ISSN 2277-3061

A Novel Chaos-Based Stream Cipher for Image Encryption

Salwa K. Abd-El-Hafiz, Gamal A. Fuad, Lina A. Sayed

Engineering Mathematics Department, Faculty of Engineering, Cairo University, Giza, Egypt
sabdelhafiz@gmail.com
Mathematics Department, University College of Women, Ain Shams University, Cairo, Egypt
gam_ismail@yahoo.com
Mathematics Department, University College of Women, Ain Shams University, Cairo, Egypt
lina.khamis@gmail.com

ABSTRACT

In this paper, we introduce a novel chaos-based image encryption scheme. We perform our scheme using both the binary form and decimal form of chaotic maps. We also use two chaotic maps; one generates the initial values of the other. Thus, we utilize the best two properties of chaos ergodicity and sensitivity to the initial value. We increase the security of the cipher by introducing a reverse order round to resist cryptanalysis attacks.

Indexing terms/Keywords

Chaotic map, Encryption, Decryption, and Reverse order round.

SUBJECT CLASSIFICATION

Mathematics Subject Classification; 37D45, 94A60.

ISSN 2277-3061

INTRODUCTION

In recent years, the transmission of digital images over communication media has developed greatly. The problem of security in storage and transmission of confidential visual information is therefore growing in importance, and requires solutions for many applications. Recently, a number of chaos-based image encryption systems and random number generation algorithms based on discrete chaos were proposed [1,3,7,9-13], but security is generally not high enough. In fact, many chaos-based cryptography schemes have shortcomings. In this paper, we introduce a new scheme to encrypt an image. Its idea is to generate from one initial condition several chaotic initial conditions by the first chaotic map, e.g., Logistic map, for another chaotic map, e.g., PWLCM. Then, we use the resultant chaotic sequences of the second map in both forms, the binary form and the real number form, and apply it to diffuse and confuse the image. Also, we use a delay operation to increase the security.

ENCRYPTION SCHEME

In this section, we briefly discuss our proposed method to encrypt any colored image. Let us begin with the key design and then introduce the encryption process and decryption process to recover the plain image.

Key Design: The key is the basic part in the encryption-decryption processes. We set it to be composed of three real numbers between 0 and 1 (each part is 64 bits). As we will discuss below, we will use two different maps; one generates the initial conditions for the other. Hence, these three numbers will be chosen to be linearly mapped into the intervals identifying the initial condition x_0, the parameter p1 for the first map and the second map parameter p2. It is to be noted that we are concerned only with the parameters' intervals causing the chaotic behavior. Here we will use the Logistic map firstly and apply it for the PWLC map where $0< x_0 <1$, $3.56994567\ldots< p1 <4$ and $0< p2 <0.5$ (please refer to their bifurcation diagrams in [2]). Therefore, the total **key length** is 192 bits (3 times 64 bits).

Let the colored image has M×N pixels. Fig. 1 shows the encryption scheme diagram. To encrypt the image, we proceed as follows:

1) Take the initial condition and run the first map N times.

2) Use each of the N outputs as an initial value to the second map and run it up to M iterates. So we get S an M×N matrix of chaotic real numbers.

3) Now, we start with diffusion. We will take the horizontal form of S to be horzS. Multiply it first by M andconsidering the integer part only and calculate 'q1' vector by picking up the first different M integer numbers and then by N for the rest of horzS to calculate 'q2' vector by the same way.

   For example: if we have 3×4 image matrix in the form,

$$\mathcal{A}=\begin{pmatrix}1 & 2 & 3 & 4\\ 5 & 6 & 7 & 8\\ 9 & 10 & 11 & 12\end{pmatrix}$$

   and horzS=[0.5051 0.2320 0.8737 0.2745 0.5966 0.8769 0.2677 0.5819 0.9088 0.1982 0.7891 0.9369], then integer value of (horzS×3) = [**2 1 3** 1 2 3 1 2 3 1 2 3] and q1 = [2 1 3]. For q2, we multiply the rest of horzS×4 to get the sequence [**1 2 4** 1 2 4 1 **3** 4] and q2 = [1 2 4 3].

