2013 Fourth International Conference on Emerging Intelligent Data and Web Technologies

Image encryption scheme based on multiple chaotic maps

Xuan Li*
Information Science School
Guangdong Commercial College
Guangzhou GuangDong 510320, China
e-mail: lx_careerlife@163.com

Abstract—**a novel image encryption scheme based on even-symmetric chaotic maps and skew tent chaotic map was proposed. In the permutation process, a P-box produced by sorting an even-symmetric chaotic sequence is applied to shuffle the position of all image pixels. In the diffusion process, both even-symmetric chaotic maps and skew tent map are used to generate the key stream. Then the pixels in the permuted-image determine which of two even-symmetric chaotic maps to iterate for next byte in the keystream each time. The performance and security of the proposed method are evaluated thoroughly using key space analysis, statistical analysis, sensitivity analysis and so on. Results are encouraging and suggest that the scheme is reliable to be adopted for the secure image communication application.**

Keywords-image encryption; symmetric chaotic map; skew tent chaotic map; permutation; diffusion

I. INTRODUCTION

With the rapid developments of computer technology and multimedia industry, the protection of digital information transmitted or stored on the internet becomes more and more important. An effective method is to encrypt the digital information so that only the authorized entities with the key can decrypt it. Since digital image has some intrinsic features such as bulky data capacity, high redundancy and high correlation among pixels, the conventional encryption schemes such as DES, AES and RSA are not considered suitable for image encryption [1]. In recent years, chaos-based image encryption system has been widely investigated because of the fundamental feature of chaotic map such as sensitivity to initial condition, random-like behaviors and aperiodicity [2-8].

A general chaos-based image encryption method is composed of two steps: pixel permutation and diffusion. In the permutation stage, a P-box is usually generated to shuffle the pixel position in order to destroy high correlation of pixels in the image. In the diffusion stage, the pixel values are modified sequentially so that the histogram is totally changed and a tiny change of plain image can spread out to almost all pixels in the cipher image. However, many chaos-based image encryption algorithms with permutation-diffusion structure are broken recently [9-15]. The main reason is that the keystream used in diffusion step only depends on the key [16]. If the key keeps unchanged, the keystreams generated to encrypt different plaintexts are the same. Therefore the attackers can obtain the keystream by known-plaintext attack and chosen-plaintext attack [10, 12-13]. Correspondingly, the encryption scheme degenerates to permutation-only architecture which has already been broken [17].

To improve the security of encryption algorithm, some researchers recently proposed that the key stream in diffusion should be relevant to the plaintext [12]. They use the plaintext to control the iteration times of chaotic system, so the generated key stream is different if the plaintext is different, even though the key is the same. This scheme is successful to resist the known-plaintext attack and chosen-plaintext attack. Nevertheless, there still exist some possible problems. First, it is well known that one-dimensional chaotic map has periodic problem when implemented in finite precision, which may result in consequent degradation in security [16]. And the key space of low dimensional chaotic map is small. Second, the necessary iteration times of chaotic system in diffusion process increases largely. In addition, the P-box in permutation process is irrelevant to the plaintext. In this paper, a novel image encryption scheme based on even-symmetric chaotic maps is proposed. In the permutation process, the iteration time of even-symmetric chaotic map to avoid transient effect is not fixed but relevant to the plaintext so the P-box changes even though the initial value is the same. In the diffusion process, two even-symmetric chaotic maps are used to generate the key stream. The pixels in the permuted-image determine which even-symmetric chaotic map to iterate for next byte in the keystream each time instead of increasing the iteration times. Meanwhile, the problems of short period and small key space in one-dimension chaotic map are practically avoided by using two chaotic systems. In this paper, we use the even-symmetric chaotic map for its good statistical properties [18-19]. Other one-dimension chaotic maps such as skew tent map are also applicable for our scheme. This method can also be extended to other fields such as text encryption since the algorithm treats the image as a vector.

