Image Encryption Using Modified AES with Bio-Chaotic

Haider Kadhim Hoomod[1] Mustafa Hussein Zewayr[2]

Assistant Professor Doctor[1], Under Graduate Student[2]

"Mustansiriya University"

Department of Computer Science

College of Education

Baghdad

Iraq

ABSTRACT

Due to the vast using of digital images and the fast evolution in computer science and especially the using of images in the social network.This lead to focus on securing these images and protect it against attackers, many techniques are proposed to achieve this goal. In this paper we proposed a new chaotic method to enhance AES (Advanced Encryption Standards) by eliminating Mix-Columns transformation to reduce time consuming and using palmprint biometric and Lorenz chaotic system to enhance authentication and security of the image, by using chaotic system that adds more sensitivity to the encryption system and authentication for the system.

Keywords: *Lorenz, Palmprint, Modified AES.*

1. INTRODUCTION

Image encryption techniques are extensively used to overcome the problem of secure transmission for both images and text over the electronic media by using the conventional cryptographic algorithms. But the problem is that it cannot be used in case of huge amount of data and high resolution images [1]. Security in transmission of digital images has its importance in today's image communications, due to the increasing use of images in industrial process, it is essential to protect the confidential image data from unauthorized access, Image security has become a critical issue. The difficulties in ensuring individuals privacy become increasingly challenging [2].

Chaos has been introduced to cryptography thanks to its ergodicity, pseudo-randomness and sensitivity to initial conditions and control parameters, which are close to confusion and diffusion in cryptography. These properties make chaotic systems a potential choice for constructing cryptosystems [3].

Increasing researches of image encryption technology are based on chaotic systems. Recently there have been many papers on chaotic encryption scheme [4].

The nature of chaos has initiated a lot of interests in different engineering disciplines, where cryptography must be one of the most potential applications. The distinct properties of chaos, such as ergodicity, quasi-randomness, sensitivity dependence on initial conditions and system parameters, have granted chaotic dynamics as a promising alternative for the conventional cryptographic algorithms.

Unlike the conventional cryptographic algorithms which are mainly based on discrete mathematics, chaos-based cryptography is relied on the complex dynamics of nonlinear systems or maps which are deterministic but simple. Therefore, it can provide a fast and secure means for data protection, which is crucial for multimedia data transmission over fast communication channels, such as the broadband internet communication [5].

Instead of using the traditional way of cryptography for image encryption we can also use biometric e.g. fingerprint, iris, face, voice etc for the same purpose. The main advantage of a biometric is that it is ever living and unstable characteristics of a human being and it cannot be compromised [1].

2. RELATED WORK

Salim M. Wadi, NasharuddinZainal [6] new modification to AES-128 algorithm which reduce the computations and hardware requirements are proposed by enforcement Mixcolumn transformation in five rounds instead of nine rounds as in original AES-128. Second proposed is suggest new simple S-box used for encryption and decryption. The implementation of advanced encryption standard algorithm is important requirement where many researches proposed different items to this purpose. A simply item proposed in this paper to speedy, low cost implementation of Modified Advanced Encryption Standard (MAES) cryptographic algorithm is 8085A microprocessor. The results prove that the modifications of AES make implementation it by 8085A microprocessor more effective.

Ajish S [7] in this paper, a new image encryption algorithm is proposed which is based on wavelet transform and dynamic S-box based AES algorithm. First of all, wavelet decomposition is used for concentrating original image in low-frequency wavelet coefficients , then dynamic S-Box based AES algorithm is applied to encrypt the low-frequency wavelet coefficients. In dynamic S-Box based AES algorithm the S-Box is generated from the key by using pairwise linear chaotic maps. Secondly, an XOR operation is used for high-frequency wavelet coefficients and the encrypted low-frequency wavelet coefficients (as a key stream), so that the image information contained in high-frequency wavelet coefficients is hidden; Thirdly, a wavelet reconstruction is used for spreading the encrypted low-frequency part to the whole image.

ChittaranjanPradhan, Ajay Kumar Bisoi [8] in this paper, they have tried to give focus on the security of the key used. Here, the proposed modified algorithms for the AES have been simulated and tested with different chaotic variations such as 1-D logistic chaos equation, cross chaos equation as well as combination of both. For the evaluation purpose, the CPU time has been taken as the parameter. Though the variations of AES algorithms are taking some more time as compared to the standard AES algorithm, still the variations can be taken into consideration in case of more sensitive information. As they are giving more security to the key used for AES algorithm.

3. BIOMETRIC

A biometricis defined as a unique, measurable, biological characteristic or trait for automatically recognizing or verifying the identity of a human being. Statistically analyzing these biological characteristics has become known as the science of *biometrics*. These days, biometric technologies are typically used to analyze human characteristics for security purposes [9].Biometrics offers certain advantages such as negative recognition and non-repudiation that cannot be provided by tokens and passwords.

Negative recognition is the process by which a system determines that a certain individual is indeed enrolled in the system although the individual might deny it. This is especially critical in applications such as welfare disbursement where an impostor may attempt to claim multiple benefits (i.e., double dipping) under different names. Non-repudiation is a way to guarantee that an individual who accesses a certain facility cannot later deny using it (e.g., a person accesses a certain computer resource and later claims that an impostor must have used it under falsified credentials) [10].

3.1 Palmprint

One of the reasons of this work is to suggest the usage of palmprint biometric in the development of a cryptographic construct. The palmprint based cryptosystem can have higher user acceptance and performance. In defiance of the recently popularity of palmprint based subsystem. For that the Palmprint biometric is suitable for everyone hand it is neither non-intrusive as it doesn't required many personality information of their user[11] [12].

Palm-prints are now also used in Automatic Fingerprint Identification System (AFIS)systems. The first reported commercially palmprint system was in Hungary in 1994, and by 1997 similarity processes were being built into other AFIS systems [13]. The palmprint features are shown in Figure (3.1) [14].

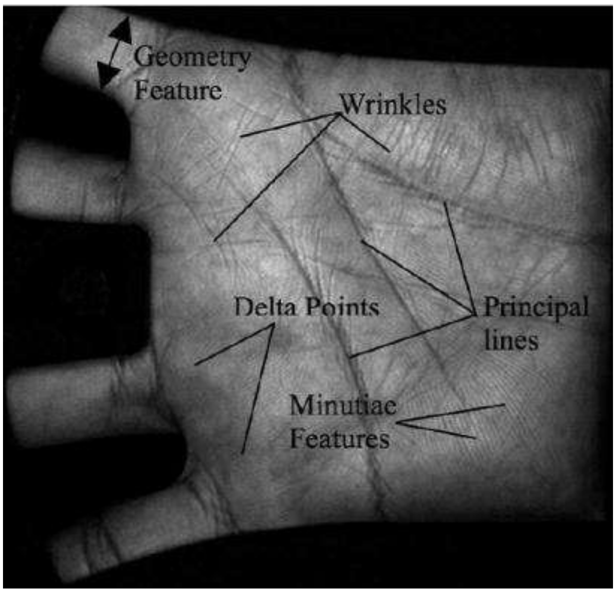


Figure 3.1: Different Feature of Palm

4. AES

In January 1997, the US National Institute of Standards and Technology (NIST) announced the start of an initiative to develop a new encryption standard: the AES. The new encryption standard was to become a Federal Information Processing Standard (FIPS), replacing the old Data Encryption Standard (DES) and triple-DES [15] p1.

This standard specifies the Rijndael algorithm, a symmetric block cipher that can process data blocks of 128 bits, using cipher keys with lengths of 128, 192, and 256 bits. Rijndael was designed to handle additional block sizes and key lengths, however they are not adopted in this standard. Throughout the remainder of this standard, the algorithm specified herein will be referred to as “the AES algorithm.” The algorithm may be used with the three different key lengths indicated above, and therefore these different "flavors" may be referred to as "AES-128", "AES-192", and "AES-256" [16].

