94	26	1
280	176	14	3
285	273	18	3
294	111	27	1
300	28	10	3
306	59	27	1
306	215	31	3
307	228	16	3
311	354	3	3
316	148	13	1
318	93	11	1
319	353	19	3
349	228	31	1
351	272	16	3
355	174	14	3
361	266	16	3
371	72	12	1
374	305	0	3
385	98	12	3

5. CONCLUSION

This research demonstrates the use of fingerprint samples to generate a cryptographic key for increased security. There are many biometric samples available for use in this recognition model; however, the fingerprint was selected as the biometric. Fingerprints are stable and remain consistent throughout a person's lifetime. During this work, it was discovered that each person has unique minutiae coordinates, and orientation angles for their fingerprints. In this study, the Veri-Finger 9.0/MegaMatcher9.0 algorithm is used for minutiae extraction and matching process. The proposed method can be used as an efficient biometric security system for application such as online banking, border security control, forensics etc. By using NIST Special Database 4, which contains very low quality images, the matching process implemented in the proposed method can be demonstrated and improved further to suit the high security applications. Furthermore, the possibility of implementing the fingerprint identification process on Field Programmable Gate Array (FPGA) can be explored for higher performance. By utilizing the parallel processing capabilities of FPGA, the overall computational speed of the algorithm can be improved.

6. REFERENCES

[1] Davide Maltoni, Dario Maio, Anil K. Jain and Salil Prabhakar, "Fingerprint Sensing," in **Handbook of Fingerprint Recognition**, Springer, 2009, pp. 57-95.

[2] Jayapal, R. "**Biometric encryption system for increased security**," [online] Available: https://digitalcommons.unf.edu/cgi/viewcontent.cgi?referer=&httpsredir=1&article=1773&context=etd [Accessed 06 February 2018].

[3] A.Jaya Lakshmi and I. Ramesh Babu, "**Design of Secured Key Generation Algorithm Using Fingerprint Based Biometric Modality**," in IOSR Journal of Engineering, 2012.

[4] Jayapal, R., & Govindan, P. (2016). **Biometric encryption system for increased security**. *The IEEE International Carnahan Conference on Security Technology (ICCST)* . Orlando, FL, USA: IEEE.

[5] B. Raja Rao, Dr. E.V.V.Krishna Rao, S.V.Rama Rao and M. Rama Mohan Rao, "**Finger Print Parameter Based Cryptographic Key Generation**," International Journal of Engineering Research and Applications, vol. 2, no. 6, pp. 1598-1604, 2012.

[6] Ann Cavoukian and Alex Stoianov, "**Biometric Encryption Chapter from the Encyclopedia of Biometrics**," Information and Privacy Commissioner, Ontario, Canada, [Online]. Available: https://www.ipc.on.ca/wp-content/uploads/Resources/bio-encrypt-chp.pdf. [Accessed 20 January 2016].

[7] Anil K. Jain, Karthik Nandakumar and Abhishek Nagar, "**Biometric Template Security**," EURASIP Journal on Advances in Signal Processing, vol. 2008, no. Biometrics, pp. 1-17, 2008.

[8] Aditi Roy, Nasir Memon and Arun Ross, "**Master Print: Exploring the Vulnerability of Partial Fingerprint-based Authentication Systems**," IEEE Transactions on Information Forensics and Security, pp. 1-13, 2017.

[9] A.Jagadeesan and Dr.K.Duraiswamy, "**Secured Cryptographic Key Generation from Multimodal Biometrics: Feature Level Fusion of Fingerprint and Iris**," in International Journal of Computer Science and Information Security, 2010.

[10] Mofeed Turky Rashid and Huda Ameer Zaki, "**RSA Cryptographic key generation using fingerprint minutiae**," Iraqi commission for computers & informatics, vol. 1, no. 1, 2014.

[11] Le Hoang Thai and Ha Nhat Tam, "**Fingerprint recognition using standardized fingerprint model**," International journal of computer science issues, vol. 7, no. 3, pp. 11-17, 2010.

[12] Seungjin Sul, "**Classification-based Automatic Fingerprint Identification System for Large Distributed Fingerprint Database**," Biometrics & Biostatistics, vol. 2, no. 2JBMBS, 2011.

[13] Kai Cao and Anil K Jain, "**Learning Fingerprint Reconstruction from Minutiae to Image**," IEEE.

[14] Roli Bansal, Priti Sehgal and Punam Bedi, "**Minutiae Extraction from Fingerprint Images - a Review**," IJCSI International Journal of Computer Science Problems, vol. 8, no. 5, pp. 74-85, 2011.

[15] Anil K. Jain, Arun Ross and Sharath Pankanti, "**Biometrics: A Tool for Information Security**," IEEE Transactions on Information's forensics and security, vol. 1, no. 2, pp. 125-143, 2006.

[16] Roy, S. M. **Biometrics Data Security Techniques for Portable Mobile Devices**. *INAE Letters*, pp. 123–131.

[17] "**False Acceptance Rate (FAR) and False Recognition Rate (FRR) in Biometrics**," BIOMETRIC TECHNOLOGY, 2007-2016. [Online]. Available: https://www.bayometric.com/false-acceptance-rate-far-false-recognition-rate-frr/. [Accessed 28 April 2017].

[18] R. K. Sharma, "**Generation of Biometric Key for Use in DES**," [Online]. Available: https://arxiv.org/ftp/arxiv/papers/1302/1302.6424.pdf. [Accessed 13 September 2016].

[19] Aniket Kore, Shiwani Gupta and Kiran Bhandari, "**Symmetric Encryption Algorithm Based on Keys Generated from Biometrics**," International Journal of Recent Trends in Engineering & Research, vol. 02, no. 2455-1457, pp. 343-352, 2016.

[20] Lin You, Guowei Zhang and Fan Zhang, "**A Fingerprint and Threshold Scheme-Based Key Generation Method**," pp. 615-619.

[21] Bon K. Sy and Arun P. Kumara Krishnan, "**Generation of cryptographic keys from personal Biometrics: An illustration based on fingerprints**,"