

An FPGA Implementation of Chaos based Image Encryption and its Performance Analysis

¹ Bikash Baruah, ² Monjul Saikia

¹ Research Scholar, Department of CSE
NERIST, Nirjuli Arunachal Pradesh

² Assistant Profesor, Department of CSE
NERIST, Nirjuli Arunachal Pradesh

Abstract - Today, hardware chip design with FPGA implementation for designing secure crypto processor is a growing topic due to rapidly increasing attack on digital images over internet network. In this paper, an FPGA implementation of Chaotic Map based two phase image encryption technique is proposed. First phase consists of pixel position permutation and second phase consists of bit value position permutation among different bit planes. In the first phase, original pixel values remain unchanged and in second phase, though pixel values are not directly changed, but finally due to position permutation over bit-planes, values are also modified. These permutations in first and second phase are done by using chaotic behaviour of Arnold Cat Map and Logistic Map respectively. A complete analysis on robustness of the method is shown. Correlation, Encryption time, Decryption time and key sensitivity show that the proposed crypto processor offers high security and reliable encryption speed for real-time image encryption and transmission.

Key words – Chaos, Image Encryption, Crypto-processor, FPGA, Logistic Map, Cat Map.

1. Introduction

Crypto processor [1] is a specialized processor (consists of accelerated encryption, protection against tampering, protection of data and Secure I/O) that executes cryptographic algorithms within hardware. The demand of secured and less expensive crypto processor is being increased day by day to resist the attack of third parties, which is also increased rapidly with the development of internet technology [7][8]. Figure 1 shows the general working model of a crypto processor.

In this paper, two different Chaotic Maps i.e., Arnold Cat Map [10] and Logistic Map [12][13] are implemented through FPGA (Field Programmable Gate Array) [1][14][15] to design a real-time crypto processor.

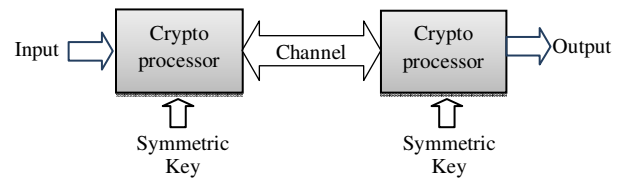


Fig 1: Crypto-processor model

Chaos theory [2][3][5] is the field of study in mathematics that studies the behaviour of dynamical systems that are highly sensitive to initial conditions. In mathematics, a chaotic map is a map (= evolution function) that exhibits some sort of chaotic behaviour. The important characteristics of a Chaos are as follows:

- it must be sensitive to initial conditions
- it must be topologically mixing
- it must have dense periodic orbits

It can be said that any function that fulfils the above mentioned behaviour are called a chaotic function [4]. The map or the graph obtained by plotting the values which is again found by infinite iteration of that function is called Chaotic Map for that function.

Arnold Cat Map was introduced by Arnold and Avez [2]. Its working formula for NxN matrix and geometric representation are as follows:

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & a \\ b & ab + 1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \text{mod } N = A \begin{bmatrix} x_n \\ y_n \end{bmatrix} \text{mod } N \quad (\text{eq.1})$$

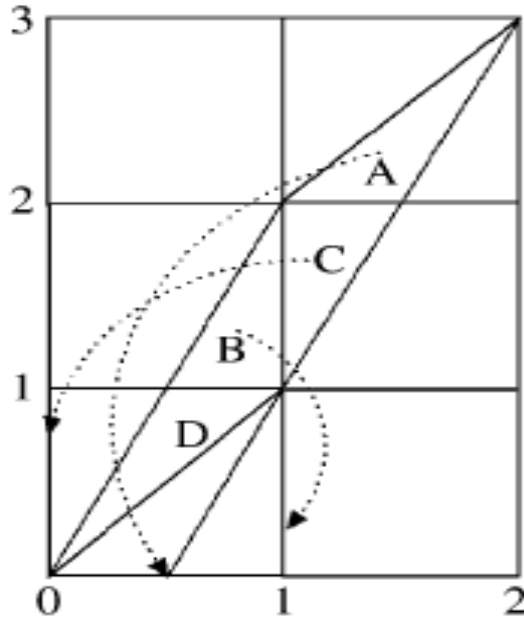


Fig. 2: Geometrical explanation of Arnold cat map

Logistic mapping was originally proposed by P. Verhulst in 1845, but has become widely known through the work by R. May. It is the simplest among all the chaotic maps.

$$\begin{aligned} x_{n+1} &= rx_n(1 - x_n) \\ y_{n+1} &= rx_n(1 - y_n) \\ z_{n+1} &= rx_n(1 - z_n) \end{aligned} \quad (\text{eq.2})$$

Here, r is the constant and x_n, y_n, z_n are the state value, and combination of x_n, y_n, z_n will be the Encryption key.

