Multi-Chaotic Image Encryption Algorithm Based on One Time Pads Scheme

Hazem Mohammad Al-Najjar and Asem Mohammad AL-Najjar

Abstract—**In this paper, we propose a new image encryption algorithm based on using multi-chaotic functions; to enhance the encryption algorithm and to increase the complexity of the encryption system so it's very difficult to break it and predict on it. Because of this, we use the Rossler attractor and logistic map to generate the encryption keys that used to change the value and the position of the pixels in the tested image. Moreover, our Approach depends on generating a onetime pad chaotic value by using a logistic map (Key1) with Z-plane from Rossler chaotic function as a second encryption key (Key2); to change the image pixels value. Where, X and Y planes as Key3 and Key4 respectively, are used to shuffle the image positions; to increase the uncertainty in the cipher image. However, by analyzing our algorithm, we show that the key space of our algorithm is equal to 10^{60}, where, the entropy tests show that the average entropy for the tested images is not less than 7.9970. Furthermore, after analyzing the histogram and correlation between the adjacent pixels, we show that our algorithm is strong against different types of attacks and it's sensitive to the initial conditions.**

Index Terms—**Image encryption, logistic map, rossler attractor, onetime pad.**

I. Introduction

The chaos theory was firstly used in the computer system by Edward Lorenz 1963. In his model, the system has two non-linear terms which increases the complexity of the system (sometimes called Lorenz attractor). Because of this, Otto E. Rössler, tries to enhance the Loernz attractor and designs his own model for chaos called Rossler equation or Rossler attracter. In his attractor, only one non-linear term was used. Moreover, many researchers try to design encryption systems by using chaos, like [1] they used a hybrid method by applying multi- chaotic systems; to increase the key space and make system's breaking very difficult. Where, in [2] they used Rossler chaos system by applying changes in the pixels value and their positions; to No index entries found. Increase the uncertainty in the cipher image. The one time pads with the logistic map (as a chaotic function) are used in [3] to encrypt the image and increase the size of the encrypted keys in the cipher. Where, in [4] an improved DES and the logistic map are used to encrypt the image. However, security analysis results and drawbacks of some chaotic cryptosystems are analyzed in [5]-[7]. On another hand, there are many problems that make the applying the chaos in the image encryption very weak such as: the existing number of invalid and weak keys and some keys are not sensitive for initial conditions. For this reason, this paper conducts to use two chaotic systems; to encrypt the image and increase the keys space. In our system, we used a Logistic map and Rossler system to encrypt the image. Where, the onetime pad chaotic function was used to change the pixels value and X and Y planes are used to shuffle the padded image.

The rest of this paper is organized as follows. In Section II, the onetime pad chaotic function is described on detail. Section III describes the Rossler chaotic function and how to use in the encryption system. Experimental results and security analysis are presented in Section IV. Finally, our conclusions are drawn in section V.

II. One-Time Pad Chaotic Function

A. Logistic Map

Chaotic function was mainly presented to be used in a computer technology in 1963 by Lorenz. Moreover, the chaotic functions are very sensitive to the initial conditions and have a deterministic behavior but unpredictable output. Then the cryptanalyst can't get the initial conditions if he has a chaotic sequence. On another hand, the sensitivity, of the system to the initial conditions means that small changes in the initial parameters will yield extremely different behavior. Therefore, because of his randomness, regularity, ergodicity and the sensitivity to the initial conditions, it is very useful to use the chaos in the encryption system. Furthermore, logistic map is one dimensional chaotic system; with X output and input variable and two initial conditions X_0 and λ that can be described as follows:

$$X_{n+1}=\lambda X_n(1-X_n) \qquad (1)$$

$\lambda \in [0,4]$, $X \in (0,1)$ in which, the chaotic behavior is achieved when $\lambda \in [3.57,4]$. In our encryption algorithm we used a logistic map; to generate chaotic sequences. Where, onetime pads modified version was used to change the pixels value (discussed in point B).

B. Onetime Pad

The main idea of the onetime pad is to generate keys with the same size as a plaintext in a random manner; to encrypt the pixels value. The onetime pad is a perfect cipher, since it's very hard to find a relationship between two cipher images; if they used in a random way. On another hand, it's very difficult to be implemented in the commercial products since it takes a long of time. The first version of the onetime pad is depending on the addition and modulus operation that's taking a long of time to implement it in the real system. Because of this, the modified version of the onetime pad that depends on the XOR operation was used; to minimize the execution time and the computation complexity that yielded from the first version. For example, in the first version to encrypt the pixel value 200 by using 100 as a key we add

Manuscript received March 10, 2012; revised May 6, 2012.