The remainder of this paper is organized as follows. Section 2 briefly reviews the even-symmetric chaotic system. Section 3 presents the encryption and decryption scheme. Section 4 analyses performance and security of the scheme is. Section 5 finally summarizes and concludes this paper.

978-0-7695-5044-2/13 $26.00 © 2013 IEEE
DOI 10.1109/EIDWT.2013.50

II. EVEN-SYMMETRIC CHAOTIC MAP

If both the map $F(x)$ and its unique invariant measure density $f_F^*(x)$ satisfy the even-symmetric properties [18]:

$$\begin{cases} F(a+e-x)=F(x) \\ f_F^*(a+e-x)=f_F^*(x) \end{cases} \tag{1}$$

And the binary function $H(x)$ satisfies the symmetric condition

$$H(a+e-x)=1-H(x), \quad x\in[a,e] \tag{2}$$

Then $H(F(x))$ can be called an Even-symmetric Chaotic Map (ESCM).

We followed the practical method in [19] to construct an ESCM by the following two steps.

Step1.Construct a piecewise-linear map $T(z)$ as formula (3), d is a parameter:

$$T(z)=\begin{cases} \dfrac{z}{d}, \text{if } z\in[0,d) \\ \dfrac{z-d}{0.5-d}, \text{if } z\in[d,0.5] \\ \dfrac{1-z-d}{0.5-d}, \text{if } z\in(0.5,1-d] \\ \dfrac{1-z}{d}, \text{if } z\in(1-d,1] \end{cases} \tag{3}$$

Step2. Obtain real number by the function $F(x)$:

$$q(x)=\int_0^x 3(2x-1)^2\,dx=4x^3-6x^2+3x\ , x\in(0,1) \tag{4}$$

$$q^{-1}(x)=0.5+0.5\times(2x-1)^{1/3} \tag{5}$$

$$F(x)=q^{-1}\{T[q(x)]\} \tag{6}$$

Step3. Apply $H(x)$ to get binary number:

$$H(F(x))=\begin{cases} 0, & \text{if } F(x)\in[0,0.378)\cup[0.5,1-0.622) \\ 1, & \text{if } F(x)\in[0.378,0.5)\cup[0.622,1] \end{cases} \tag{7}$$

The good statistical properties of ESCM satisfy the requirement of key stream generator in image encryption application.

In ESCM, the binary sequence $\{H(x_t)\}_{t=1}^{+\infty}$ with $x_{i+1}=F(x_i)\,(i=1,2,\cdots)$ satisfies the following good statistical properties [18, 19]:

(1) $\lim_{J\to+\infty}(1/J)\sum_{t=1}^{J}H(x_t)=0.5$, which means that the number of 0 and 1 in $\{H(x_t)\}_{t=1}^{+\infty}$ are almost same.

(2) $\{H(x_t)\}_{t=1}^{+\infty}$ has $\delta-like$ auto correlation function:

$\rho_F(r)=\lim_{J\to+\infty}(1/J)\sum_{t=1}^{J}(-1)^{H(x_t)}(-1)^{H(x_{t+r})}=\delta(r), r\ge 0$.

(3) $\{H(x_t)\}_{t=1}^{+\infty}$ is a binary Bernoulli (BB) sequence, i.e., a sequence of independent and identically distributed binary random variables.

III. IMAGE ENCRYPTION SCHEME BASED ON MULTIPLE CHAOTIC MAP

A. Permutation operator using ESCM

A 256 gray-scale image of size can be presented by a one-dimension vector. An even-symmetric chaotic map as constructed in section Ⅱ is used to permute the plain image. The detailed permutation process can be described as follows:

Step1. Set a real-value parameter $y_1\in(0,1)$.

Step2. Get the initial condition of ESCM

$$x_0=\frac{(p_i+1)\times y_1}{256} \tag{8}$$

Step3. Iterate $x_{i+1}=F(x_i)$ for L times to avoid transient effect where $L=200$.