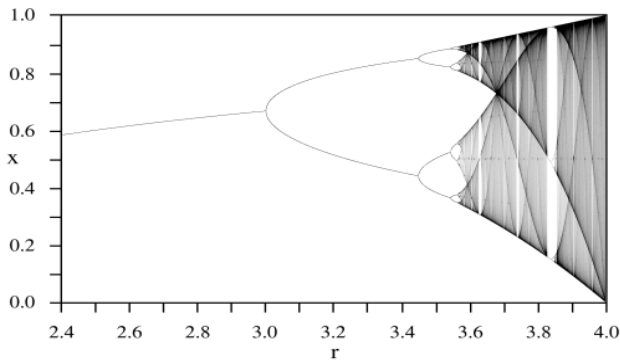


Fig. 3: Chaotic Logistic Map

2. Proposed Method

The proposed method is a two phase image encryption technique, implemented in spatial domain and utilizes an

Encryption Key of 48-bit long. First phase is named as Confusion and its output image is called Confused Image and second phase is named as Diffusion Phase and the output of diffusion phase gives diffused image or final encrypted image.

In this paper, complete and partial image encryption techniques are combined with the concept of pixel position permutation and bit-plane permutation [5][9]. To shuffle the pixels (Confusion phase) and bit-planes (Diffusion Phase) Arnold Cat Map and Logistic Map are used respectively.

2.1 Encryption

Figure 4 describes the overall flow diagram for Encryption process. Initially, in Confusion phase (x_0, y_0) is the first coordinate i.e., $(0, 0)$ position of the original image and (X_{out}, Y_{out}) is the new coordinate obtained by using Arnold Cat Map. Pixel values of the original image in these positions are exchanged in Confusion phase. For every pixels separate (X_{out}, Y_{out}) is calculated using Arnold Cat Map and exchanged with new coordinate. Once, completion of this process will give us the Confused Image.

If the Correlation between Original Image and Confused Image seems to be greater than the expected, Confusion phase can be repeated up to infinite times by considering the first confused image is as the input for second, second confused image is as the input for third and go on.

In our proposed scheme, we have iterated two times by considering an efficient encryption time and an effective correlation value between original and confused image. Figure 5 shows the images after first iteration and second iteration of an image (cameraman.tiff). In Confusion phase histogram of the original image and confused image are remain same; because only pixel value positions are interchanged but, pixel values are still same with the original image. There is a high probability to decrypt the confused image by histogram analysis attack.

Hence output image after second iteration is completely different from the original image, but it cannot be considered as final encrypted image. So, one more phase has to be added, named as Diffusion Phase.