H. N. is with the Department of Computer, Taibah University, Madina, KSA (e-mail: hazem_najjar@yahoo.com).

them and find the modulus of the addition by using 255 like (200+100) mod 255= 45 and to decrypt the value we used a reverse operation (45-300) mod 255 = 200. Where, the modified version is very simple and can be implemented by using XOR operation only, for encryption and decryption, for example: to encrypt the same pixel value (200) with the same key (100) the encrypted pixel value will be equal to 73.

C. Onetime Pad by Using a Logistic Map

To enhance the onetime pad and use it more efficiently we need to randomize the keys; to increase the uncertainty in the random distribution. To do that, the logistic map and Rossler chaotic functions (Section 3) with sensitive initial conditions were used in our algorithm; to increase the space of the encryption keys. Assume the image size *M x N* our system can be described as follows:

1) Generate the random value by using a logistic map called it RL.
2) Get the first pixel and XOR it with the RL.
3) Generate the random value by using a Rossler chaotic function (*Z*- dimension) (RC) and XOR it with the result in step 2.
4) Repeat (1-3) to all the pixels in the image. Therefore, the output pixel can be written as follow:

$$OneTimePad(RC, RL, i, j) = XOR(RC, XOR(RL, pixel(i, j))) \quad (2)$$

where, RL, RC are the input from the chaotic generator by using a Logistic map and Rossler function respectively. And i and j are the indexes of the pixel on the image.

III. Rossler Chaotic function

If The Rossler chaotic function is a three dimensional ordinary differential equation, with one non-linear term (sometimes called Rossler attractor). The Rossler attractor was proposed in 1976 as an enhanced model of the Lorenz chaotic attractor that contains two nonlinear terms. Moreover, the ordinary differential equation can generate a chaotic behavior under certain conditions, which is defined in the following equations:

$$\begin{cases} \dfrac{dx}{dt} = -(y+z) \\ \dfrac{dy}{dt} = x + ay \\ \dfrac{dz}{dt} = b + z(x-c) \end{cases} \quad (3)$$

x, y, z, t, a, b and *c* ∈ *R*, depending on the chaos theory some ordinary differential equations may have a chaotic behavior under certain conditions. In the Rossler function to generate the chaotic behavior the space variables should be in the following ranges: $-15<x<17$, $-16<y< 13$ and $0<z<36$ where the classic chaotic attractor that studied by Rossler defined *a*, *b*, and *c* as 0.15, 0.20 and 5.7, respectively. The Rossler attractor with the three dimensional system is shown in Fig. 1 and the time sequences of the three dimensional variables are shown in Fig. 2.

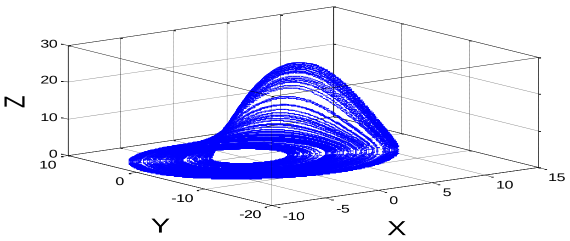


Fig. 1. Rossler attractor

In our algorithm, we take the Z-plane as the input of the onetime pad and *X*, *Y* as the input to shuffle the pixels. Moreover, after generating the *x*, *y* and *z* variables the pre-processing method is applied on them to enhance the auto-correlation as discussed in [3].

The pre-processing function can be described in the following equation [3]:

$$V(i) = 10^n V_n(i) - round(10^n V_n(i)) \quad (4)$$

In which, n is the right shift the number *V*(*i*) n digits and *V* is the plane to enhance *x*, *y* or z.

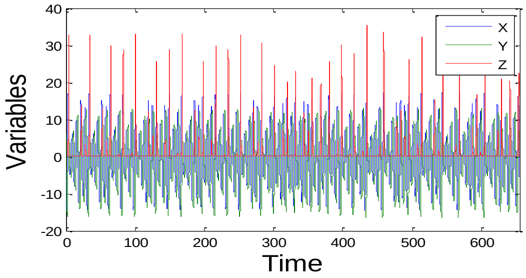


Fig. 2. Time sequences of *x*, *y* and *z* variables

A. Shuffling Method

In the shuffling method, each pixel on the image changed their position, so the adjacent pixels are no longer adjacent. Moreover, this will enhance the encrypted image and decrease the correlations between the adjacent pixels. So, no useful information will be gotten from the image. In the following our shuffling approach is described:

1) For pixel (*i*, *j*) (where *i*, *j* are the indexes of the image) generate the random position *x*, *y* by using the *X* and *Y* planes from the Rossler function and change the position of the current pixel to that position.
2) Repeat 1 to all the pixels in the image. After that, we got the shuffled image.