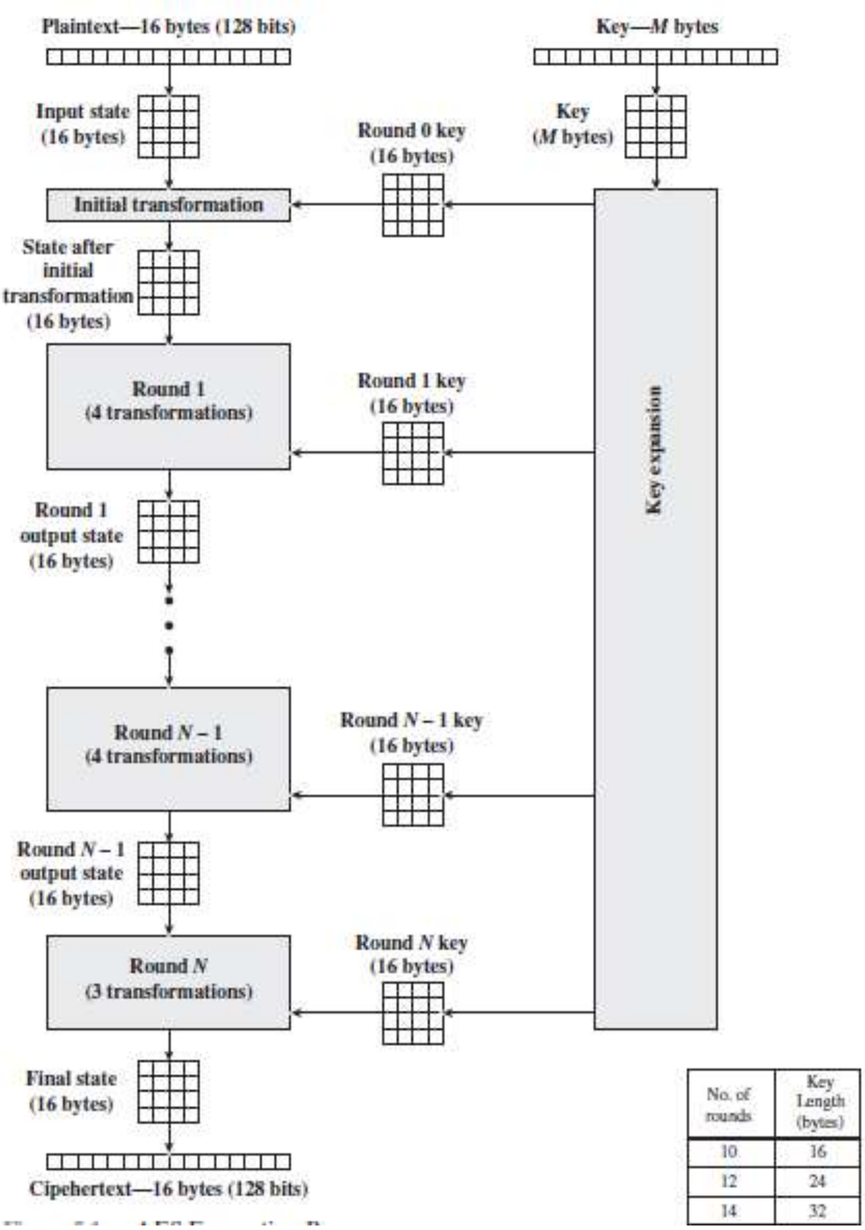
5. AES STRUCTURE

In the AES encryption process the cipher takes a plaintext block size of 128 bits, or 16 bytes. The key length can be 16, 24, or 32 bytes (128, 192, or 256 bits). The algorithm is referred to as AES-128, AES-192, or AES-256, depending on the key length. The input to the encryption and decryption algorithms is a single 128-bit block. In FIPS PUB 197, this block is depicted as a 4*4 square matrix of bytes. This block is copied into the State array, which is modified at each stage of encryption or decryption. After the final stage, State is copied to an output matrix. Similarly, the key is depicted as a square matrix of bytes.

This key is then expanded into an array of key schedule words. Each word is four bytes, and the total key schedule is 44 words for the 128-bit key. Note that the ordering of bytes within a matrix is by column. So, for example, the first four bytes of a 128-bit plaintext input to the encryption cipher occupy the first column of the in matrix, the second four bytes occupy the second column, and so on.

Similarly, the first four bytes of the expanded key, which form a word, occupy the first column of the w matrix. The cipher consists of N rounds, where the number of rounds depends on the key length: 10 rounds for a 16-byte key, 12 rounds for a 24-byte key, and 14 rounds for a 32-byte key. The first N-1 rounds consist of four distinct transformation functions: SubBytes, ShiftRows, MixColumns, and AddRoundKey, which are described subsequently. The final round contains only three transformations, and there is an initial single transformation (AddRoundKey) before the first round, which can be considered Round 0. Each transformation takes one or more 4*4 matrices as input and produces a 4*4 matrix as output.

Figure 4.1 shows that the output of each round is a 4*4 matrix, with the output of the final round being the ciphertext. Also, the key expansion function generates N+1 round keys, each of which is a distinct 4*4 matrix. Each round key serve as one of the inputs to the Add RoundKeytransformation in each round [17].


Plaintext—16 bytes (128 bits)
Key—M bytes
Input state (16 bytes)
Key (M bytes)
Round 0 key (16 bytes)
Initial transformation
State after initial transformation (16 bytes)
Round 1 (4 transformations)
Round 1 key (16 bytes)
Key expansion
Round 1 output state (16 bytes)
Round N – 1 (4 transformations)
Round N – 1 key (16 bytes)
Round N – 1 output state (16 bytes)
Round N (3 transformations)
Round N key (16 bytes)
Final state (16 bytes)
Ciphertext—16 bytes (128 bits)

No. of rounds	Key Length (bytes)
10	16
12	24
14	32

Figure 5.1: AES Structure

6. AES TRANSFORMATION

6.1 Sub-Bytes

It is a simple table lookup (Figure 5.5a). AES defines a 16*16 matrix of byte values, called an S-box (Figure 5.2a), that contains a permutation of all possible 256 8-bit values. Each individual byte of State is mapped into a new byte in the following way:

The leftmost 4 bits of the byte are used as a row value and the rightmost 4 bits are used as a column value. These row and column values serve as indexes into the S-box to select a unique 8-bit output value. For example, the hexadecimal value {95} references row 9, column 5 [17].

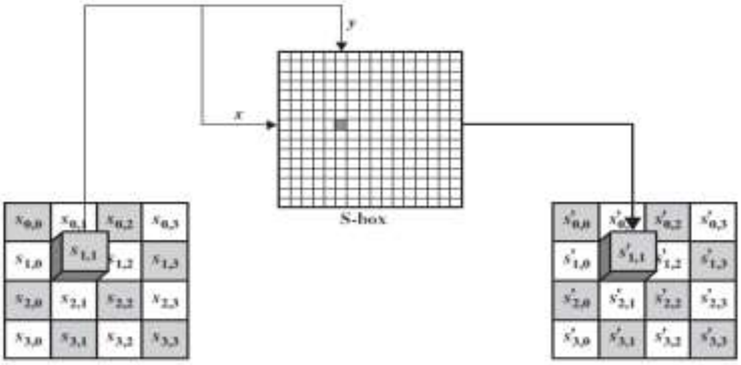
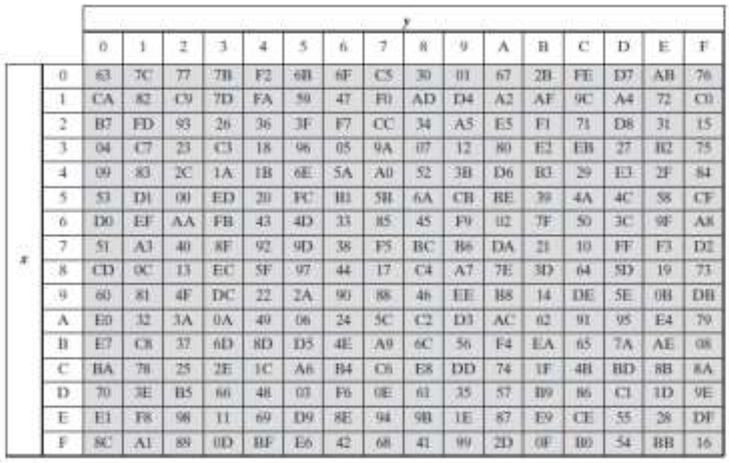
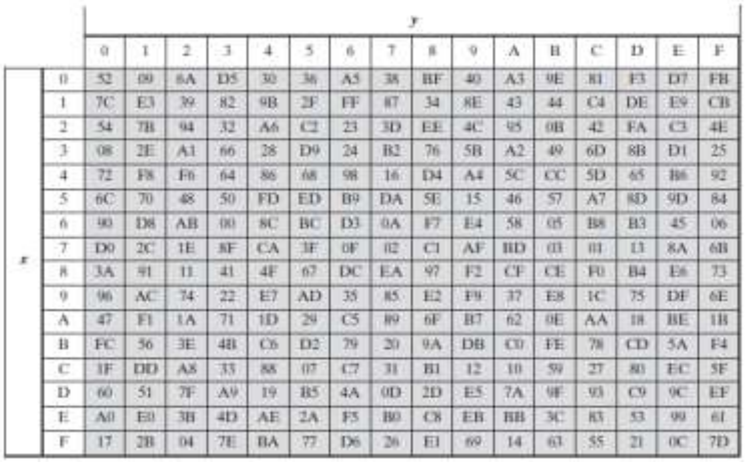

y
x
S-box
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s'_{0,0}$ $s'_{0,1}$ $s'_{0,2}$ $s'_{0,3}$
$s'_{1,0}$ $s'_{1,1}$ $s'_{1,2}$ $s'_{1,3}$
$s'_{2,0}$ $s'_{2,1}$ $s'_{2,2}$ $s'_{2,3}$
$s'_{3,0}$ $s'_{3,1}$ $s'_{3,2}$ $s'_{3,3}$