Step4. Continue to iterate $x_{i+1}=F_1(x_i)$ for $M\times N$ times and get a real-value sequence

$$X=\{x_1,x_2,\cdots,x_{M\times N}\} \tag{9}$$

Step5. Keep the position of x_j unchanged and sort the sequence $\{x_1,x_2,\cdots,x_{j-1},x_{j+1},\cdots,x_{M\times N}\}$ in ascending order to obtain a new sequence

$$X'=\{x'_1,x'_2,\cdots,x'_{j-1},x_j,x'_{j+1},\cdots x'_{M\times N}\} \tag{10}$$

Step6. Find the positions of $\{x'_1,x'_2,\cdots,x'_{M\times N}\}$ in $\{x_1,x_2,\cdots,x_{M\times N}\}$ and denote them as $T=\{t_1,t_2,\cdots,t_{M\times N}\}$ where $x_{t_i}=x'_i$.

Step7. Shuffle $P=\{p_1,p_2,\cdots,p_{M\times N}\}$ by using T as the P-box and get $P'=\{p'_1,p'_2,\cdots,p'_{M\times N}\}$ such that $p'_i=p_{t_i}$ and $p'_j=p_j$.

B. The encryption and decryption scheme

Two even-symmetric chaotic systems with different control parameters are used to generate the key stream and encrypt the plain-image in this scheme. In the procedure, we use four parameters (y_1,y_2,y_3,p) and denote two ESCMs as $H(F(x))$ and $H'(F'(x))$ respectively. In the encryption process, initial values and control parameters of two ESCM systems are used as secret key and determine the key stream together with plain image.

The detailed encryption procedure includes two rounds. The first round is described as follows:

Step1. Use the permutation method as described above to get the shuffled image $P'=\{p'_1,p'_2,\cdots,p'_{M\times N}\}$.

Step2. Calculate $j = \mathrm{mod}(\mathrm{floor}(y_2 \times 2^{52}), M \times N)$ and $x_0 = \frac{(p_j + 1) \times y_1}{256}, x_0' = \frac{(p_j + 1) \times y_2}{256}$.

Step3. Iterate $H(F(x))$ for 8 times and obtain 8 bits denoted as b_1.

Step4. Calculate the corresponding pixel value D_i by using the currently operated pixel p_i', the previous pixel of plain image p_{i-1} and b_i $(i = 1,2,\cdots, M \times N)$. The computing formula is:

$$D_i = \begin{cases} p_i' \oplus \mathrm{mod}(p_i + b_i, 2^8), & \text{if } i = 1 \\ D_{i-1} \oplus p_i' \oplus \mathrm{mod}(\mathrm{floor}(y_3 \times 2^{48}), 2^8), & \text{if } i = j \\ p_i' \oplus \mathrm{mod}(D_{i-1} + b_i, 2^8), & \text{else} \end{cases} \quad (11)$$

Here $\oplus$ is bitwise XOR operator, y_3 is the parameter.

So the inverse formula of Eq. (11) for p_i' can be described as:

$$p_i' = \begin{cases} D_i \oplus \mathrm{mod}(p_i + b_i, 2^8), & \text{if } i = 1 \\ D_i \oplus D_{i-1} \oplus \mathrm{mod}(\mathrm{floor}(y_3 \times 2^{48}), 2^8), & \text{if } i = j \\ D_i \oplus \mathrm{mod}(D_{i-1} + b_i, 2^8), & \text{else} \end{cases} \quad (12)$$

Step5. Calculate $v_i = D_i \bmod 2$ and use the value of v_i to decide which even-symmetric chaotic system is applied to generate the next element in key stream $B = \{b_1, b_2, \cdots, b_{M \times N}\}$:

(a) If $v_i = 0$, iterate $H(F(x))$ for 8 times to obtain b_{i+1};

(b) If $v_i = 1$, iterate $H'(F'(x))$ for 8 times to obtain b_{i+1}.