B. Encryption Scheme Diagram

Our algorithm as shown in Fig.3 is divided into mainly two parts: one time pad; to change the pixels value and the shuffling approach; to change the pixels position. Moreover, to create onetime pad we used a logistic map with Z-plane from the Rossler function. Where, to shuffle the padded image pixels position we used *X*, *Y* planes from the Rossler functions. Finally, the decryption process is done on the reverse order.

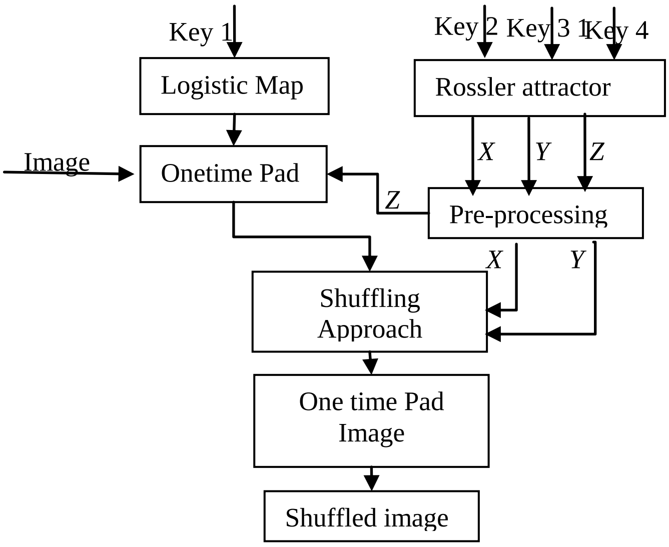
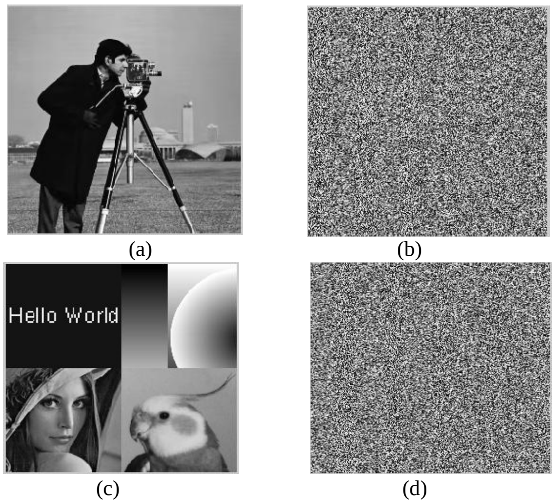


Fig. 3. Encryption algorithm diagram

IV. Experimental Results and Security Analysis

To test our algorithm we take cameraman image and montage image with a size 256 x 256. In Fig.4 (a–b) shows the cameraman image and cipher image respectively, where, Fig.4 (c-d) shows the montage image and cipher image respectively. With input keys Key1= 1 x10^{-14}, Key2= 1.1045, Key3= 1.2831 and Key4 =1.3682 for two images.



(a) (b)

(c) (d)

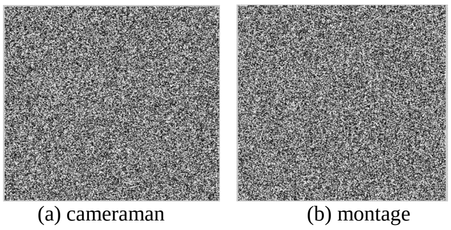
Fig. 4. Encryption for cameraman and montage images

A. Keys Space Analysis

The strength of any encryption algorithm depends on the key space of the system. If the system has a very large key space, then the brute force attack is not possible. For our algorithm, the key space is calculated as follow: we have four keys key1, key2, key3 and key4, the keys space for each one is equal to 10^{15} then the key space of the algorithm is equal to 10^{60}.