Figure 6.1:ByteSub acts on the individual bytes of the State



		y															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
	1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
	2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
	3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
	4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
	5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
	6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
	7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
x	8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
	9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
	A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
	B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
	C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
	D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
	E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
	F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

(a) S-Box



		y															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	52	09	6A	D5	30	36	A5	38	BF	40	A3	9E	81	F3	D7	FB
	1	7C	E3	39	82	9B	2F	FF	87	34	8E	43	44	C4	DE	E9	CB
	2	54	7B	94	32	A6	C2	23	3D	EE	4C	95	0B	42	FA	C3	4E
	3	08	2E	A1	66	28	D9	24	B2	76	5B	A2	49	6D	8B	D1	25
	4	72	F8	F6	64	86	68	98	16	D4	A4	5C	CC	5D	65	B6	92
	5	6C	70	48	50	FD	ED	B9	DA	5E	15	46	57	A7	8D	9D	84
	6	90	D8	AB	00	8C	BC	D3	0A	F7	E4	58	05	B8	B3	45	06
	7	D0	2C	1E	8F	CA	3F	0F	02	C1	AF	BD	03	01	13	8A	6B
x	8	3A	91	11	41	4F	67	DC	EA	97	F2	CF	CE	F0	B4	E6	73
	9	96	AC	74	22	E7	AD	35	85	E2	F9	37	E8	1C	75	DF	6E
	A	47	F1	1A	71	1D	29	C5	89	6F	B7	62	0E	AA	18	BE	1B
	B	FC	56	3E	4B	C6	D2	79	20	9A	DB	C0	FE	78	CD	5A	F4
	C	1F	DD	A8	33	88	07	C7	31	B1	12	10	59	27	80	EC	5F
	D	60	51	7F	A9	19	B5	4A	0D	2D	E5	7A	9F	93	C9	9C	EF
	E	A0	E0	3B	4D	AE	2A	F5	B0	C8	EB	BB	3C	83	53	99	61
	F	17	2B	04	7E	BA	77	D6	26	E1	69	14	63	55	21	0C	7D

(b) Invers S-Box

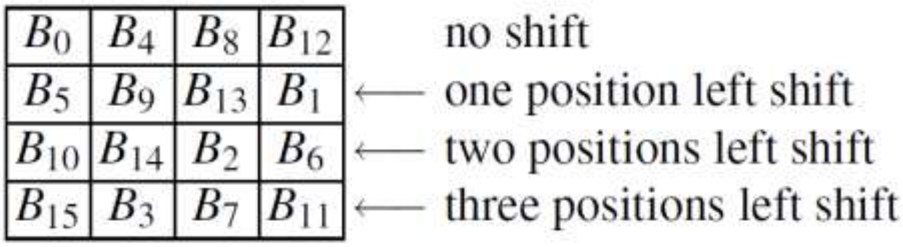
Figure 6.2: (a) S-box (b) Invers S-Box

The inverse of ByteSub is the byte substitution where the inverse table is applied. This is obtained by the inverse of the affine mapping followed by taking the multiplicative inverse in GF(2^8) [18].

6.2 Shift-Rows

The ShiftRows step operates on the rows of the state, it cyclically shifts the bytes in each row by a certain offset as shown in Figure 4. The first row is left unchanged. For the second row, a 1-byte circular left shift is performed. For the third row, a 2-byte circular left shift is performed.

For the fourth row, a 3-byte circular left shift is performed. The inverse shift row transformation, called InvShiftRows, used in the decryption, performs the circular shifts in the opposite direction for each of the last three rows, with a one-byte circular right shift for the second row, and so on [7].



B_0	B_4	B_8	B_{12}	no shift
B_5	B_9	B_{13}	B_1	⟵ one position left shift
B_{10}	B_{14}	B_2	B_6	⟵ two positions left shift
B_{15}	B_3	B_7	B_{11}	⟵ three positions left shift

Figure 6.3: ShiftRows Transformation

6.3 Mix-Columns

MixColumns is a Mixing function in the Cipher round. In the MixColumns step, In the MixColumns step, the four bytes of each column of the state are combined using an invertible linear transformation. The MixColumns function takes four bytes as input and outputs four bytes, where each input byte affects all four output bytes.

Together with ShiftRows, MixColumns provides diffusion in the Cipher. Figure 6 shows the MixColumns operates on the State column-by-column[19].

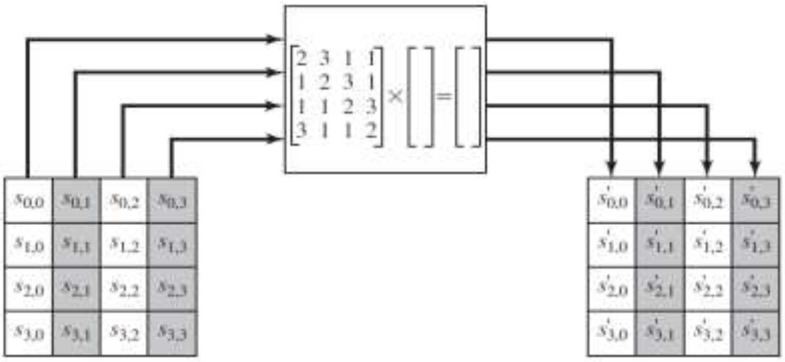

$\begin{bmatrix}2&3&1&1\\1&2&3&1\\1&1&2&3\\3&1&1&2\end{bmatrix} \times \begin{bmatrix}\ \end{bmatrix} = \begin{bmatrix}\ \end{bmatrix}$
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s'_{0,0}$ $s'_{0,1}$ $s'_{0,2}$ $s'_{0,3}$
$s'_{1,0}$ $s'_{1,1}$ $s'_{1,2}$ $s'_{1,3}$
$s'_{2,0}$ $s'_{2,1}$ $s'_{2,2}$ $s'_{2,3}$
$s'_{3,0}$ $s'_{3,1}$ $s'_{3,2}$ $s'_{3,3}$

Figure 6.4: MixColumns Transformation

6.4 Add Round Key

In the AddRoundKey step [3], the sub-key is combined with the state. For each round, a sub-key is derived from the main key using AES key schedule, each sub-key is the same size as the state. The sub-key is added by combining each byte of the state with the corresponding byte of the sub-key using bitwise Exclusive OR (XOR).

The inverse add round key transformation is identical to the forward add round key transformation, because the XOR operation is its own inverse [7].

7. LORENZ SYSTEM

The Lorenz equations are a fairly simple model in which to study chaos.

$$x = \sigma(y - x)$$

$$y = rc - xz - y$$

$$\mathrm{z} = \mathrm{xy} - \beta\mathrm{z}$$

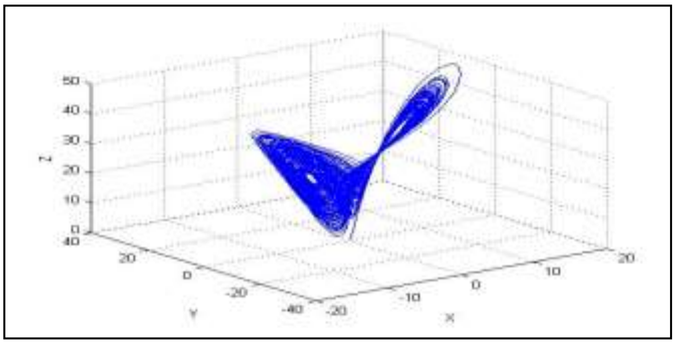


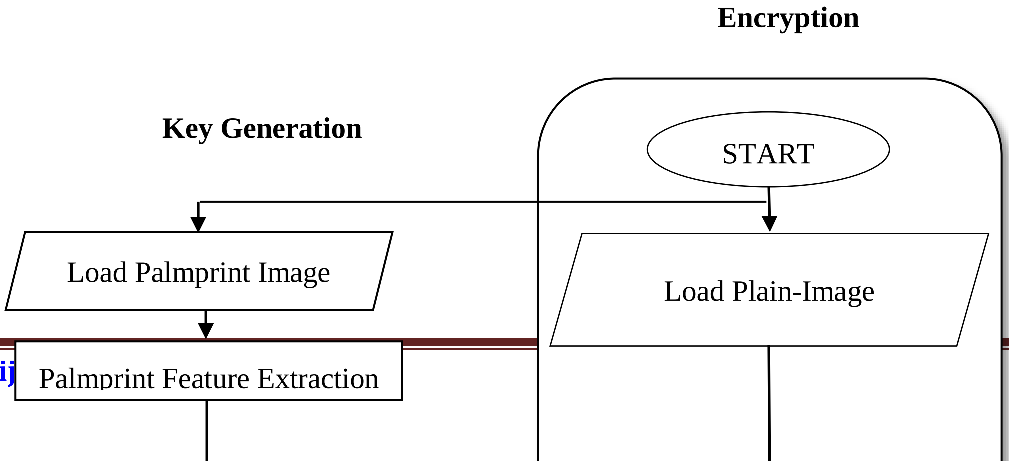
Figure 7.1: Lorenz Chaotic System Orbit

The arbitrary parameters σ, r and $b > 0$ and for this example are σ= 10, r = 28, β= 8/3. This system of differential equations is then solved numerically using Matlab's ode 45 RungeKutta routine. The results are plotted in Figure7.1[20].