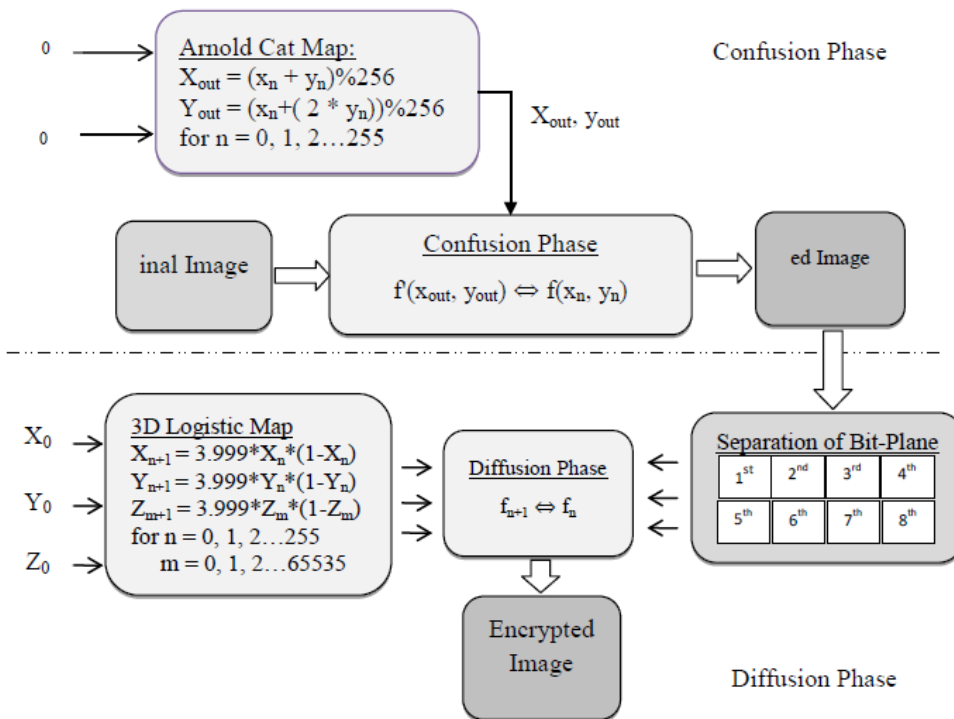


Fig 4: Encryption Process

In Diffusion phase gray scale confused image is divided into eight different bit-planes [6] from MSB to LSB and 3D Logistic Map [2][11][13] is implemented to shuffle the bits among the bit-planes of MSB onwards. Figure 7 clearly shows that the bit-planes of MSB onwards contains more information about the original image (shown in figure 6) compared to the bit-planes of LSB onwards. Since, lower bit-planes contain almost negligible information about the original image; hence in our proposed scheme we are going to implement 3D Logistic Map only in higher bit-planes.

To implement 3D Logistic Map, 48-bit key is divided into three 16-bit subparts and assigned as (X_0, Y_0, Z_0) to calculate $(X_{n+1}, Y_{n+1}, Z_{m+1})$. Now 3D logistic map will be implemented one by one on every bit of different bit-planes and a unique (X, Y, Z) value has to be generated every time for each and every bits.

There are total 8 numbers of bit-planes from 0, 1, 2 ... 7 (MSB to LSB) and every bit-plane consists of 65536 numbers of individual bits and its positions in 2D coordinate are $(0, 0), (0, 1) \dots (255, 255)$.

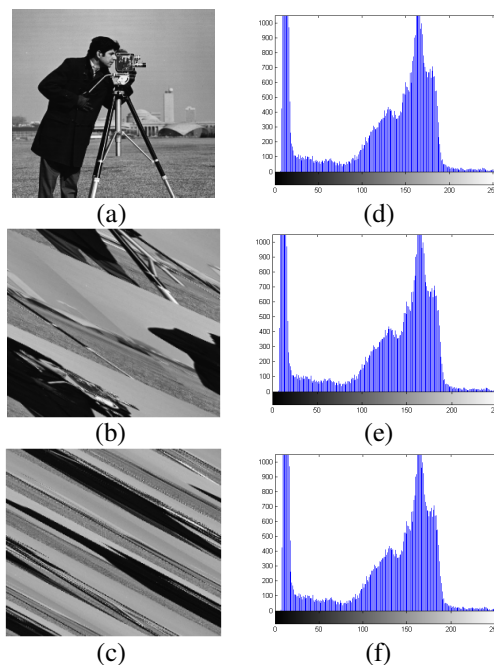


Fig. 5: (a) Original Image; (b), (c) are confused image after 1st and 2nd iteration respectively; (d), (e), (f) are histogram of image (a), (b), (c) respectively



Fig. 6: Bit Plane of the image (a) - (h) is the bit-planes from highest bit to the lowest respectively

In the block diagram shown in figure 4, Diffusion Phase takes two different inputs; one from 3D Logistic Map which gives (X, Y, Z) values and another from Separation of Bit-Plane block which provides separated bit-planes of confused image.

To explain how bit-planes are being shuffled using 3D logistic map in Diffusion Phase, let us consider a single bit, positioned at (0, 0) of 0th bit-plane i.e., the 1st bit of MSB plane. Here, Z value decides the bit-plane with which that single bit will be shuffled and once the bit-plane is decided, (X, Y) gives the exact coordinate to be interchanged. Then, this process will be continued until all the bits of a single bit-plane are completed and then same process will be done for next bit-plane and going on. In our proposed scheme we get an effective result by shuffling three MSB planes in terms running time, correlation among pixels and histogram analysis. Once,

shuffling is completed, the new bit-planes are combined to obtain the final Encrypted Image.

2.2 Decryption

Since Decryption is the reverse process of Encryption, hence for decryption diffusion phase has to implemented first followed by confusion phase. Firstly encrypted image has to be separated into eight different bit-planes. In the diffusion phase shuffled bit-planes has to be reshuffled using 3D Logistic Map to retrieve the confused image. In our proposed encryption scheme we have implemented 3D logistic map into three higher bit-planes from first bit-plane to third bit-plane to shuffle the bits. So, during decryption we have to reshuffle bits starting from third higher bit-plane to first higher bit-plane and bits positioned starting from (255, 255) to (0, 0). Once it is being completed, then combined bit-planes will give the confused Image.