4) Diffuse the image by permuting its columns and rows for each color using q1 and q2. Let the diffused image be P.

   For the previous example,

$$P=\begin{pmatrix}5 & 6 & 8 & 7\\ 1 & 2 & 4 & 3\\ 9 & 10 & 12 & 11\end{pmatrix}$$

5) Return to S and change its decimal form to binary form for 24 bits. For example, let S(i,j)=0.123456789012345 then its digital form will be e=000111111001101011011101. Divide e into 3 bytes e1, e2 and e3. Put e3 in E1(i,j), e2 in E2(i,j) and e1 in E3(i,j). We get a chaotic digital image E.

6) The delay element: we will use the P image and E image in a horizontal form and apply the pixels encryption delay by this way (it is similar to that in [3], but we will add Reverse order round in the next step):

$$C'(i)= E(i) \oplus (P(i) + C'(i-1)) \bmod 256.$$

7) Reverse order round: we run step 6) again but by taking the reverse order of C' (i.e., C'(MN), C'(MN-1), C'(MN-2), …, C'(1)) instead of P. And reperform this equation

$$C(i)= E(i) \oplus (C'(i) + C(i-1)) \bmod 256$$

   Finally, the resulted image is the Cipher image C.

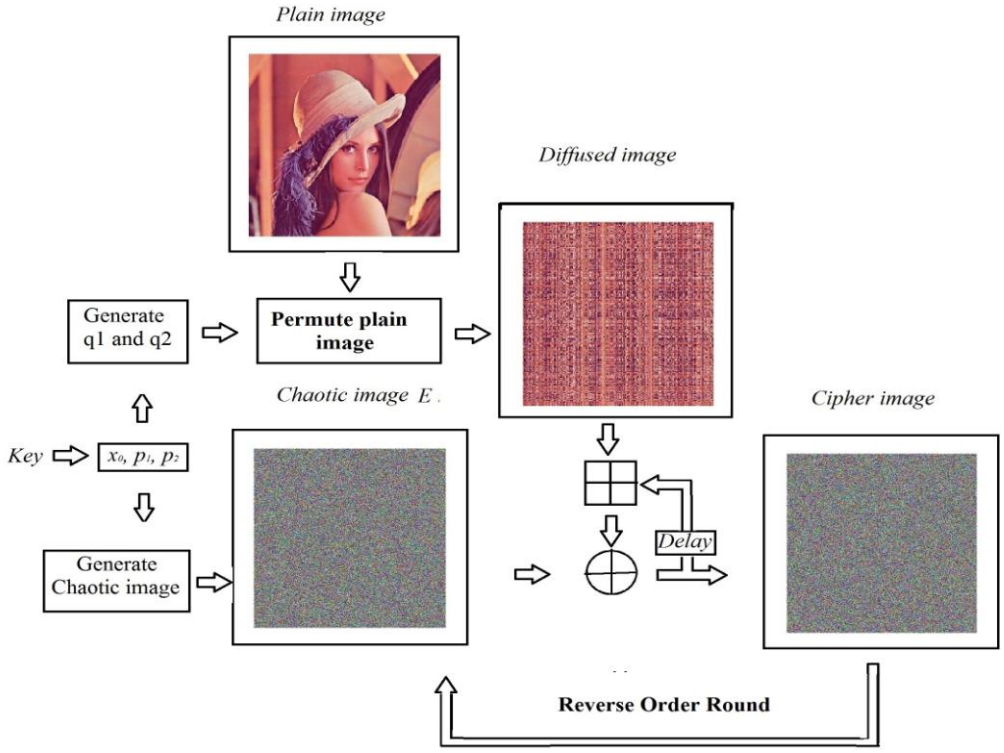


Fig. 1: Encryption scheme

Decryption:

It is easy to decrypt the cipher image. We can reverse these steps and get the original image as follows:

1) The inputs to the decryption are the cipher image C and the key. So we begin by extracting q1, q2 and the chaotic image E by using the key.
2) Apply the next formula:

$$C'(i) = (E(i) \oplus C(i) - C(i-1)) \bmod 256.$$

3) Repeat step 2 with the reverse of C' instead of C to get the P image as follows,

$$P(i) = (E(i) \oplus C'(i) - C'(i-1)) \bmod 256.$$

4) Use q1 and q2 to get the plain image.