8. PROPOSED SYSTEM

In this proposed system, there are two main parts: key generation and encryption process. In the encryption Process, first will read the plain image and resize it according to the proposed system as (256*256*3), this image will be encrypted with the modified AES, the encrypted image with modified AES in which its keys are from block masks that generated from Lorenz system that take its input from palmprint features, then XOR the encrypted image with modified AES with the three masks which it is previously generated by key generation process.

First reading the plain-image that we want to encrypt after that performing resize operation on it to have the size that is used in this proposed system then extract the size of the image (M*N*U) which (M) is the width and (N) is the height and (U) is the number of layers in this proposed system the size is (256*256*3) and (U) is (3) because dealing with RGB image, after that this resized image will be encrypted with the modified AES Figure (7) below.



Keys

Figure 8.1: Proposed Encryption System Flowchart

9.MODIFIED AES

The modified AES is like the normal AES completely except the parts of key and Mix-Columns operation, in the encryptions hand decryptions reprocesses them algorithm will reprocess them imaged as blocks each block is (4*4 block) this block will be passed to 4*4 state array, this state array will be changed at each operation in the encryption and decryption process, and modified AES is one of most important thing in this proposed system which is modified AES.

Modified AES is the known AES but with little change in its algorithm the change is in cancelling Mix-Columns operation and leaving all other operations as it is, the reason behind that (removing Mix-Columns) operation is to reduce the time required in this operation.

And because of that, (a lot of time consuming) this operation eliminated and if this operation is kept in the AES and we are dealing with large amount of data in this proposed system (images), the result will be a lot of time consuming will be more than one minute, and must refer for that the Mix-Columns operation is the operation that takes the largest amount of time in comparison with the other operations.

And in this proposed system when eliminating this operation (Mix-columns) we put new operations instead of it, these operations eliminate the disadvantage of consuming a lot of time and in the same time did not just give us the same security power.

These new operations enhanced the security by giving us more entropy and better correlation compared with the old Mix-Columns operation as we will describe in details in chapter four, these operations are the changing chaotic keys that are generated from Lorenz system used in the modified AES and the XOR operation of the cipher image produced from modified AES with the chaotic masks that also generated from Lorenz system.

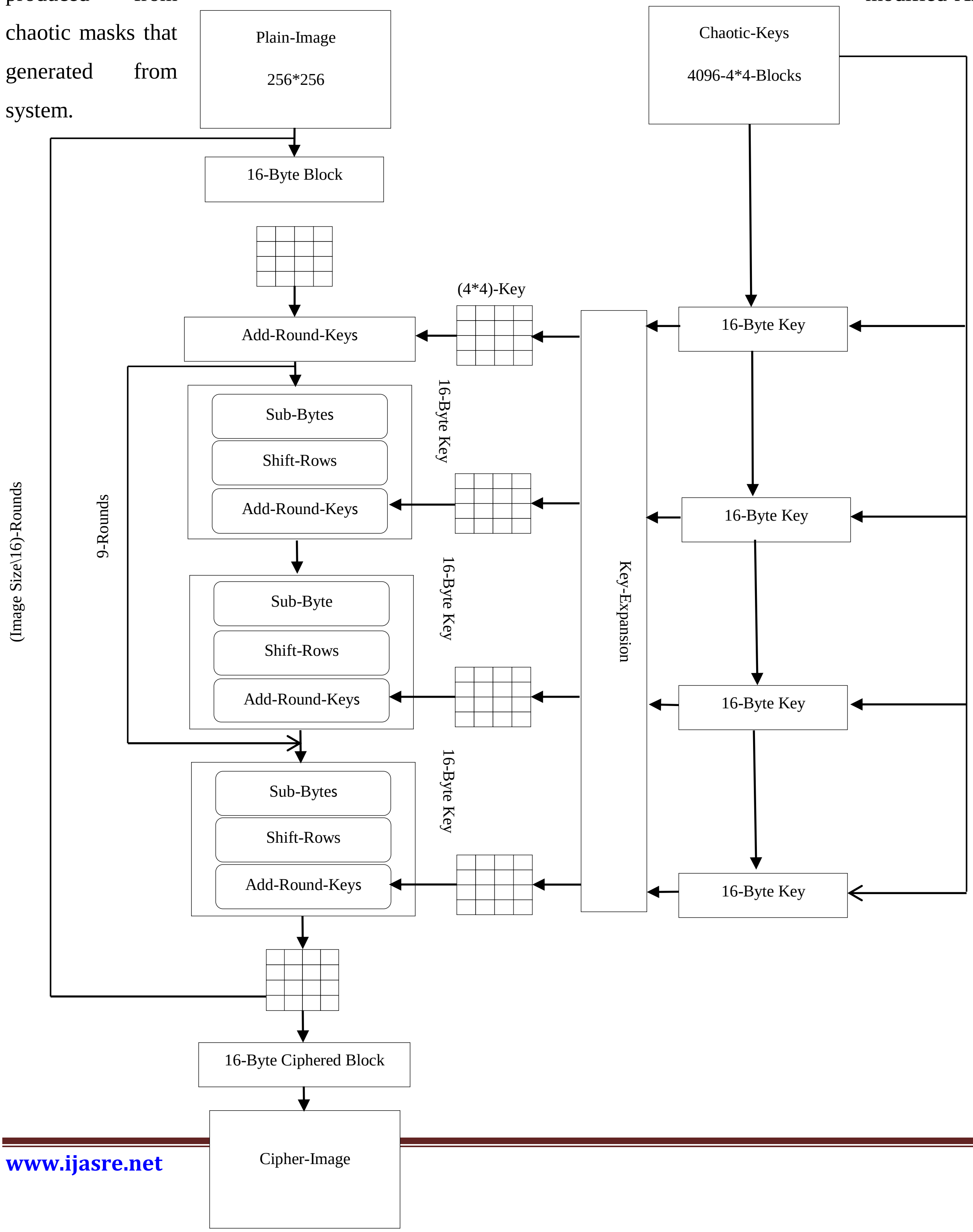


Figure 9.1: Modified AES Encryption Process

10.PALMPRINT MINUTIA (FEATURES) EXTRACTION

The steps of the minutia extraction are described as flowchart below in Figure (10.1).

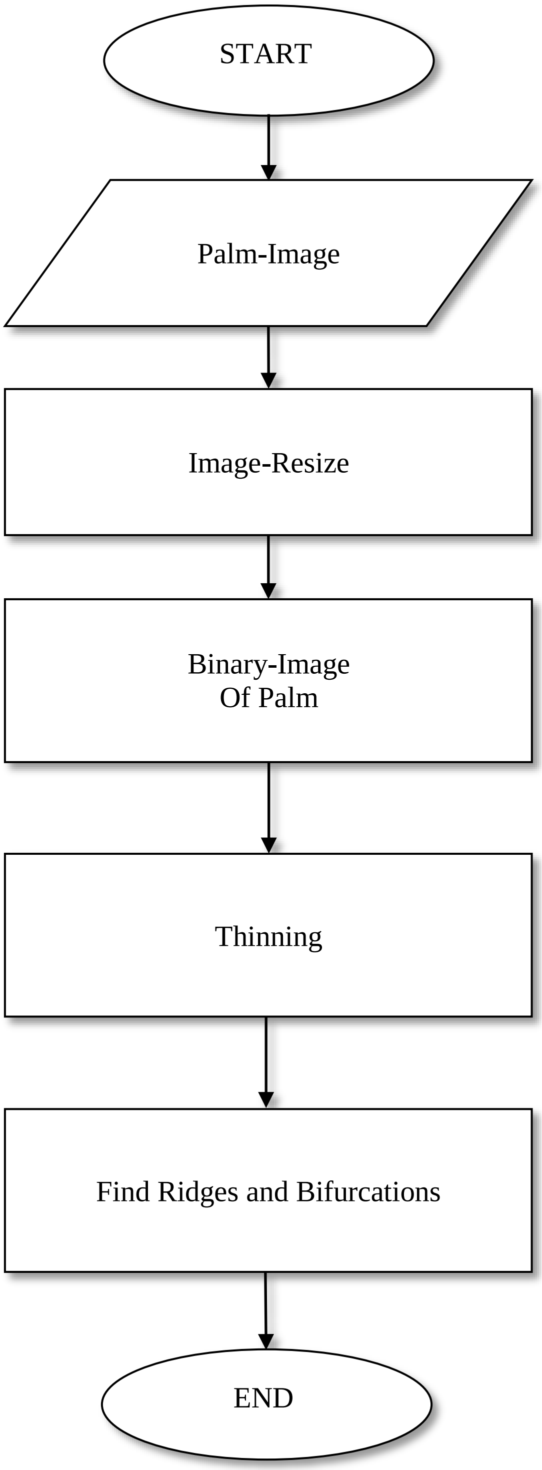


Figure 10.1:Palmprint Minutia Extractions

When reading the palmprint image after that resizing it to get just the most important part of the palmprint image the part that has the details (minutia) that we want to extract, after that this image will be converted to binary image by using binrization which convert the pixel image to binary image to focus only on the important details (minutia) in the palmprint image the pixel pattern that localized as shown in Figure (10.2).

