Multimed Tools Appl (2014) 70:1559–1584
DOI 10.1007/s11042-012-1173-2

A new image encryption scheme based on cyclic elliptic curve and chaotic system

Ahmed A. Abd El-Latif · Li Li · Xiamu Niu

Published online: 28 July 2012

Abstract Recently, several cryptosystems based on chaos have been proposed. Nevertheless, most of them hinder the system performance, security, and suffer from the small key space problem. This paper introduces an efficient symmetric encryption scheme for secure digital images based on a cyclic elliptic curve and chaotic system that can overcome these disadvantages. The cipher encrypts 256-bit of plainimage to 256-bit of cipherimage within eight 32-bit registers. The scheme generates pseudorandom bit sequences for round keys based on a piecewise nonlinear chaotic map. Then, the generated sequences are mixed with the key sequences derived from the cyclic elliptic curve points. Results of statistical and differential analysis demonstrate that the proposed algorithm has adequate security for the confidentiality of digital images. Furthermore, it has key sensitivity together with a large key space and the encryption is fast compared to other competitive algorithms.

Keywords Image encryption · Chaotic system · Cryptographic primitive operations · Cyclic elliptic curve

1 Introduction

Along with the rapid development of Internet and universal application of multimedia technology, media data has been transmitted over insecure channels. In particular, the use of images is an ascending need because it is the main data information provided by most of the advanced sensors of today like infrared cameras, optical cameras, millimeter wave cameras, radar imagers, x-ray imagers, etc.

Image encryption is the process of realigning the original image into an incomprehensible or unintelligible one that is non-recognizable in appearance, disorderly and

A. A. A. El-Latif (✉) · X. Niu
School of Computer Science and Technology, Harbin Institute of Technolog, Harbin 150080, China
e-mail: ahmed_rahiem@yahoo.com

L. Li
School of Computer Science and Technology, Harbin Institute of Technology,
Shenzhen Graduate School, Shenzhen 518055, China

A. A. A. El-Latif
Department of Mathematics, Faculty of Science, Menoufia University, Shebin El-Koom 32511, Egypt

unsystematic [8, 17]. In recent years, various encryption algorithms have been proposed and widely used, such as DES, IDEA or AES. However, these encryption schemes have been invented to text encryption and appear not to be ideal for image applications due to some intrinsic features of images such as bulk data capacity, high correlation between pixels and high redundancy, which are troublesome for traditional encryption [15]. Recently, chaos theory has received ever increasing research interests from cryptographers. Based on chaotic systems several image encryption algorithms have been studied, which can be classified into two types, i.e., stream cipher and block cipher. The first one is carried out by applying the same transformation on individual data bits and using chaos to generate pseudorandom sequences [17]. The security of stream ciphers depends on the sequences' randomness. It is noted, however, that the real random sequence is still difficult to generate in practice, which restricts their applications. The second one makes use of chaotic system properties to encrypt the media data block by block. The security of block ciphers depends on their computing complexity, which is determined by the confusion and diffusion criteria [30].

A short overview of the main recently proposed chaos-based cipher schemes are given hereafter. In [5], Chen et al. have proposed an image encryption algorithm in which a two-dimensional chaotic map is generalized to three-dimension for designing a real-time secure image encryption scheme. This approach used the three-dimensional cat map to shuffle the position of the image pixels and uses another chaotic map to confuse the relationship between the original and encrypted images. Further, Mao et al. [18] extended the same idea with the 3D chaotic baker's map at the substitution stage instead of the 3D cat map. Guan et al. [10] used the 2D cat map to shuffle the position of the image pixels in the spatial domain and the output of a discretized Chen's system is used to mask the pixel values. In [20], Pareek et al. have proposed an approach for image encryption based on chaotic logistic maps. An external secret key of 80-bit length and two chaotic logistic maps are employed. The initial conditions for the logistic maps are derived by using the external secret key. Eight different operation-types are used to encrypt the image pixels. In [9] Gao et al. have presented an image encryption scheme, which used an image total shuffling matrix to shuffle the position of image pixels and then utilized a hyper-chaotic system to confuse the relationship between the plainimage and the cipherimage. Recently, Lian [16] has proposed an encryption scheme for images based on spatio-temporal chaos system. The spatiotemporal lattices are used to generate pseudorandom sequences. Then, the sequences are used to encrypt the selected parameters in each image block.