EVALUATION CRITERIA

1) **Histogram of the cipher**: It illustrates how characters/pixels in a text/image are distributed by graphing the number of characters/pixels at intensity level. To avoid statistical attacks, the histogram of encrypted should be different from original plaintext and fairly **uniform**.
2) **Correlation coefficient:** it describes the relation between two sequences values x, y by the formula:

$$r_{x,y} = \frac{\operatorname{cov}(x,y)}{\sqrt{D(x)D(y)}}$$

where

$$\operatorname{cov}(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y}), \qquad D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2$$

3) **Entropy:** It is the most important feature of randomness. Let m be a character, and the entropy formula is

$$\mathcal{H}(m) = \sum_{i=0}^{255} p(m_i)\log_2 \frac{1}{p(m_i)}$$

since there are 2^8 characters, if we assume that each will appear with the same probability then *H(m)=8* which shows that the cipher is random.

4) **The Mean Absolute Error (MAE) and the Mean Square Error (MSE):**The Mean Absolute Error (MAE) is calculated to measure how the cipher image C is different from the plain image P. It is given by

$$\mathcal{MAE} = \frac{1}{\mathcal{MN}} \sum_{i=1}^{\mathcal{M}} \sum_{j=1}^{\mathcal{N}} |\mathcal{P}(i,j) - C(i,j)|$$

The Mean Square Error (MSE) is used to measure the difference between the plain image P and wrong decrypted image D. It is calculated by

$$\mathcal{MSE} = \frac{1}{\mathcal{MN}} \sum_{i=1}^{\mathcal{M}} \sum_{j=1}^{\mathcal{N}} (\mathcal{P}(i,j) - \mathcal{D}(i,j))^2$$

5) **Differential attacks:** It is used to analyze image pixels. If the encrypted image can cause significant changes even with one-bit difference in pixel of the original plain image, then, the algorithm can resist the known plaintext attacks and the chosen-plaintext attacks. The Number of Pixels Change Rate (NPCR) and the Unified Average Change Intensity (UACI) are defined as follows [11]:

$$\mathcal{D}(i,j) = \begin{cases} 0 & C(i,j) = C'(i,j) \\ 1 & C(i,j) \neq C'(i,j) \end{cases}, \quad \mathcal{NPCR} = \frac{1}{\mathcal{MN}} \sum_{i=1}^{\mathcal{M}} \sum_{j=1}^{\mathcal{N}} \mathcal{D}(i,j) \times 100\%$$

$$\mathcal{UACI} = \frac{1}{\mathcal{MN}} \sum_{i=1}^{\mathcal{M}} \sum_{j=1}^{\mathcal{N}} \frac{|C(i,j) - C'(i,j)|}{255} \times 100\%$$

where C is the encrypted image and C' is the encrypted image after one pixel change in the plain image. If NPCR is over 99% and UACI is near 33.4% then the encryption method can resist differential attacks efficiently.

6) **NIST tests:** The National Institute of Standards and Technology (NIST) introduced the NIST Statistical Tests [8]. NIST tests are a group of different tests, we will perform 13 of them to investigate the randomness of the encrypted image.

ANALYSIS

We perform this scheme on Lena image of size 256×256 (as shown in Fig. 1). We apply the previous tests discussed in the previous section. Table 1 and Table 2 show the plain and cipher images analysis, respictively.

Table 1. Plain image entropy and correlation analysis

	Plain image			
	Entropy	**Corr-H**	**Corr-V**	**Corr-D**
Red	7.3055	0.9762	0.9783	0.8879
Green	7.6197	0.9671	0.9702	0.8893
Blue	7.0723	0.9677	0.9706	0.8129
Ave	7.3325	0.9703	0.9730	0.8634