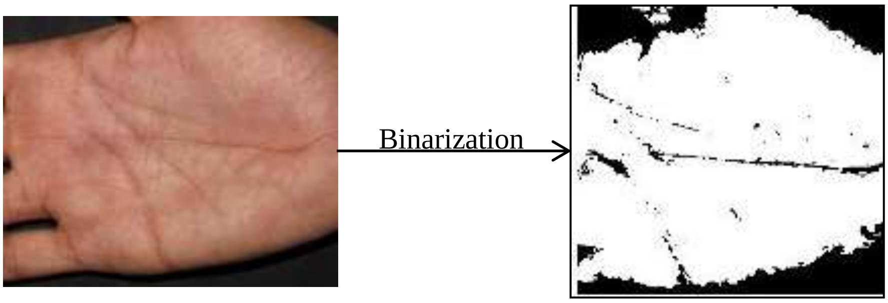


Figure 10.2:Binrization

After that performing thinning operation which is a morphological operation on the binary image to remove the additional pixels of ridges that spread on wide area of the binary image, the result will be thinned ridges as shown in Figure (10.3).

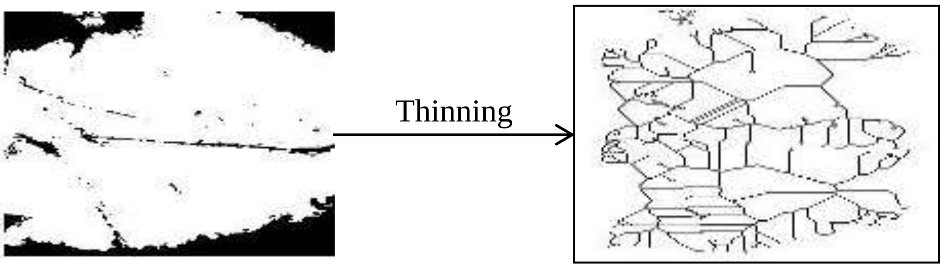


Figure 10.3: Thinning

After the thinning operation will perform the operation that finding the ridges and bifurcations which it is a morphological operation as in Figure (10.4).

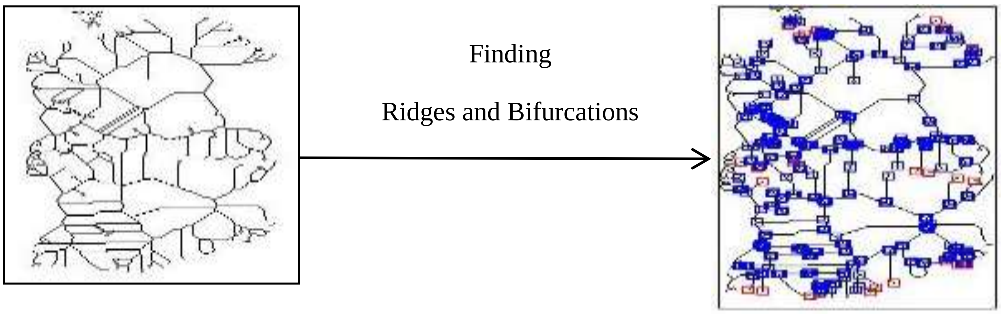


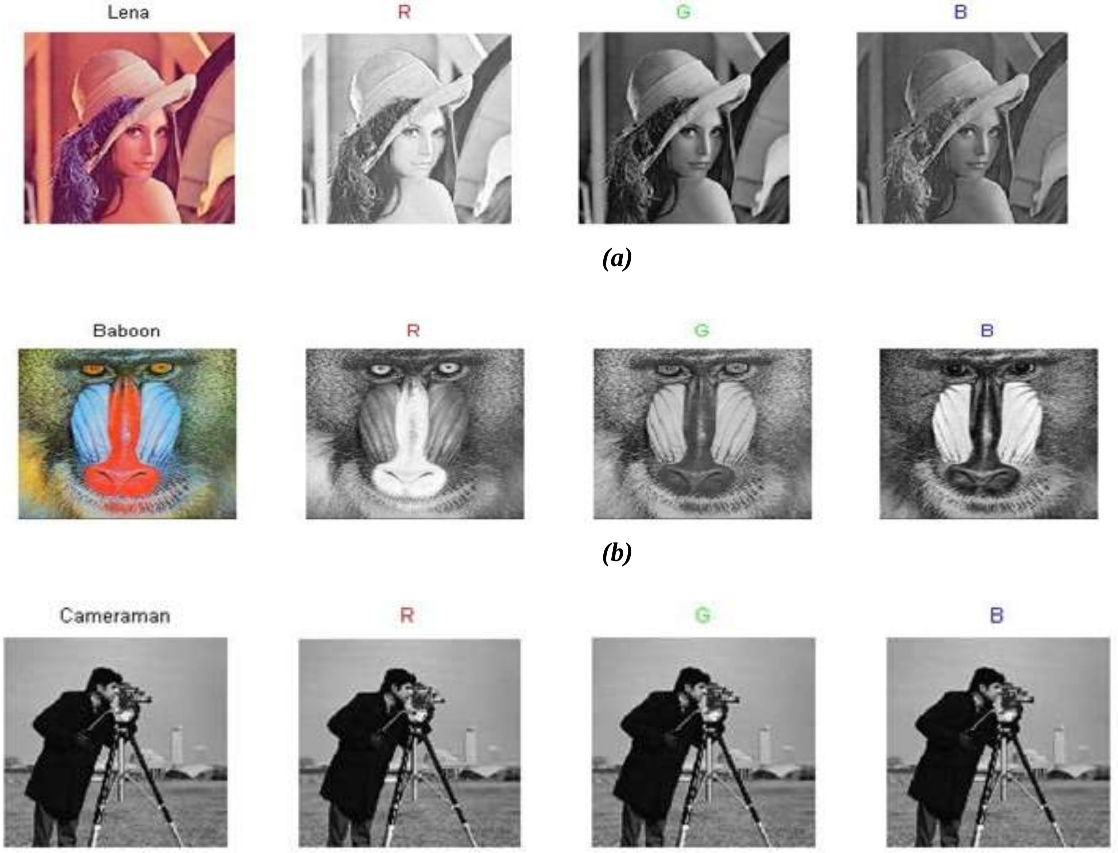
Figure 10.4: Finding Ridges and Bifurcations

11. PROPOSED SYSTEM IMPLEMENTATION PLATFORM

The proposed system is programmed via (Matlab programming language version 8.1.0.604 (R2013a)) and it is performed on computer system that has the following properties:

- Windows-7 Ultimate (32-bit).
- Processors: Intel (R) Cores (TM) i7-3517U CPU @ 1.90GHz (4 CPUs), ~2.4GHz.
- Memory: 4-GB.

And using three samples, which these three samples are images of type (JPEG), these images are of size (256*256) and three color component (R,G,B), and these images are shown in Figure (9) with its color components.



(a)

(b)

(c)

Figure 11.1: (a) Lena Image with its RGB Components (b) Baboon Image with its RGB Components (c)Cameraman Image with its RGB Components

12. EXPERMINTAL RESULTS

A good encryption algorithm (system) must be strong enough to resist to all attacks, statistical and cryptanalytic, such as the key should be large enough to make the brute-force attack infeasible, and the system must be sensitive to the key change even if it is a tiny change, and must has a uniform histogram to avoid the statistical attacks, and has high randomness, and in addition the system should have a low time consuming in both encryption and decryption processes [21].