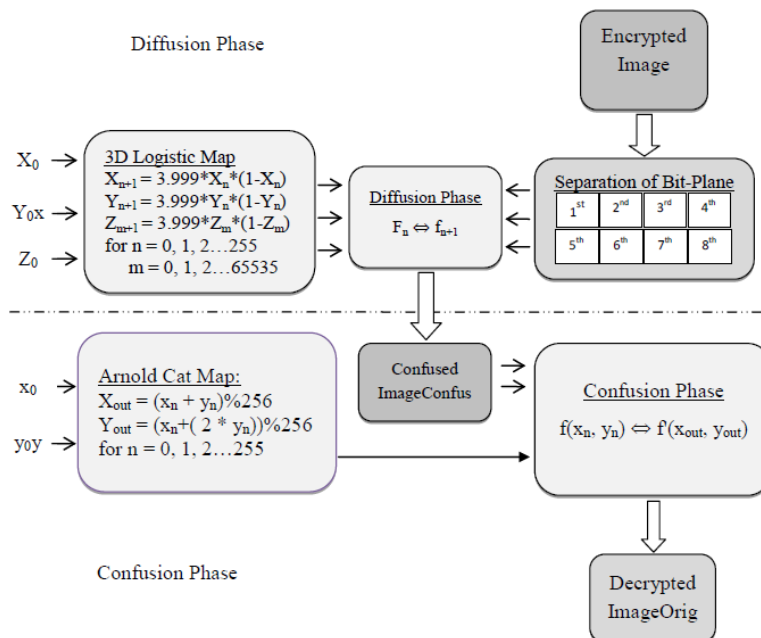


Fig 7: Decryption Process

In Confusion Phase Arnold Cat Map is implemented on confused image to reordering the pixels one by one to retrieve the final decrypted image. Finally, two times

reordering of confused image pixels using Arnold Cat Map will give the expected Decrypted Image.

3. Result and Analysis

In this section we have shown the encryption/decryption results using different keys as well as different number of iterations. Here experiment is done on image size of 256x256 and. We have also studied the correlation between the original images and encrypted images, original images and decrypted images and also between original and incorrectly decrypted using the equation. Higher correlation indicates high similarities between the images and lower the correlation coefficient indicates low similarities between the images. Original image should have low correlation with the encrypted images and correlation value one with the decrypted images. Table 1 to 5 represents the various experimental results such as correlation, encryption/decryption time for an image size of 256x256.

The correlation coefficients can be calculated as follows

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \bar{A})^2)(\sum_m \sum_n (B_{mn} - \bar{B})^2)}} \quad (\text{eq.3})$$

Here A and B are two images, where $\bar{A} = \text{mean2}(A)$ and $\bar{B} = \text{mean2}(B)$.

Table 1: Simulation Result for Confused Image

No. of Iteration	Correlation
1	0.0518
2	-0.0053
3	0.0153
4	0.0021
5	-0.0026
6	-3.2178e-04
7	6.0959e-04
8	-2.7057e-04
9	0.0024
10	-8.0466e-04

Table 2: Simulation result for 2 Bit-Plane

No of planes MSB onward	No of Iteration	Correlation between Original and Encrypted	Correlation between Original and Decrypted	Correlation between Original and Incorrectly Decrypted	Encryption Time (ms)	Decryption Time (ms)
2	1	0.0094	1	0.2295	460	466
2	2	0.0075	1	0.1492	462	469
2	3	0.0017	1	0.1329	470	472
2	4	-4.6996e-04	1	0.1274	480	480
2	5	2.0007e-04	1	0.1261	483	484
2	6	0.0046	1	0.1259	485	493
2	7	-0.0017	1	0.1283	492	499
2	8	0.0019	1	0.1196	499	503
2	9	0.0077	1	0.1221	493	507
2	10	0.0049	1	0.1224	507	517

Table 3: Simulation result for 3 Bit-Plane

No of planes MSB onwards	No of Iteration	Correlation between Original and Encrypted	Correlation between Original and Decrypted	Correlation between Original and Incorrectly Decrypted	Encryption Time (ms)	Decryption Time (ms)
3	1	-0.0059	1	0.1688	510	514
3	2	0.0011	1	0.0882	513	520
3	3	-0.0017	1	0.0450	517	526
3	4	0.0019	1	0.0263	520	528
3	5	0.0034	1	0.0191	521	530
3	6	0.0041	1	0.0090	530	536
3	7	0.0024	1	0.0155	535	541
3	8	0.0026	1	0.0176	539	547
3	9	0.0028	1	0.0101	543	552
3	10	0.0013	1	0.0060	546	555