Table 2. Cipher Image analysis results

	Entropy	**Corr-H**	**Corr-V**	**Corr-D**	**MAE**	**NPCR**	**UACI**
Red	7.9976	-0.0120	-0.0134	0.0106	84.5608	99.6338	33.3816
Green	7.9972	-0.0174	0.0132	-0.0155	78.3732	99.6216	33.7062
Blue	7.9974	-0.0115	0.0129	-0.0110	70.2240	99.6552	33.2839
Ave	7.9974	0.0136	0.0132	0.0124	77.7193	99.6369	33.4572

ISSN 2277-3061

The cipher image performed analysis shows that its pixels are truely uncorrelated and demonstrates very high entropy values compared to that of the plain image results. In Table 2, The entropy test is more than 7.99, which indicates highly randomized unpredictable values between 0 and 255 for the cipher images. The Histograms of Fig. 2 indicate also the uniform distribution of the pixel values. Pixel correlation coefficients for horizontal, vertical, and diagonal representations were around 0.0l3, implying that the encryption process produces ciphers with extremely low correlation between neighboring pixels (see also Fig. 3). The differential attack results show great independence between the two cipher images slightly different in their plain images; the NPCR was 99.6369 and its UACI was 33.4572.

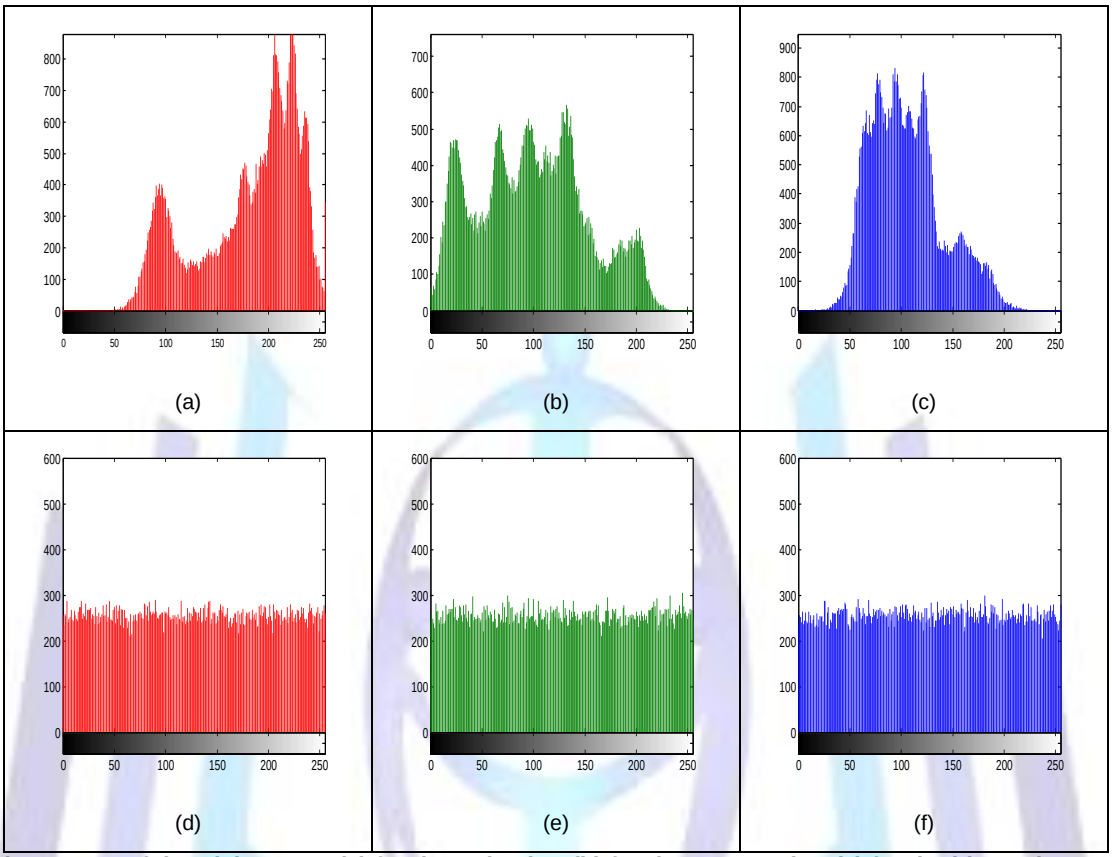
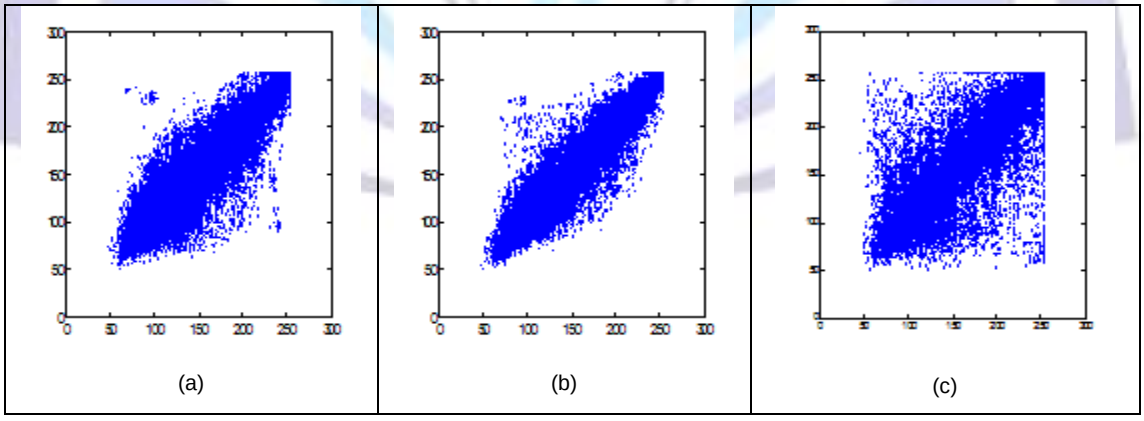