12.1 Analysis of the Key space

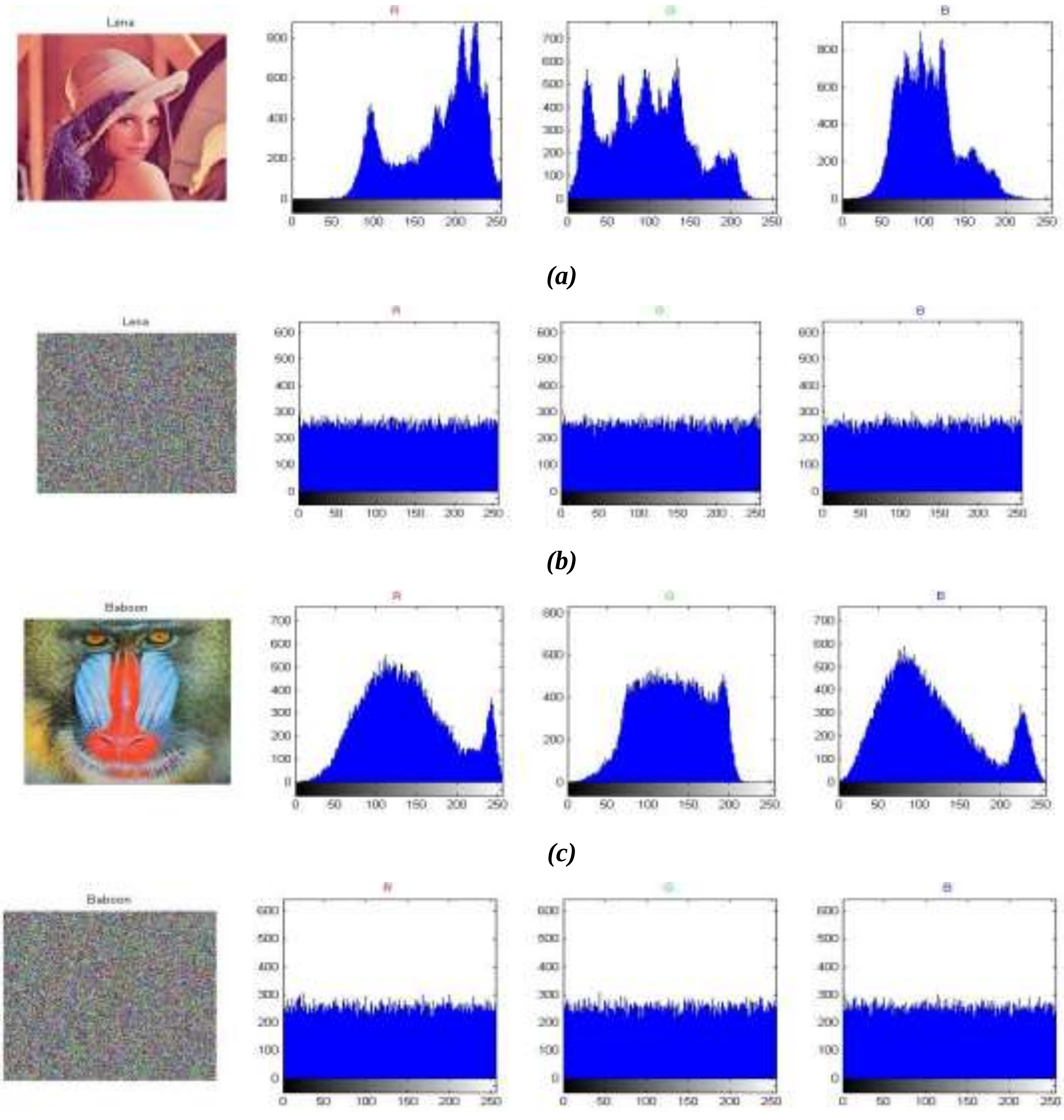
First defining key space size which it is the total number of different keys that can be used in theencryption process. In this proposed system we used the initial conditions as secret key, the key precision is 10^{-15}, then the size of the key space will be 10^{120}, this is so large comparing with 2^{128}, then the keys space is large enough to resist the brute-force attack.

12.2 Analysis of the Key Sensitivity

The proposed system should be sensitive to the change of the encryption key even if it was tiny, for example in the encryption process a tiny change in the encryptions key should give totally different image, for example we encrypted an image with the encryption key (0.87), a tiny change in the key should not decrypt the image, and perform decryption process with the key (0.8700000000000001) the result was the encrypted image with key (0.87) is not decrypted as shown in the Figure (9), so the system is sensitive to the change of the encryption key even if it was a tiny change.

12.3 Them Histograms Analysis

The image histogram show how pixels in an image are distributed by graphing the number of pixels at each color intensity level. The Figure (12.1) showing the histogram of the proposed system for plain and encrypted image.

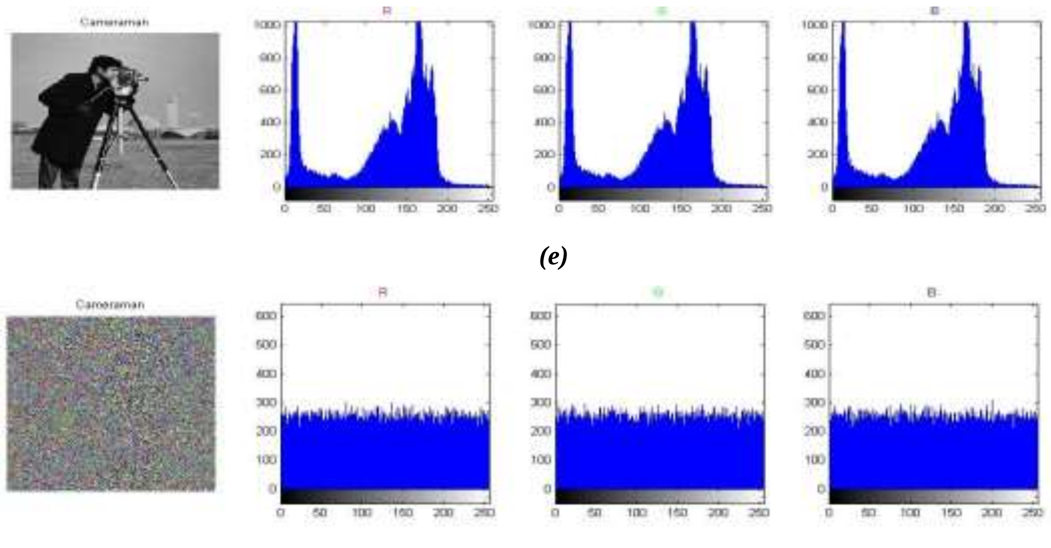


(a)

(b)

(c)

(d)



(e)

(f)

Figure 12.1: (a) Original Lena Image with the Histogram of its RGB Components (b) Encrypted Lena Image by the Proposed System with the Histogram of its RGB Components, (c) Original Baboon Image with the Histogram of its RGB Components (d) Encrypted Baboon Image by the Proposed System with the Histogram of its RGB Components with the Histogram of its RGB Components, (e) Original Cameraman Image with the Histogram of its RGB Components

(f) Encrypted Cameraman Image by the Proposed System with the Histogram of its RGB Components with the Histogram of its RGB Components

12.4 Correlation Analysis

In this operation will compute the correlation of the three color components (R, G, B) between the plain image and the encrypted image,and this will compare between the original AES and the proposed system as shown in the Table (12.1) and the correlation equation (1) is as follows:

$$r = \frac{\sum_m \sum_n (A_{mn}-\bar{A})(B_{mn}-\bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn}-\bar{A})^2)(\sum_m \sum_n (B_{mn}-\bar{B})^2)}} \text{-----------------(1)}$$

Table12.1: Correlation Analysis for the Three Samples and their Components

	Encryption Algorithms					
Images	Proposed System			Original AES		
	R	G	B	R	G	B
Lena	0.0044	-0.0052	0.0017	-0.01023	-8.3520e-04	-8.3509e-04
Baboon	-0.0036	-0.0078	0.0091	0.0039	0.0094	-0.0094
Cameraman	0.0081	5.0258e-04	0.0036	-0.0026	-0.0026	-0.0026

Where r is the correlation and A is the original image and B is the encrypted image and $\bar{A}$ is the mean of the original image and $\bar{B}$ is the mean of the encrypted image and m is the width of the image and n is the height of the image for both, the correlation is referred to the association of the pixels between the original image and encrypted image, and the correlation value is between (-1 and 1) and the good encryption should have a correlation value near to zero.

Table (12.1) above illustrates that the proposed system has correlation value near to zero and better than original AES.

Table 12.2: PSNR values for the Three Samples and their Components

Images	Encryption Algorithms					
	Proposed System			Original AES		
	R	G	B	R	G	B
Lena	7.83750	8.57848	9.58938	7.81947	8.59777	9.58234
Baboon	8.88995	9.47391	8.61959	8.92651	9.50899	8.54958
Cameraman	8.41278	8.38506	8.42141	8.38819	8.38819	8.38819

12.5 PSNR Analysis

PSNR reflects the encryption quality (3). Mean square error (MSE) is the cumulative squared error between original and decrypted image (2).

$MSE = \frac{1}{M\times N}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j)-K(i,j)]^2$----------------(2)

$PSNR = 10 \times \log_{10}\left(\frac{MAX_I^2}{MSE}\right)$----------------------(3)

Where the I (i,j) refers to the plain image and K(i,j) is the encrypted image and M is the width and N is the height of the image.

The Table above illustrates that the proposed system has near or better PSNR value than the original AES, and the following Figures (12.2, 12.3 and 12.4) give a better illustration.