B. Keys sensitivity Analysis

The sensitivity of the initial conditions is one of the important properties of the chaotic functions. Therefore, the encryption system should be sensitive to the small changes in the decrypted keys and generate a wrong decrypted image in that case. Only the same keys should give the same image to the receiver side. Our sensitive tests keys are Key1= 2 x10^{-14} Key2= 1.1046, Key3= 1.2832 and Key4 =1.3683, in which, Fig.5 shows the decrypted image for the cameraman and montage images by using a wrong keys.



(a) cameraman (b) montage

Fig. 5. Sensitivity tests of keys

C. Information Entropy Analysis

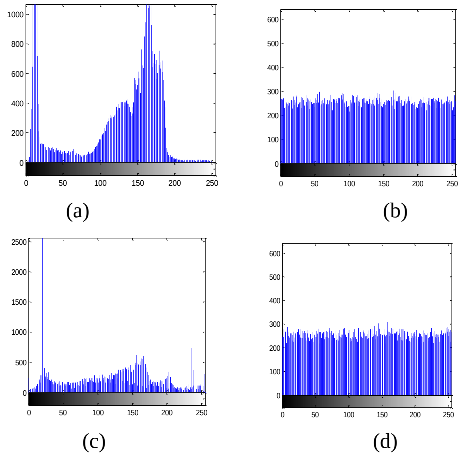
There are many definitions in the dictionary for the term entropy, in the data transmission and information theory we define the entropy as a measure of the loss of information in a transmitted signal or message. Where, in the statistical mechanics, is defined as a measure of the randomness of the microscopic constituents of a thermo-dynamic system. In this part, we interested in the randomness of our system, where, the true random variable should generate 2^{8} symbols with equal probability and the entropy value equal 8. To check the randomness of our random cipher image we used a following equation [8]:

$$H(s) = \sum_{s} P(S_i) log \frac{1}{P(S_i)} \qquad (5)$$

where P (S$_i$) represents the probability of symbol S$_i$, in our tests the average entropy of the cameraman cipher image is 7.9973 and for the montage cipher image is equal to 7.9970, which are very close to the optimal value then the entropy attack is not possible.

D. Histogram Analysis

The histogram is the one of the important properties of the image, since the cryptanalyst can get very useful information from the image by using the histogram. In which, the good encryption algorithm should generate uniformly distribution of the histogram, so no information can be gotten from the histogram. In our tests, it’s very difficult to get any information from the histogram. Fig. 6 shows the histogram analysis of cameraman and montage image and there cipher images. Finally, after the histogram analysis, we show that the histogram is uniformly distributed.



(a) (b)

(c) (d)

Fig. 6. Histogram of cameraman and montage and cipher images, respectively.

E. Correlation Analysis

It's known that some algorithm was broken by using correlations between two adjacent pixels (in vertical, horizontal and diagonal). For this reason, we try to test our system by using a correlation analysis and by calculating the correlation coefficient, for all possible cases.

To find a correlation between the adjacent pixels the correlation coefficient is calculated by using the following formula [8]:

$$r = \frac{Cov(x,y)}{\sqrt{D(x)}\,\sqrt{D(y)}} \qquad (6)$$

$$D(x) = \frac{1}{M}\sum_{i=1}^{M}(x-\bar{x}\ )^2 \qquad (7)$$

$$Con(x,y) = \frac{1}{M}\sum_{i=1}^{M}(x-\bar{x})(y-\bar{y}) \qquad (8)$$

where, M is the total number of randomized pairs, x and y are the two vectors that contains x values and y values of the pair in the tested image, respectively. Table.1 shows the correlation coefficients between two adjacent pixels in all possible cases (vertically, horizontally and diagonally) of the plain-text images and cipher images. The results revealed that the proposed method randomized the pixels in very good way.

TABLE I: CORRELATION COEFFICIENTS OF ADJACENT PIXELS

Image Coefficient	Cameraman		Montage	
	Plain	Cipher	Plain	Cipher
Vertical	0.9535	-0.0062	0.9737	-0.0077
Horizontal	0.9524	0.0417	0.9363	0.0014
Diagonal	0.9198	-0.0061	0.9069	0.0425