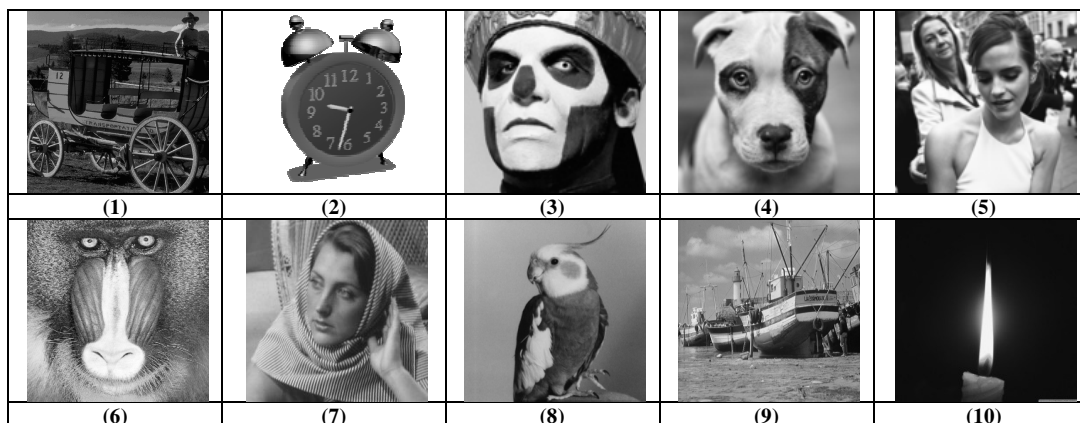
Table 4: Simulation result for 4 Bit-Plane

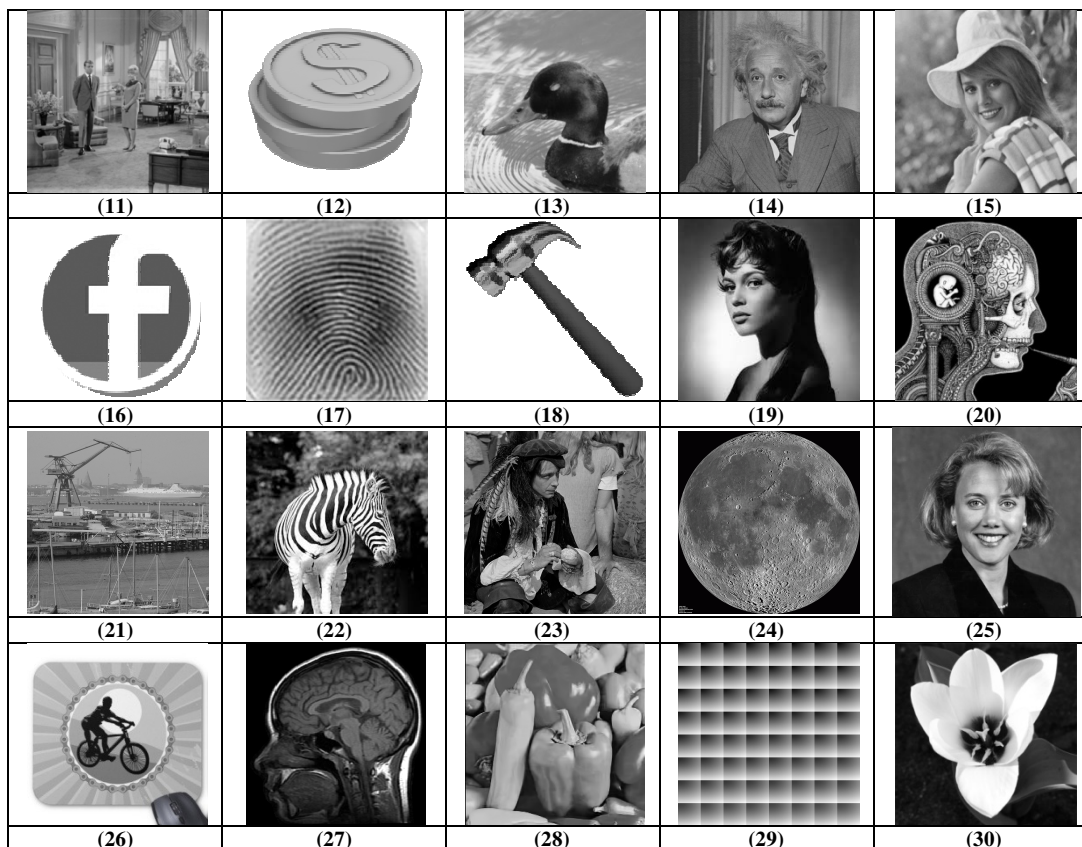
No of planes MSB onwards	No of Iteration	Correlation between Original and Encrypted	Correlation between Original and Decrypted	Correlation between Original and Incorrectly Decrypted	Encryption Time (ms)	Decryption Time (ms)
4	1	0.0016	1	0.2177	525	540
4	2	0.0018	1	0.1344	532	544
4	3	0.0106	1	0.0648	536	560
4	4	0.0035	1	0.0326	541	562
4	5	0.0025	1	0.0060	546	567
4	6	-0.0046	1	0.0080	560	570
4	7	-0.0011	1	8.0558e-04	561	577
4	8	0.0094	1	-0.0020	565	580
4	9	-0.0039	1	-0.0040	570	585
4	10	-0.0020	1	-0.0024	580	590