Step6. Let $i = i + 1$, return to step 4 until i reaches $M \times N$.

Now we get the result of first encryption round $D = \{D_1, D_2, \cdots, D_{MN}\}$. To improve the security, skew tent map is used in the second round:

Step1. Iterate skew tent map as described in formula (13) with initial value $z_0 = y_3$ for L times to get ride of transient effect.

$$G(z) = \begin{cases} z / q, & \text{if } z \in [0, q) \\ (1 - z) / (1 - q), & \text{if } z \in (q, 1] \end{cases} \quad (13)$$

Step2. Continue to iterate skew tent map for $M \times N$ times to obtain a sequence $Z = \{z_1, z_2, \cdots, z_{M \times N}\}$. Then calculate the pixel value of cipher image as following formulas:

$$\phi_i = \mathrm{floor}(256 \times z_i) \quad (14)$$

$$c_i = c_{i-1} \oplus \phi_i \oplus \mathrm{mod}(D_i + \phi_i, 2^8) \quad (15)$$

Here c_0 is a given constant and $\{c_1, c_2, \cdots, c_{M \times N}\}$ is the final cipher vector.

The inverse formula can be described as following:

$$\alpha = c_i \oplus c_{i-1} \oplus \phi_i \quad (16)$$

$$D_i = \begin{cases} \alpha - \phi_i & \text{if } \alpha \geq \phi_i \\ 256 + \alpha - \phi_i & \text{if } \alpha < \phi \end{cases} \quad (17)$$

The decryption procedure includes the following steps which is similar to the encryption steps:

Step1. Iterate Eq. (13) and (14) to get $\{\phi_1, \phi_2, \cdots, \phi_{M \times N}\}$.

Step2. Calculate $D = \{D_1, D_2, \cdots, D_{M \times N}\}$ by Eq. (17).

Step3. Use D to obtain $P' = \{p_1', p_2', \cdots, p_{M \times N}'\}$ by Eq. (12).

Step4. Calculate the initial condition x_0 by p_j' which equals to p_j and regenerate the P-box T as described in section A.

Step5. Remove the effect of permutation from P' by performing the reverse operation of permutation with the P-box T. So we get the plaintext P.

IV. SECURITY ANALYSIS

A. Key space analysis

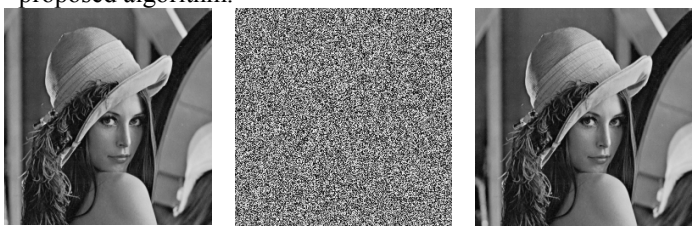
Key space size is the total number of different keys which can be used in the encryption method [16]. A good encryption algorithm should have a large key space to resist brute-force attack.

The encryption scheme proposed in section Ⅲ uses two different even-symmetric chaotic map and the skew tent map to generate the key stream. So the keys include the parameters in ESCM and skew tent map $d, d' \in (0,0.5)$, $y_1, y_2, y_3 \in (0,1)$, $q \in (0,1)$ and $j \in \{1,2,\cdots, M \times N\}$.

Since the computational precision of 64-bit double-precision number is 2^{-52} according to the IEEE floating-point standard, the key space of our scheme reaches $2^{52 \times 5 + 26 \times 2} + M \times N$. This key space size is large enough to make brute-force attack unfeasible.

B. Statistical analysis

To resist statistical attack, the proposed encryption scheme uses both permutation and diffusion operations. In this subsection, histogram, correlation of adjacent pixels and information entropy are computed to evaluate the statistical performance of our encryption algorithm. Figure.1 shows the plain image "lena" and the encrypted image using the proposed algorithm.