More recently, Patidar et al. [22] (hereafter referred as PPS09) proposed a lossless image cipher based on substitution–diffusion architecture using chaotic standard and logistic maps. It is specifically designed for colour images. They used three processes to encrypt the colour images, confusion-diffusion-confusion. Two kinds of confusion are used: confusion using the secret key (via XORing keys) and confusion using the chaotic standard and logistic maps. Soon after the proposal of PPS09, its algebraic description was analyzed and a drawback of its structure was shown by Rhouma et al. [26]. After that, Patidar et al. [21] modified the original scheme and claimed that the modified scheme is secure against Rhouma et al.'s attack [26]. Very recently, Li et al. [14] found that Patidar et al.'s scheme [21] is still insecure against a known/chosen-plaintext attack, which can break the original scheme in [22]. In addition, extra security weaknesses of both the original and the modified image encryption schemes are reported: insufficient randomness of pseudo randomness and insufficient sensitivity with respect to a change of the plainimage.

Summarily, most of these schemes encounter some problems such as the lack of efficiency and security [13, 24, 25, 33]. This is mainly due to the fact that, poor diffusion operation leads to weaknesses against a differential attack, analytical floating-point

computation and small key space leading to slow performance speed, which makes it difficult to promote these chaotic encryptions into practical service. Table 1 gives a brief overview on some chaotic-based image encryption schemes and their properties as well as their weaknesses. Accordingly, designing a good cryptosystem needs further improvement to enhance their security. In this paper, a new image encryption scheme based on the cyclic elliptic curve and chaotic system is introduced. The new algorithm generates pseudorandom bit sequences for round keys based on adopting the chaotic system XORed with the key sequence derived from cyclic elliptic curve points. This, in turn, will increase the nonlinearity and the randomness of the round keys used for encryption/decryption. The philosophy of the well-known block cipher RC6 is to exploit operations (such as rotations and integer multiplication) that are efficiently implemented in modern processors. The proposed algorithm continues this trend as well as performing four rotations per round rather than the two found in RC6 and use more bits of data to determine the rotation amounts per round. Simulation results and analysis confirm that the proposed scheme has more superior performance than other algorithms such as RC5, RC6 and other competitive algorithms. In addition, it has high enough key space to resist any brute force attack.

This paper is organized as follows. In Section 2, we present some preliminaries about designing the keystream for encryption/decryption. The proposed algorithm is described in Section 3. Performance and security analysis are reported in Section 4. Section 5 summarizes the most important findings of this paper and the conclusion is drawn in Section 6.

2 Preliminaries

In most cryptosystems, the cryptographic key is a significant part. No matter how strong and how well designed the encryption algorithm might be, if the key is poorly chosen or the key space is too small, the cryptosystem will be easily broken. Due to this principle, a chaotic system and the cyclic elliptic curve are chosen because of their properties and easy

Table 1 Chaotic based- image encryption schemes and their properties

Algorithm	Strengths	Weakness and known attacks
Mao *et al.*'s [18]	Adequate key space, good key sensitivity	Insensitivity to changes in plaintext and key stream, poor diffusion function. [33]
Pareek *et al.*'s [20]	Good confusion and diffusion	Weaknesses in the key. Weak to chosen and known plaintext attacks. [13]
Gao *et al.*'s [9]	Adequate key space	Weak to chosen and known plaintext attacks. [24]
Lian's [16]	Unapparent change in the compression ratio and less increase in computational cost compared with video compression.	Weaknesses in the key. Weak to chosen and known plaintext attacks . [25]
Patidar *et al.*'s [22]	Simple mixing and diffusing operations	Weak to chosen and known plaintext attacks. [26]
Patidar *et al.*'s [21]	Good confusion and diffusion	Weak to chosen and known plaintext attacks. [14]

implementation. In what follows, there are some preliminaries about designing the proposed round keys for encryption and decryption.