Fig. 2: Histograms of the Plain Image (a) for the red color, (b) for the green color, (c) for the blue color, and that of the Cipher image (d) for the red color, (e) for the green color and (f) for the blue color.



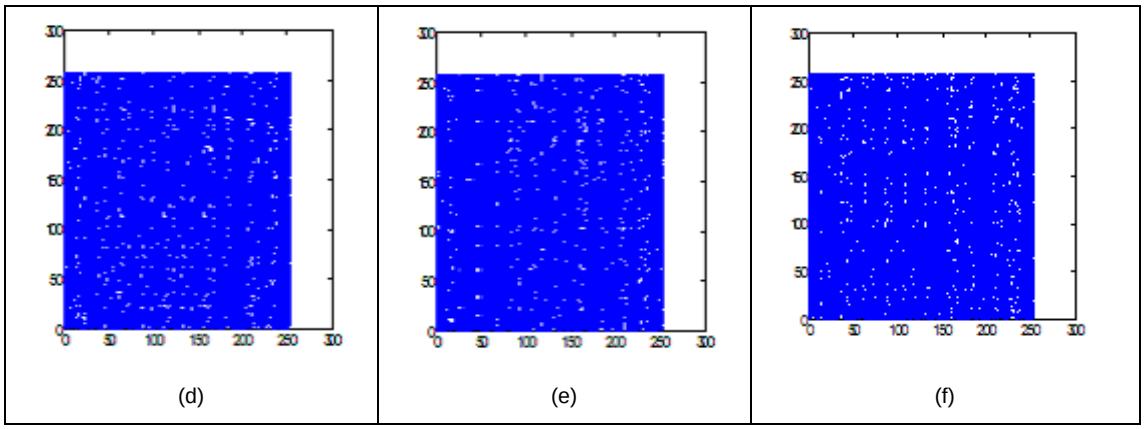


Fig. 3: Red color Plain and Cipher images correlation analysis (a, d) the horizontal correlations, (b, e) vertical correlations and (c, f) the diagonal correlations, respectively.

In addition, Table 3 shows the NIST results where all the thirteen tests have been passed successfully.