Table 5: Simulation result for 5 Bit-Plane

No of planes MSB onwards	No of Iteration	Correlation between Original and Encrypted	Correlation between Original and Decrypted	Correlation between Original and Incorrectly Decrypted	Encryption Time (ms)	Decryption Time (ms)
5	1	-0.0016	1	0.2086	605	587
5	2	-0.0074	1	0.1467	611	590
5	3	-0.0079	1	0.0454	619	597
5	4	0.0135	1	0.0605	623	601
5	5	0.0030	1	0.0200	629	610
5	6	-0.0053	1	0.0214	636	604
5	7	-0.0069	1	0.0152	645	608
5	8	0.0011	1	0.0119	648	612
5	9	-0.0011	1	0.0144	655	619
5	10	-0.0056	1	0.0070	660	625

Table 6: Gray scale images with different sizes from (1) to (30)





The Number of Changing Pixel Rate (NPCR) and the Unified Averaged Changed Intensity (UACI) are two most common quantities used to evaluate the strength of image encryption algorithms with respect to differential attacks. Here we have shown study of the correlation, NPCR and UACI using a set of gray scale images of different sizes, i.e., (128 x 128), (256 x 256), (512 x 512) and (1024 x 1024) [in Table 6] by using the proposed encryption algorithm and all those experimental results are given in Table 7. From the analysis it can be concluded that the proposed crypto processor is quite realistic and effective and it can be used for real time media transmission.

Table 7: Analysis on sizes of images

File name	Size	Correlation	NPCR	UACI
(1)	128x128	0.0023	98.5390	31.7639
(2)	128x128	0.0056	98.7901	32.4682
(3)	128x128	0.0022	96.0001	32.7902
(4)	128x128	-0.0127	97.0102	32.7825
(5)	128x128	-0.0010	96.4409	33.0025
(6)	128x128	0.0035	98.4329	32.5632
(7)	128x128	0.0106	97.2821	31.9045
(8)	128x128	0.0009	95.4520	35.5624
(9)	256x256	0.0096	97.7556	31.7832

(10)	256x256	0.0183	96.8903	30.2537
(11)	256x256	-0.0910	99.0245	31.4467
(12)	256x256	0.0661	95.6673	32.8529
(13)	256x256	0.0039	97.2105	31.4628
(14)	256x256	0.0021	97.0024	33.2690
(15)	256x256	0.0017	96.5622	31.3480
(16)	256x256	0.0741	98.1092	34.4671
(17)	512x512	-0.0025	99.0013	32.2379
(18)	512x512	-0.0022	95.2390	33.0091
(19)	512x512	-0.0178	93.7893	31.0093
(20)	512x512	0.0362	94.1342	29.6538
(21)	512x512	0.0039	96.2561	32.9084
(22)	512x512	-0.0101	97.4430	31.4681
(23)	512x512	0.0200	95.8905	31.4630
(24)	512x512	-0.0078	97.3356	30.3589
(25)	1024x1024	-0.0560	97.4461	30.2680
(26)	1024x1024	-0.0102	98.2573	29.3496
(27)	1024x1024	-0.0023	94.6790	32.7950
(28)	1024x1024	-0.0298	95.6648	32.3512
(29)	1024x1024	0.0012	97.9046	31.9051
(30)	1024x1024	0.0245	97.2555	31.7326

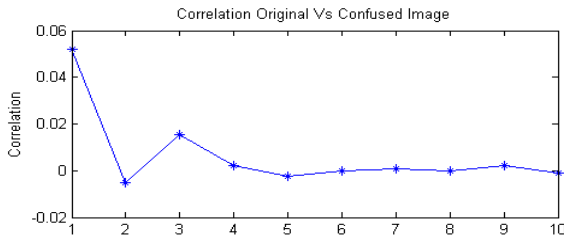


Fig 8: Iteration vs Correlation (Original vs Confused Image)

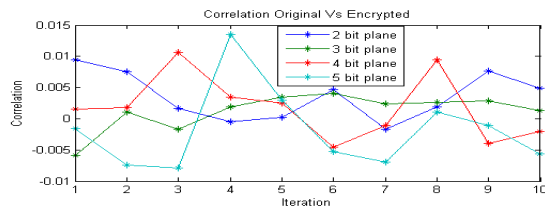


Fig 9: Iteration vs Correlation (Original vs Encrypted)