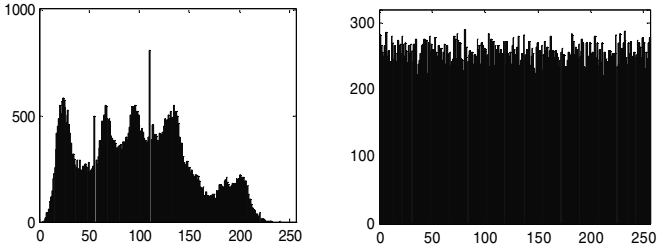


(a) (b) (c)

Figure 1. (a) plain-image (b) cipher-image (c) decryption image.

(1) Histogram

The histogram of an image illustrates that how pixels are distributed by plotting the number of pixels at each grayscale level. A good encryption method should hide the distributing character of the plain-image and avoid leaking any information. So the ideal histogram of a cipher image should be uniformly distributed. Fig.2 compares the histograms of plain-image and cipher-image using the proposed algorithm. The result demonstrates that the histogram of the encrypted image is nearly uniform distributed.



(a) (b)

Figure 2. Histogram of (a) plain-image (b) cipher-image

(2) Correlation of two adjacent pixels

High correlation of adjacent pixels always exists in an ordinary image which has definite visual content. So a secure image encryption method should remove this weak feature to produce a cipher-image with sufficiently low correlation of adjacent pixels, which improves the resistance against statistical attack. Fig.3 illustrates the distributing of adjacent pixels in the plain-image "lena" and cipher-image. It shows that the strong correlation between adjacent pixels in plain-image is greatly reduced after encrypted.

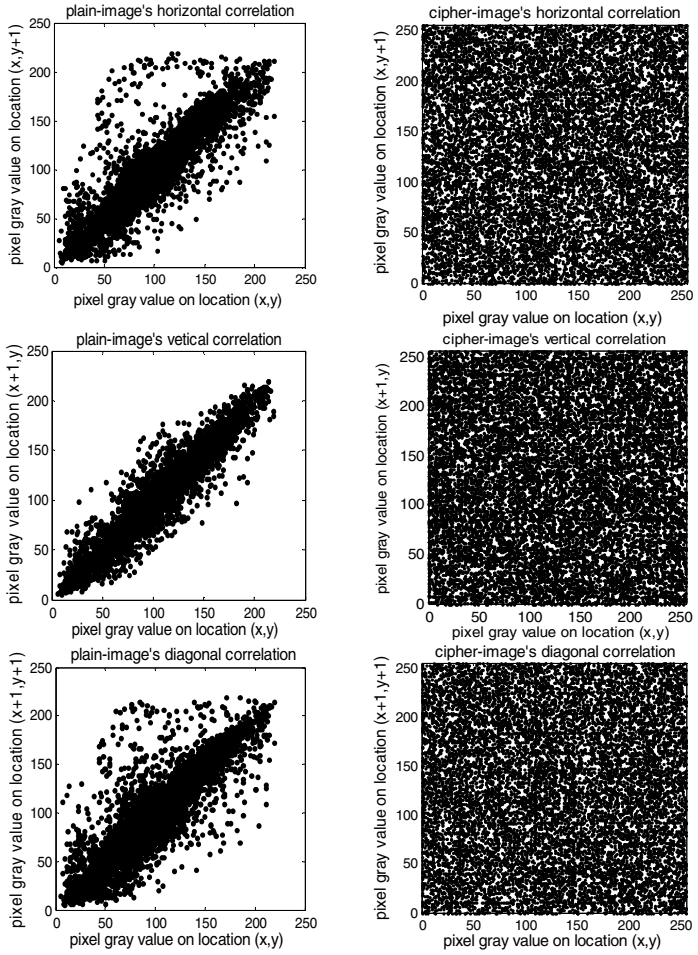


Figure 3. Correlation of the plain-image and cipher-image

(3) Information entropy

In information theory, information entropy is the most important feature of unpredictability and randomness. To calculate the entropy $H(s)$ of message source s, we have

$$H(s)=\sum_{i=0}^{2^N-1} p(s_i)\log_2\frac{1}{p(s_i)} \tag{18}$$

where s_i denotes the symbol in the message s, N is the number of bits to represent a symbol s_i, 2^N is the total number of symbols, and $p(s_i)$ represents the probability of s_i in s. For a true random source emitting 2^N symbols, the information entropy $H(s)$ should be N. Therefore the theoretical entropy value for a 256 gray-scale image is 8.