Table 3. NIST tests results of the cipher images

	Lena Image	
Test	**PV**	**PP**
Frequency	✓	0.9996
Block Frequency	✓	0.9996
Runs	✓	0.9996
Longest Runs	✓	0.9996
Approximate Entropy	✓	0.9996
Universal	✓	0.9996
Linear Complexity	✓	0.9996
Overlapping Template	✓	0.9996
Non-overlapping Template	✓	0.9996
Serial	✓	0.9996
	✓	0.9996
Cumulative Sum	✓	0.9996
	✓	0.9996
Random Excursion	✓	0.9999
Random Excursion Variant	✓	0.9958

SENSITIVE ANALYSIS

Any image encryption algorithm thought to be secure should have a large key space that makes brute attacks ineffective. In our proposed algorithm the key affects both the diffusion and confusion stages. So, a good encryption algorithm should be sensitive to the secret key and the plaintext. If the algorithm is sensitive to the secret key, then if we use another key with only small different for image encryption; the results should be very different. The sensitivity to the secret key can be quantified by the Mean Square Error (MSE). The MSE test is used to determine how much the original image is different from the wrong decrypted image.

For the Lena encrypted image, we used a key K = (0.232323, 0.9534942480407069, 0.92) which are linearly mapped to be corresponding to $x_0 = 0.232323$, $p_1 = 3.98$, and $p_2 = 0.46$. Let us test the key sensitivity by decrypting the cipher image three times using 10^{-14} difference in the three key parts.

Test 1: If we take $k_0 = 0.232323000000001$ instead of $k_0 = 0.232323$, the decrypted images will be as shown in Figure 4(a).

Test 2: If we take $k_1 = 0.95349424804071$ instead of $k_1 = 0.95349424804070$, the decrypted images are as shown in Figure 4(b).

Test 3: If we take $k_2 = 0.92000000000001$ instead of $k_2 = 0.92$, the decrypted images are as shown in Figure 4(c).

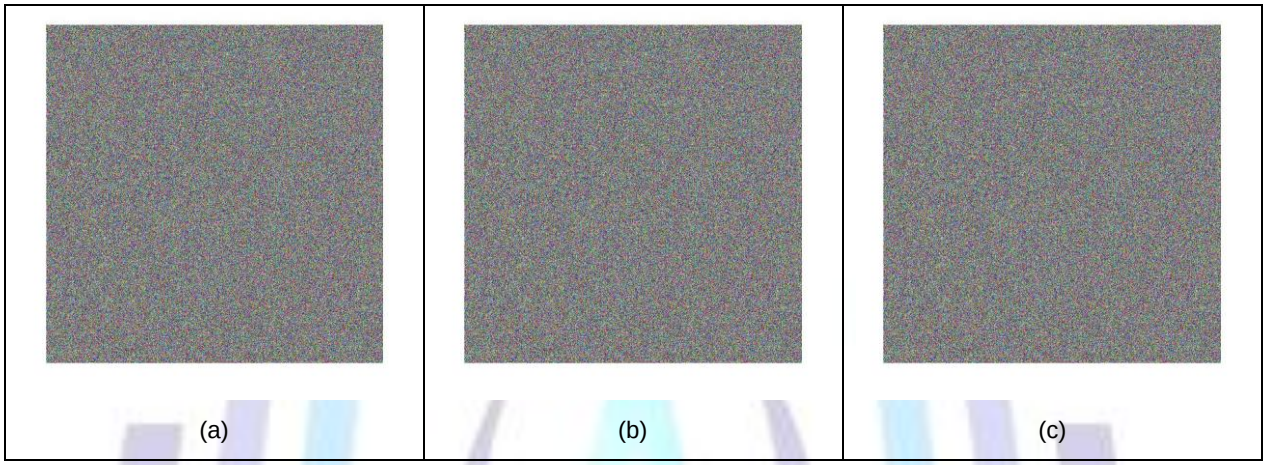


Fig 4: (a) Test 1, (b) Test 2, and (c) Test 3.

Table 4. Entropy and MSE for the three sensitivity tests

	Test 1		Test 2		Test 3	
	Entropy	**MSE**	**Entropy**	**MSE**	**Entropy**	**MSE**
Red	7.9969	10773.9726	7.9970	10682.5554	7.9973	10729.9917
Green	7.9972	9773.6320	7.9970	9147.4489	7.9973	9085.6175
Blue	7.9967	7140.0783	7.9971	7233.5787	7.9972	7160.1420

Table 4 shows the MSE and entropy results of the three mentioned tests for the Lena wrong decrypted image. It is clear from the results that the key is very sensitive for a very small change and that the key space is 10^{42}.