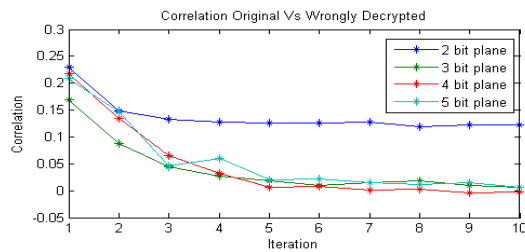


Fig 10: Iteration vs Correlation (Original vs Wrongly Decrypted)

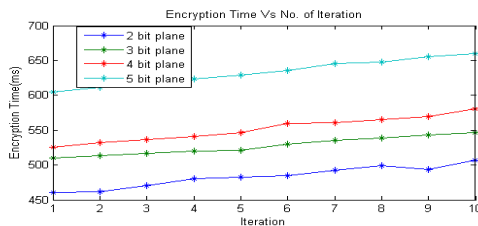


Fig 11: Iteration vs Encryption time (ms)

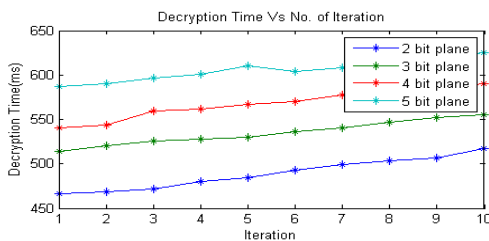


Fig 12: Iteration vs Decryption time (ms)

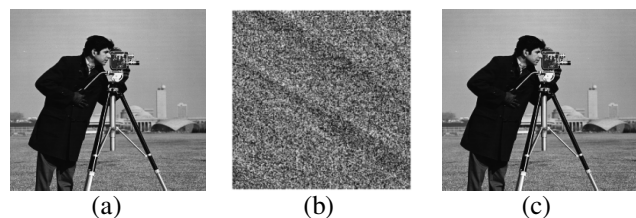
Experimental results shown in Table 1 to Table 5, we have done a systematic analysis using line graph from figure 8 to figure 12. Figure 8 shows the different correlations among original image and confused images obtained after implementing Arnold Cat Map with number of iterations

from one to ten times. This graph shows that two times onward iterations are giving an efficient correlation between original image and confused image which is near about zero, so we consider the minimum effective iteration for confused image i.e., two times iteration of Arnold Cat Map on the original image.

Once confused image is being finalised then it is being separated to eight different bit-planes to shuffle the bits using 3D Logistic Map to obtain the expected encrypted image. We have implemented 3D Logistic Map on four set of bit-planes MSB planes which are {first, second}, {first, second, third}, {first, second, third, fourth}, {first, second, third, fourth, fifth} to shuffle these bits among all the eight bit-planes and analysed with the number of iterations one to ten times for all the four sets. The reason for not considering 6th, 7th and 8th bit-planes from direct implementation of logistic map is that lower bit-plane consists of almost negligible information which is shown in figure 6.

Now figure 9 shows that all the four sets with number of iterations one to ten times give nearly zero correlation between original image and encrypted image. So, we have gone through one more analysis shown in figure 10 i.e., decrypt the encrypted image by changing a single bit on the original 48-bit secret key. Here, we noticed that though original image and wrongly decrypted image by considering two MSB planes are different their correlations among pixels are quite higher and it may be easier for attackers to predict the original image from wrongly decrypted image. In case of {first, second, third}, {first, second, third, fourth}, {first, second, third, fourth, fifth}; for three set of MSB planes from third iteration (i.e., logistic map) onwards correlation between original image and wrongly decrypted images are quite lower and can be said to nearly about zero.

Again more number of bit-planes and more iteration will slow down the encryption and decryption process and the rate of slow down with increasing number of bit-planes and iterations are shown in figure 11 and figure 12. After analysing all those parameters it can be concluded that three number of MSB planes with three times 3D Logistic Map iteration gives an effective encrypted image with sufficient security with expected amount of encryption and decryption time.



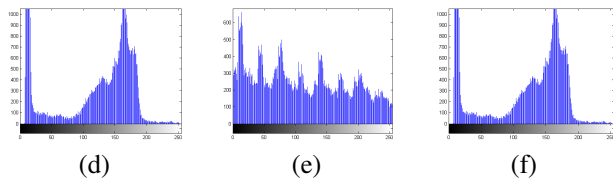


Fig 13: (a), (b), (c) are original image, encrypted image and decrypted image; (d), (e), (f) are the histogram of the images (a), (b) and (c) respectively

4. Conclusion

We propose a new chaotic map based two phase image encryption/decryption scheme in hardware implementation using Verilog code with an external secret key of 48-bit. Here, two different chaotic maps, namely Arnold Cat Map and 3D Logistic Map are used in confusion phase and diffusion phase respectively. In the proposed encryption method pixels are shuffled in confusion phase using Arnold Cat Map and in diffusion phase pixel value manipulations are done through bit-plane shuffling using 3D Logistic Map. Figure 14 shows the original image, encrypted image, decrypted image and its respected histograms. To check the robustness of the ciphered image we have tried to decipher the encrypted image by changing a single bit of the original secret key, but the original image is not retrieved and the correlation between original image and wrongly decrypted image is also quite lower. Finally, we can conclude that the proposed method can be implanted for real-time encryption for confidential gray scale image files.