In Table 1, the information entropy of cipher images are computed, and the results are very close to the theoretical value 8. This means that information leakage in the encryption process is negligible and the encryption system is secure against entropy attack.

Table 1 Information entropy of cipher-image

Images	Entropy of cipher-image	Entropy of plain-image
Lena	7.9976	7.5489
Airplane	7.9977	7.3735
Baboon	7.9971	7.0275

C. *Sensitivity analysis*

To resist brute-force attack or differential attack, a good encryption method should be sensitive to the key and the plain image. Two most common measure are *NPCR* (number of pixels change rage) and *UACI* (unified average changing intensity) which test the different range between two images. If $c_1(i,j)$ and $c_2(i,j)$ denotes two images $(1 \le i \le M, 1 \le j \le N)$, *NPCR* and *UACI* can be calculated as follows:

$$NPCR = \sum_{i,j} D(i,j) / (W \times H) \times 100\% \quad (19)$$

$$UACI = [\sum_{i,j} |c_1(i,j) - c_2(i,j)| / 255] / (W \times H) \times 100\% \quad (20)$$

If $c_1(i,j) = c_2(i,j)$, then $D(i,j) = 0$; otherwise, $D(i,j) = 1$.

(1) Key sensitivity

Key sensitivity of a cryptosystem can be observed in this way: completely different cipher image are produced with tiny different keys.

We encrypt the plain image "Lena" using the key $d = 0.21, d' = 0.13, y_1 = 0.912487, y_2 = 0.536192, y_3 = 0.753418, q = 0.57$ and slightly different key which just changes 10^{-10}. The result shows that more than 99% pixels in cipher-image change their gray value. Thus, a high key sensitivity is provided by the proposed method.

(2) Plaintext sensitivity

Plaintext sensitivity means that a tiny change in plaintext can cause a substantial change in cipher-image. A good encryption method should be sensitive to plaintext to resist differential attack. We randomly choose one pixel of the plain-images (Lena, Airplane and Baboon) and change its value, then calculate *NPCR* and *UACI* between the cipher-images. From Table 2, we can see that one pixel different in the plain image can cause more than 99% pixel change in the cipher images, and the *UACI* values are desirable [16].

Table 2 *NPCR* and *UACI* between cipher-images with slightly different plain-image

Cipher-images	*NPCR* (%)	*UACI* (%)
Lena	99.60	33.53
Airplane	99.58	33.60
Baboon	99.62	33.45

D. *Resistance to known-plaintext and chosen-plaintext attack*

From the encryption scheme, we see that a feedback from plain image is concerned with determining which even-symmetric chaotic system to iterate for next byte of the key stream. Thus, when different plain-images are encrypted, the corresponding key streams and cipher images are not same even though the same key is used. The attackers cannot obtain any useful information by encrypting some special images since the result is related to those chosen images.

V. CONCLUSION

This paper proposed a novel image encryption method based on even-symmetric chaotic maps and skew tent map. The method employs the permutation-diffusion architecture and uses two even-symmetric chaotic maps to shuffle and diffuse the pixels of plain image. In the scheme, the key space is large enough to resist brute-force attacks and the good statistical properties protect the cipher image from statistical attack. Meanwhile, the proposed scheme owns high sensitivity to key and plaintext. Moreover, the key stream and cipher image are related to the key and the plain image, so the method is able to resist known-plaintext and chosen-plaintext attack. Results are encouraging and suggest that the scheme is reliable to be adopted for the secure image communication application.