Now, we can compare our results to other references for colored Lena image of size 256 × 256 as follows:

Table 5: Comparison between our proposed scheme results with other works

	Corr-H	**Corr-V**	**NPCR**	**UACI**
Proposed	0.0136	0.0132	99.6369	33.4572
[9]	0.0112	0.0104	99.5051	33.4889
[5]	0.0068	0.0033	99.6538	33.9218
[1]	0.0209	0.0144	99.61	33.41
[3]	-0.01145	-0.0192	----------	----------

Table 5 shows that our proposed scheme is good as other encryption methods in correlation and differential attack results.

CONCLUSION

In our encryption scheme, we introduced a novel image encryption method based on chaotic maps. We used the good randomness properties of the PWLC map and Logistic map to get a highly secured encryption process associated with a large key length to resist brute force attacks. We also introduced the reverse order round, which makes the cipher image highly sensitive for any bit change in any pixel of the plain image. The cipher image shows very good statistical analysis results, correlation coefficients, histogram distributions and differential attack measures.

REFERENCES

[1] M. Amin, O. S. Faragallah and A. A. Abd El-Latif, "*A chaotic block cipher algorithm for image cryptosystems*". Commun Nonlinear Sci Numer Simulate 15 (2010)3484-3497.

[2] S. El-Assad, H. Noura and I. Taralova, "*Design and analyses of efficient chaotic generators for crypto-systems*", Advances in Electrical and Electronics Engineering- IAENG Special Edition of the World Congress on Engineering and Computer Science IEEE, vol. I, pp. 3-12, 2008.

[3] M. K. Feng, S. S. Qiu, X. Y. Liu and J. X. Jin "*Research on Test of Random-like Property of Chaotic Sequences in Image Encryption*", Fifth International Conference on Natural Computation IEEE, 2009.

[4] L. Hongjun and W. Xingyuan, "*Color image encryption based on one-time keys and robust chaotic maps*", ScienceDirect: Computers and Mathematics with Applications 59(2010) 3320-3327.

[5] R. Joshi and S. Joshi, "*Color Image Encryption through a Novel Chess Based Confusion Scheme using Chaotic Map*", Signal Processing and Information Technology (ISSPIT), IEEE International Symposium on,2011.

[6] S. Lian, J. Sun and Z. Wang, "*Security Analysis of A Chaos-based Image Encryption Algorithm*", The paper was accepted by Phisica A, Elsevier Science, 2005.

[7] Y. Mao, G. Chen and S. Lian, "*A Novel Fast Image Encryption Scheme Based On 3D Chaotic Baker Maps*", International Journal of Bifurcation and Chaos, Vol. 14, No. 10(2004) 3613-3624.

[8] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray and S. Vo. "*A Statistical test suite For random and pseudorandom number generators FOR Cryptographiic appliictiions*". NIST Special Publication, (2010)800-22.

[9] X. Tong, Y. Liu, M. Zhang and Z. Wang, "*A Novel Image Encryption Scheme Based on Dynamical Multiple Chaos and Baker Map*", 11th International Symposium on Distributed Computing and Applications to Business, Engineering and Science, IEEE, 2012.

[10] M. Usamaa, M. K. Khana, K. Alghathbar and C. Lee, "*Chaos-based secure satellite imagery cryptosystem*", Science Direct: Computers and Mathematics with Applications, 60(2010)326-337,.

[11] Y. Wu, J. P. Noonan and S. agaian, "*NPCR and UACI Randomness Tests for Image Encryption*", Cyber Journal: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT), April Edition, 2011.

[12] W. Yanling, "*Image Scrambling Method Based On Chaotic Sequences and Mapping*", First International Workshop on Education Technology and Computer Science IEEE, 2009.

[13] R. Ye, W. Zhou and H. Zhao, "*An Image Hiding Scheme Based on 3D Skew Tent Map and Discrete Wavelet Transform*", Fourth International Conference on Computational and Information Sciences IEEE, 2012.