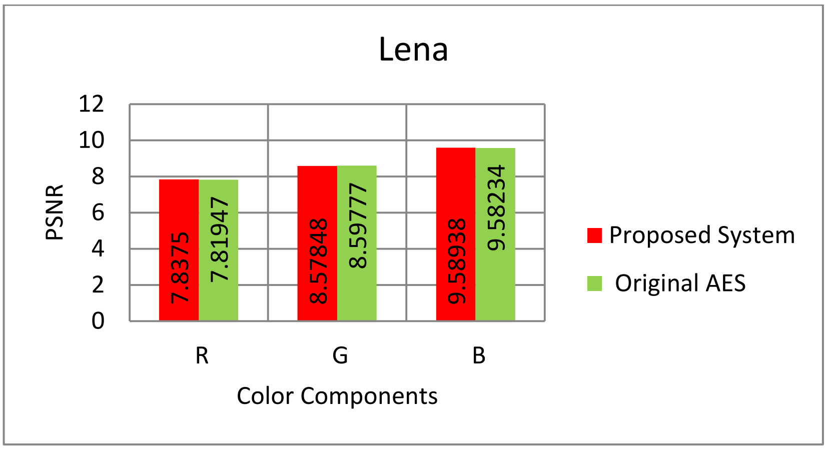


Figure 12.2: PSNR for Lena Encrypted Image

Figure (12.2) above shows that the proposed system and the Original AES in the image (Lena), has a PSNR value that is nearly the same.

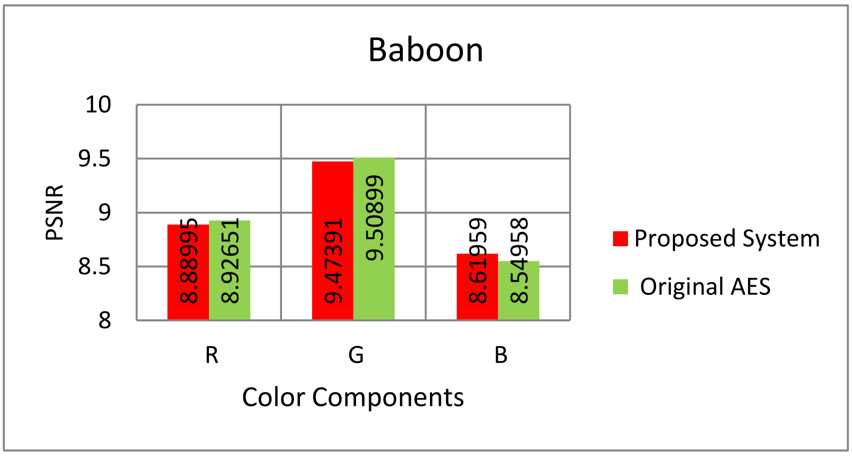


Figure 12.3: PSNR for Baboon Encrypted Image

Figure (12.3) above shows that the proposed system and the Original AES in the image (Baboon), has a PSNR value that in some cases better for AES and in some better for the proposed system.

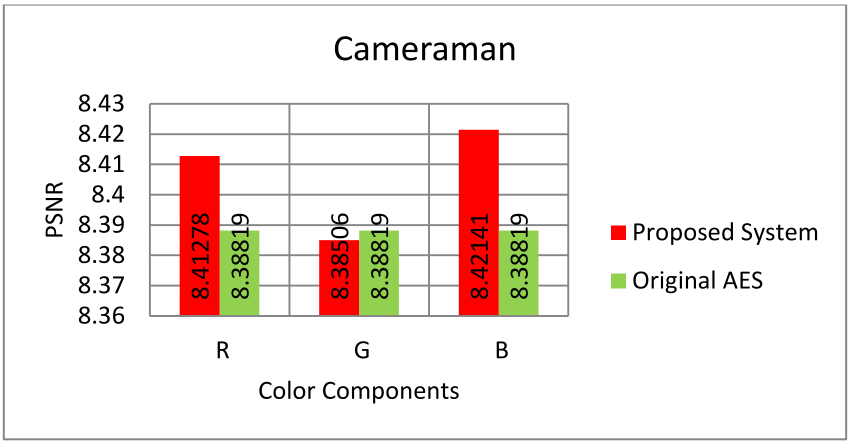


Figure 12.4: PSNR for Cameraman Encrypted Image

Figure (12.4) shows that the proposed system and the Original AES in the image (Cameraman), has a PSNR value that is better for the proposed system than AES in all cases (color components).

12.6 NPCR and UACI Analysis

The NPCR and UACI measure test the different range between two images, NPCR means them numbers off unchanging pixels bitrate tin them encryptions reprocess (4), hand them UACI them unified as averaged that unchanged in intensity hare two almost commonly tests that's are fused tom evaluates them performance off them algorithms that are fused tin imaged (5). Their NPCR hand UACI measures tests them differential ranges between two images.

$$NPCR = \frac{\sum_{ii,jj} D(ii,j)}{M \times N} \times 100\%\text{---------------}(4)$$

$$UACI = \frac{1}{M \times N}\left[\frac{\sum_{ii,jj} C(i,j) - C\prime(ii,j)}{255}\right] \times 100\%\text{-----------------}(5)$$

So, the NPCR and UACI calculate the encryption algorithm sensitivity to them unchanged tin them complain imaged, which mean if as unchanged tin tone pixels tin them complain imaged even if it was tiny in one bit this will lead to totally different decrypted image, this means the proposed system is robust against differential attack. The Table (12.3) below displaying the result that gained for the NPCR and UACI from both the proposed system and original AES, and the results denote that they are nearly the same.

Table . 12.3: The NPCRR and UACI values for the Three Samples and their Components

Image	Statistics	Encryption Algorithms					
		Proposed System			Original AES		
		R	G	B	R	G	B
Lena	NPCR	99.6643	99.5849	99.5956	99.6170	99.5575	99.5742
	UACI	33.1457	30.5538	27.6738	33.1915	30.4897	27.6884
Baboon	NPCR	99.6368	99.6078	99.5773	99.6032	99.5956	99.6521
	UACI	29.6556	28.0338	30.3882	29.4851	27.8510	30.6863
Camera Man	NPCR	99.5789	99.6307	99.6185	99.6093	99.6093	99.6093
	UACI	31.0579	31.1404	31.1180	31.1837	31.1837	31.1837

And the following Figures (12.5, 12.6 and 12.7) give more details.

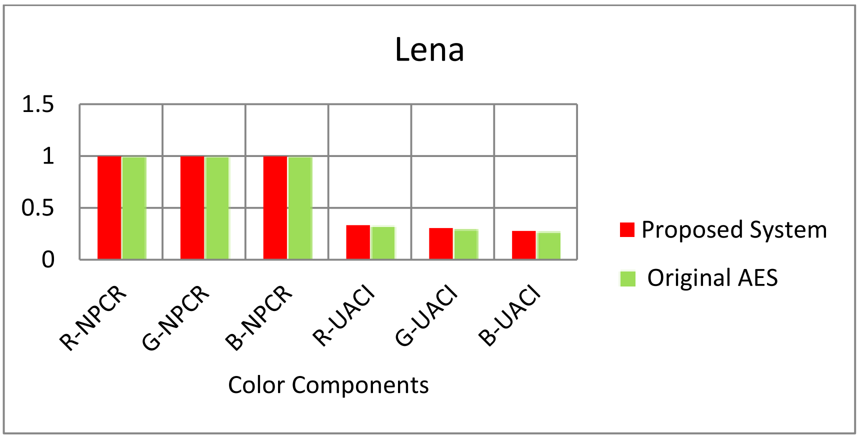


Figure 12.5: NPCR and UACI form Lena Encrypted Image

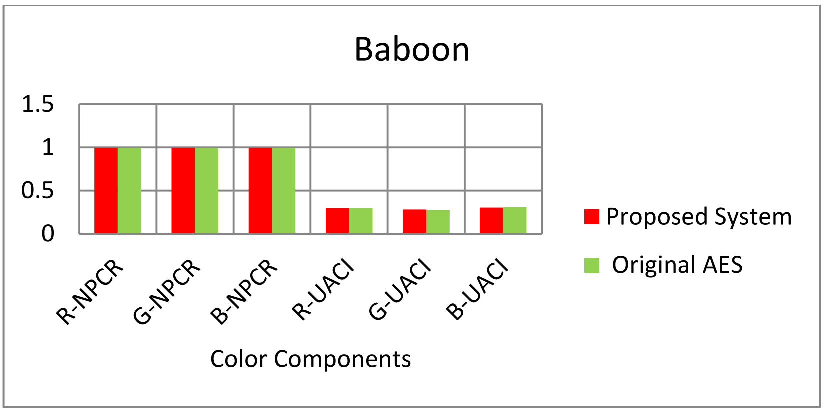


Figure 12.6: NPCR and UACI for Baboon Encrypted Image

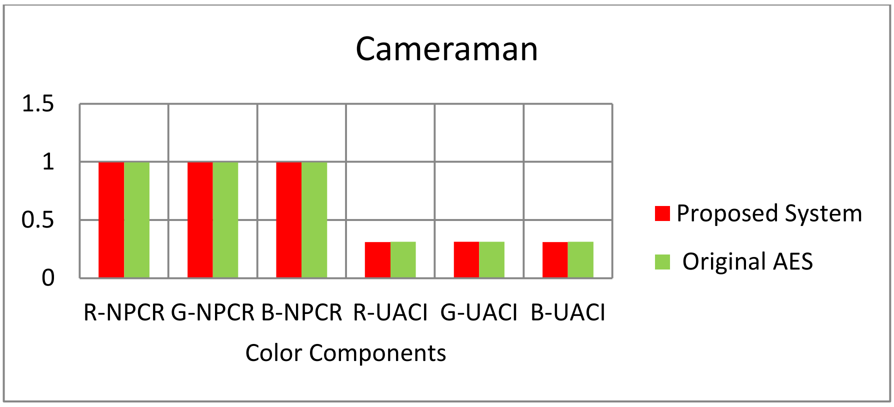


Figure 12.7: NPCR and UACI for Cameraman Encrypted Image

12.7 Entropy Analysis

The entropy calculate the uncertainty association of the random values, the good encryption algorithm should give a low mutual information of the pixel values of the encrypted image, and this means that the entropy will be increased, and the entropy equation is as follows in the (6):

$$Entropy\boldsymbol{y} = -\sum_{ii=\mathbf{0}0}^{\mathbf{0}255} p(x_i) \times \log_2 p(x_i)\text{---------------------(6)}$$