ACKNOWLEDGE

This work was supported by the Natural Science Foundation of Guangdong Province (Grant no. 8151064101000033).

REFERENCES

[1] K. Wang, W. Pei, et al, "On the security of 3D Cat map based symmetric image encryption scheme", Physics Letters A, Vol 343, 2005, pp. 432–439.

[2] N.K. Pareek, V. Patidar and K.K. Sud, "Image encryption using chaotic logistic map", Image and Vision Computing, Vol 24, 2006, pp. 926–934.

[3] H.S. Kwok, K.S. Tang , "A fast image encryption system based on chaotic maps with finite precision representation", Chaos, Solitons and Fractals, Vol 32, 2007, pp. 1518−1529.

[4] A.N. Pisarchik, M. Zanin, "Image encryption with chaotically coupled chaotic maps", Physica D, Vol 237, 2008, pp. 2638–2648.

[5] Y. Wang, K. Wong, et al, "A new chaos-based fast image encryption algorithm", Applied Soft Computing, Vol 11, 2011, pp/ 514–522.

[6] X. Tong , M. Cui, "Image encryption with compound chaotic sequence cipher shifting dynamically", Image and Vision Computing, Vol 26, 2008, pp. 843−850.

[7] M. Amin, S.Osama, et al, "A chaotic block cipher algorithm for image cryptosystems,", Commun Nonlinear Sci Numer Simulat, Vol 15, 2010, pp. 3484–3497.

[8] C.K. Huang, H.H. Nien, "Multi chaotic systems based pixel shuffle for image encryption", Optics Communications,Vol 282, 2009, pp. 2123–2127.

[9] C. Li, S.Li, et al, "On the security defects of an image encryption scheme", Image and Vision Computing, Vol 27, 2009, pp. 1371−1381.

[10] C. Li, S. Li, et al, "Cryptanalysis of an image encryption scheme based on a compound chaotic sequence", Image and Vision Computing , Vol 27, 2009, pp. 1035−1039.

[11] G. Alvarez, S. Li, "Cryptanalyzing a nonlinear chaotic algorithm (NCA) for image encryption", Commun Nonlinear Sci Numer Simulat , Vol 14, 2009, 3743-3749.

[12] X.Di, L. X. feng and P. Wei, "Analysis and improvement of a chaos-based image encryption algorithm" , Chaos, Solitons & Fractals, Vol 40 , 2009, pp. 2191-2199.

[13] R. Rhouma, E . Solak and S . Belghith, "Cryptanalysis of a new substitution-diffusion based image cipher", Communications in Nonlinear Science and Numerical Simulation , Vol 15, 2010, pp. 1887-1892.

[14] D. Xiao, F.Y. Shih, "Using the self-synchronizing method to improve security of the multi chaotic systems-based image encryption", Optics Communications, Vol 283, 2010, pp. 3030–3036.

[15] E. Solak, R. Rhouma and S. Belghith, "Cryptanalysis of a multi-chaotic systems based image cryptosystem", Optics Communications, Vol 283, 2010, pp. 232–236.

[16] G.J. Zhang, Q. Liu, "A novel image encryption method based on total shuffling scheme", Optics Communications, Vol 284, 2011, pp. 2775−2780.

[17] Li C., K.-T. Lo, S. Belghith, "Optimal quantitative cryptanalysis of permutation-only multimedia ciphers against plaintext attacks", Signal Processing, Vol 91, 2011, pp. 949.

[18] T. Kohda, "Statistics of chaotic binary sequences", IEEE Transaction on Information Theory, Vol 43, 1997, pp. 104-112.

[19] T. Sang, R. Wang, Y. Yan, "Generating binary bernoulli sequences based on a class of Even-symmetric chaotic map", IEEE Transaction on Communications, Vol 49, 2001, pp. 620-623.