References

[1] Leong, M.P.; Naziri, S.Z.M.; Perng, S.Y., "Image encryption design using FPGA," in *Electrical, Electronics and System Engineering (ICEESE), 2013 International Conference on*, vol., no., pp.27-32

[2] Pawan N. Khade, Prof. Manish Narnaware, "3D Chaotic Functions for Image Encryption", *IJCSI International Journal of Computer Science Issues*, Vol. 9, Issue 3, No 1, May 2012 ISSN (Online): 1694-0814

[3] Guanrong Chen, Yaobin Mao, Charles K. Chui, "A symmetric image encryption scheme based on 3D chaotic cat maps," *Chaos, Solutions and Fractals*, vol. 21, no. 3, pp. 749–761, 2004.

[4] Mao, Y., & Chen, G. 2005, "Chaos-based image encryption.", *Handbook of Geometric Computing*, 231-265

[5] Xin Ma, Chong Fu, Wei-min Lei, Shuo Li, "A Novel Chaos-based Image Encryption Scheme with an Improved Permutation Process", *International Journal of Advancements in Computing Technology* Volume 3, Number 5, June 2011

[6] Nitumoni Ha zarika , Monjul Saikia , "A Novel Partial Image Encryption Using Chaotic Logistic Map", 2014

International Conference on Signal Processing & Integrated Networks (SPIN), IEEE, 2014

[7] Shiguo Lian "Multimedia Content Encryption: Techniques and Application", CRC Press, ISBN 987-1-4200-6527-5, Pg 43-85.

[8] M. Saikia, S.J. Bora and Md. A. Hussain "A Review on Applications of Multimedia Encryption" in ISBN: 987-81-8487-088-6 in national conference on Network Security- issues, challenges and Techniques, at Tezpur University

[9] Monjul Saikia , Nitumoni Hazarika , Margaret Kathing "Partial Image Encryption using Peter De Jong Chaotic Map based Bit-Plane Permutation and it's Performance Analysis" published in ACEEE ITC 2014 on Mar 21st at Chandigarh, India ISBN: 978-94-91587-21-3 Page(s): 1 – 10

[10] Mayank Mishra, Prashant Singh, Chinmay Garg, "A New Algorithm of Encryption and Decryption of Images Using Chaotic Mapping", *International Journal of Information & Computation Technology*. ISSN 0974-2239 Volume 4, Number 7 (2014), pp. 741-746

[11] Pawan N. Khade, Prof. Manish Narnaware, "3D Chaotic Functions for Image Encryption", *IJCSI International Journal of Computer Science Issues*, Vol. 9, Issue 3, No 1, May 2012 ISSN (Online): 1694-0814

[12] K S Tamilkodi, (Mrs) N Rama, "A comprehensive survey on performance analysis of chaotic colour image encryption algorithms based on its cryptographic requirements", (*IJITCA*) Vol.5, No.1/2, April 2015

[13] N.K. Pareeka, Vinod Patidar, K.K. Sud, "Image encryption using chaotic logistic map", *Image and Vision Computing* 24 (2006) 926–934

[14] Reaz, M.B.I.; Mohd-Yasin, F.; Tan, S.L.; Tan, H.Y.; Ibrahimy, M.I., "Partial encryption of compressed images employing FPGA," in *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, vol., no., pp.2385-2388 Vol. 3, 23-26 May 2005

[15] Ta Thi Kim Hue; Chu Van Lam; ThangManh Hoang; Al Assad, S., "Implementation of secure SPN chaos-based cryptosystem on FPGA," in *Signal Processing and Information Technology (ISSPIT), 2012* pp. 129-134, 12-15 Dec. 2012

Bibliography:

Bikash Baruah: Research Scholar in the Department of Computer Science and Engineering, North Eastern Regional Institute of Science and Technology. His area of research VLSI design, cryptography, image processing etc.

Monjul Saikia: Assistant Professor in the Department of Computer Science and Engineering, North Eastern Regional Institute of Science and Technology. His major research interests include Information Security, Cryptography, Image and Video Processing, VLSI etc. He is a member of professional societies like IEEE, CSI (India), IEI (India) and ISTE (India).