Where$p(x_i)$ the probability of each occurrence is in the encrypted image, if the probability of each pixel occurrence is equal the entropy value will be 8 this means complete entropy, and the expected value should be 8 or near to it.

Table 12.4: Entropy values for the Three Samples

Images	Proposed System	Original AES
Lena	**7.9990**	**7.9990**
Baboon	**7.9991**	**7.9991**
Cameraman	**7.9991**	**7.9965**

Table (12.4) above illustrates that the proposed system scored an entropy better than the original AES, especially in the sample (the image) of Cameraman, this is another advantage that the proposed system gave and the Figure (12.8) bellow illustrate the Table above.

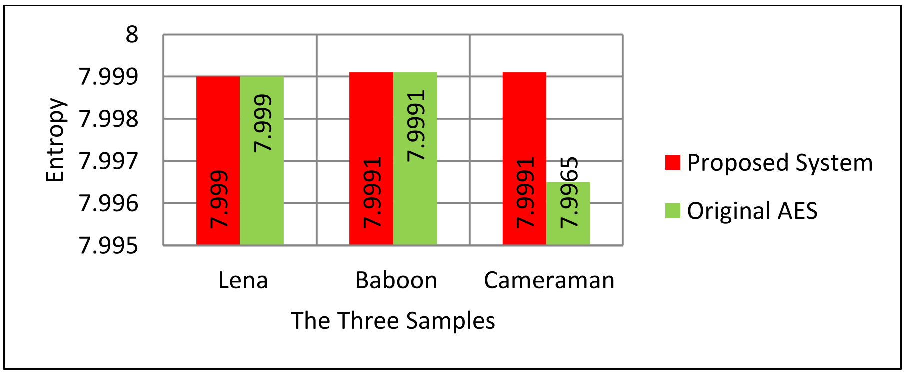


Figure 12.8: Entropy values for the Three Samples

12.8 The Execution Time

The execution time is one of the most important measures of encryption and decryption processes that is improved, the execution time for the encryption and decryption processes of the original AES is about (86.5342) seconds, and the execution time of the encryption and decryption processes is about (9) seconds, and this is a huge difference between the original AES and the proposed system and this is another important advantage to the proposed system.

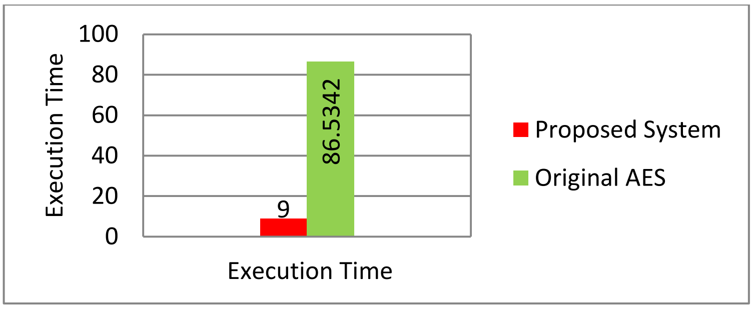


Figure 12.9: Execution (Encryption / Decryption) Time for both Proposed Systems and AES

ACKNOWLEDGMENT

Thanks and praise to Allah, and prayer and peace be upon his messenger Mohammed, and his own pure relatives and best companions.

First, I would like to thank all my family for their help and support. And my sincere gratitude goes to my supervisor, **Assist.Professor Dr. Haider K. Hoomod**, who has guided and supported me continuously, this thesis owes much to his thoughtful and helpful advice.

REFERENCES

[1] Abdullah SharafAlghamdi and HanifUllah.,"A Secure Iris Image Encryption Technique using Bio-Chaotic Algorithm", (IJCNS) International Journal of Computer and Network Security, Vol.2, No.4, April 2010.

[2] P.Radhadevi and P.Kalpana., "Secure Image Encryption Using AES" IJRET: International Journal of Research in Engineering and Technology ISSN: 2319-1163.Vol. 01. October 2012.

[3] Ruisong Ye and WeichuangGuo., "A Chaos-based Image Encryption Scheme Using Multimodal Skew Tent Maps", Journal of Emerging Trends in Computing and Information Sciences ,Vol. 4, No.10, ,ISSN 2079-8407, October 2013.

[4] HaojiangGao et al., "A New Chaotic Algorithm for Image Encryption" Chaos, Solutions and Fractals 393–399, 29 (2006).

[5] BhavanaAgrawal and HimaniAgrawal., "Survey Report on Chaos Based Cryptography" Vol.2, Issue 2 (February 2012).

[6] Salim M. Wadi, NasharuddinZainal, " A Low Cost Implimantaion of Modified Advanced Encryption Standard Algorithm using 8085A Microprocessor", Journal of Engineering Science and Technology Vol. 8, No. 4 (2013) 406 – 415, School of Engineering, Taylor's University.

[7] Ajish S, "Wavelet Based Advanced Encryption Standard Algorithm for Image Encryption", International Journal of Engineering Research and General Science Volume 3, Issue 1, ISSN 2091-2730, January-February, 2015.

[8] ChittaranjanPradhan, Ajay Kumar Bisoi, " ChaoticVariions of AES Algorithm", International Journal of Chaos, Control, Modelling and Simulation (IJCCMS) Vol.2, No.2, June 2013.

[9] Colin Soutar et.al., "Biometric Encryption", Bioscrypt Inc. (formerly Mytec Technologies Inc., Explorer Drive, Suite 500, Mississauga, ONT.

[10] Anil K. Jain et al., "Handbook of Biometrics ", Springer Science 2008, ISBN-13: 978-0-387-71040-2, p3.

[11] Amioy Kumar and Ajay Kumar., " A Palmprint Based Cryptosystem using Double Encryption", Biometrics Research Laboratory, Department of Electrical Engineering, Indian Institute of Technology Delhi HauzKhas, New Delhi 110 016, India, 17 March 2008.

[12] K.Y. Rajput et al., "Palmprint Recognition Using Image Processing", ISSN 0974-3375, TECHNIA–International Journal of Computing Science and Communication Technologies, Vol. 3, NO. 2, Jan. 2011.

[13] Ted Dunstone and Neil Yager., "Biometric System and Data Analysis Design, Evaluation, and Data Mining", Springer 2009, ISBN-13: 978-0-387-77625-5.

[14] Sumalatha K.A and Harsha H., " Biometric Palmprint Recognition System: A Review", International Journal of Advanced Research in Computer Science and Software Engineering, Vol.4, Issue 1, January 2014.

[15] Joan Daemen and Vincent Rijmen., "The Design of Rijndael: AES – The Advanced Encryption Standard", Springer, 2002, ISBN 3-540-42580-2, p1.

[16] "Announcing the Advanced Encryption Standard (AES)", Federal Information Processing Standards Publication 197, United States National Institute of Standards and Technology (NIST), November 26, 2001, p5.

[17] William Stalling., " Cryptography and Network Security: Principles and Practices, Principles and Practices ", 5th ed. Prentice Hall, 2011, p150-p151.

[18] Joan Daemen, Vincent Rijmen, " AES Proposal: Rijndael", Document version 2, Date: 03/09/1999.

[19] Hyubgun Lee et el. "AES Implementation and Performance Evaluation, on 8-bit Microcontrollers", Department of Computing, Soongsil University, Seoul, South Korea, (IJCSIS) International Journal of Computer Science and Information Security,Vol. 6 No. 1, 2009.

[20] Qais H. et al., "Image Encryption Based on the General Approach for Multiple Chaotic Systems", journal of Signal and Information Processing, 2, 238-244, 2011.

[21] N.F.Elabady et al., "Image Encryption Based on New One-Dimensional Chaotic Map", International Journal of Computer Applications, Vol.108, No.19, 2014.