F. Plain –Text Sensitivity Analysis

If the cipher image is not sensitive in the changing of the plaintext then the cryptanalyst can get very useful information from the encrypted image; to check that we use two criteria, NPCR (Number of Pixel Change Rate) and UACI (Unified Average Changing Intensity). Where, NPCR defined as a percentage of different pixels number between two cipher images and UACI defined as an average intensity of differences between two cipher images as defined in the following equations [8]:

$$NPC = \frac{\sum_{i,j} D(i,j)}{MxN} x100\% \qquad (9)$$

$$UAC = \frac{1}{MxN}\sum_{i,j}\frac{|C_1(i,j) - C_2(i,j)|}{255} x100\% \qquad (10)$$

where M x N is the size of the cipher images and C1 and C2 are two different cipher images encrypted by using a different keys, where D (i , j) is defined as follow:

$$D(i,j) = \begin{Bmatrix} 0 & C_1(i,j) = C_2(i,j) \\ 1 & C_1(i,j) \neq C_2(i,j) \end{Bmatrix} \qquad (11)$$

After calculations, we get the Average NPCR and UACI of cameraman image are: NPCR = 99.6597and UACI = 33.2669 and that of the montage are: NPCR = 99.5911 and UACI = 33.4814. Then our algorithm has a good ability against known plain text attacks.

V. CONCLUSIONS

In this Paper, we used two chaotic functions; to encrypt the image and to increase the keys space of our algorithm. In which, the logistic map with Z-plane from the Rossler chaotic functions are used to generate the value of onetime pad to encrypt the pixels value. After that, the X and Y planes are used to shuffle the image positions to decrease the correlation coefficient and increase the uncertainty of the random image. So, no useful information can be gotten from the cipher images.

However, we shown by experimental results that our algorithm is sensitive to initial conditions and strong against the brute force attacks. And, we found that our algorithm has a high secure against different types of attacks with the large space of the encryption keys. Finally, in our future work we try to study different features of the chaotic functions; to enhance the image encryption system, execution time and computation complexity of the cipher images.

REFERENCES

[1] S. Chen, B. Mulgrew, and P. M. Grant, "A clustering technique for digital communications channel equalization using radial basis function networks," *IEEE Trans. on Neural Networks*, vol. 4, pp. 570-578, July 1993.

[2] H. Nien; W. Huang; C. Hung, C. Huang, and Y. Hsu. "Hybrid image encryption using multi-chaos-system, Int. Conf." in *Information, Communications and Signal Processing (ICICS)*, 2009, pp. 1-5.

[3] Y. Cao and C. Fu. "An image encryption scheme based on high dimension chaos system," *Int. Cof. Intelligent computation technology and automation*, 2008, pp. 104-108.

[4] J. Jeyamala, S. GrpiGranesh, and S. Raman. "An image encryption scheme based on one time pads- a chaotic approach," *Int. Conf. on computing*, communication and networking technologies, 2010, pp. 1 – 6.

[5] Z. Yun-Png, Z. jun, L. Wei, N. Xuan, C. Ping, and D. di. "Digital Image Encryption Algorithm Based on Chaos and Improved DES," *Int. Conf. On System*, Man and Cybernetics, San Antonio, TX, USA, 2009, pp.474-478.

[6] L. Shujun, L. Chengqping, C. Guangrong, G. Nikolas, and L. Kwok-Tung. "A general quantitative cryptoanalysis of permutation-only multimedia ciphers against attacks," *Signal Processing: Image Communication*, 2003, no. 23, pp. 212-223.

[7] R. Rhouma and B. Safya. "Cryptoanalysis of a new image encryption algorithm based on hyper-chaos," *Physical Letters A*, 2008, vol. 372, pp. 5973-5978.

[8] X. Di, L. Xiaofeng, and W. Pengcheng. "Analysis and improvement of a chaos image encryption algorithm," *Chaos, Solution and Fractals*, 2009, vol. 40, pp. 2191-2199.

[9] M. Long and L. Tan. "A chaos –based data encryption algorithm for image/video," *Int. Conf. on Multimedia and information technology*, 2010, pp. 172-175.

Hazem M. Al-Najjar was born in Jordan in 1986. He received the M. Sc. degree in computer engineering from Jordan University of Science and Technology (JUST), Irbid, Jordan, in 2011 and the B. Sc. degree in computer engineering from Yarmouk University, Irbid, Jordan, in 2008. Since February 2012, he has been with the Department of Computer, Taibah University, Madina, KSA. His current research interest is in wireless networks with emphasis on wireless sensor networks, grid computing, network coding, image and data encryption and Mobile payment systems.

Asem Al-Najjar was born in Jordan in 1990 undergraduate bachelor student in Computer Science from Yarmouk University. He is Interested In java programming, oracle SQL, PL/SQL and forms programming and image encryption .