2.1 Cyclic elliptic curve based pseudorandom bit generator

Elliptic curves, which are not directly related to ellipses, are cubic equations in two variables, x and y, with coefficients from Galois finite field $GF(2^m)$ satisfying certain conditions [6]. The general equation of an elliptic curve is

$$y^2 + xy = x^3 + \alpha x^2 + \beta \quad , \alpha, \beta \in GF(2^m) \quad where \quad 4\alpha x^3 + 27\beta^2 \neq 0 \tag{1}$$

In the above equation, each value of α and β gives a different elliptic curve. All points (P with co-ordinates (x_p, y_p)) which satisfies the above equation plus a point at infinity O lies on the elliptic curve. The total number of points on the elliptic curve along with the point at infinity O ($x=\infty$; $y=\infty$) is called the order of the elliptic curve denoted by M. Least integer N for which NP is equal to point at infinity O ($NP=O$) is called the order of point P such that N≤M. Then, P, 2P … (N−1)P are distinct points on the elliptic curve. For a certain choice of α and β it is possible to choose a base point P of highest order N=M which is square free (square root of M is not an integer) [7]. Further, P, 2P, 3P …MP are the M points of the elliptic curve, where MP is the point at infinity. Such an elliptic curve is called a cyclic elliptic curve [19]. For the property of easy implementation and good statistical properties, a Linear Feedback Shift Register (LFSR) is used for generating a sequence of integers $\{k_i\}$ modulo p where p is a prime and p≥M. Every element in the sequence $\{k_i\}$ is mapped to k_iP that is a point of the cyclic elliptic curve with co-ordinates say (x_i, y_i) as shown in Fig. 1. The sequence $\{k_iP\}$ is a random sequence of elliptic curve points. From the sequence (x_i, y_i) several binary and non-binary sequences can be derived and their randomness properties are investigated in [28]. Choosing a linear feedback with connection polynomial primitive over $GF(p)$, can generate periodic sequence with maximum period [11, 28]. For any choice of n-stages >1, an appropriate feedback connection can be obtained by using an $n-th$ degree primitive polynomial over $GF(p)$. It can be shown under this condition; the sequence $\{k_i\}$ is periodic (with all initial values $k_{n-1}, k_{n-2}...k_1, k_0$ not zero) and is of period p^n-1. If the cyclic elliptic curve is chosen, the random sequence $\{k_iP\}$ covers all points in the elliptic curve and can be used for encryption and decryption [29].

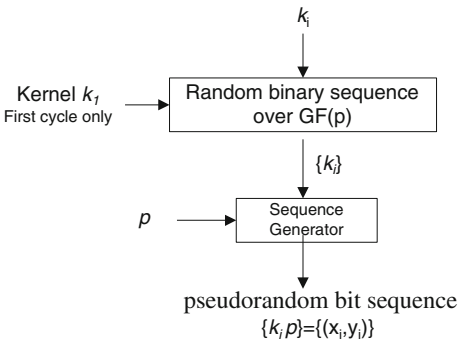


Fig. 1 Pseudorandom bit generator of the cyclic elliptic curve

2.2 Chaotic system

In our proposed encryption scheme, we use 2-D piecewise nonlinear chaotic map described as Eq. (2). The researches have shown that it has a uniform distribution (see Refs. [1, 12] for the detail)

$$\phi_2^{(2)}(x_1,\lambda_1) \Rightarrow x_1(n+1) = \frac{2{\lambda_1}^2(0.5-|x_1(n)-0.5|)}{1+2\left({\lambda_1}^2-1\right)(0.5-|x_1(n)-0.5|)}, \tag{2}$$

Where λ_1 and x_1 are system parameters. The map $\phi_2^{(2)}(x_1,\lambda_1)$ is reduced to tent map if λ_1=1.

3 The proposed algorithm

Here, we propose a new symmetric image encryption algorithm based on a cyclic elliptic curve and piecewise nonlinear chaotic map. In this algorithm, we incorporate the merits of the cyclic elliptic curve with the chaotic system as well as the cryptographic primitive operations to strengthen the round keys for encryption and to enlarge the key space required to resist brute force attacks. Furthermore, we strengthen the rate of diffusion process by simple steps during the round: rotation, integer multiplication, the quadratic function, and fixed bit shifting by five bits, which is a secure way against both linear and differential attacks. In turn, the diffusion achieved per round is significantly increased.

The proposed scheme is a fully parameterized family of encryption algorithms. A version of the proposed encryption algorithm is more accurately specified as (Block-based Image Encryption Algorithm) BEA-*w*/*r*/*b* where the word size is *w* bits, encryption consists of number of rounds *r*, and *b* denotes the length of the encryption key in bytes. These three parameters are as in Table 2.

The proposed BEA-*w*/*r*/*b* works with eight *w*-bit words as input (plainimage) block size and eight *w*-bit words as (cipherimage) output block size. It consists of three components: a key expansion algorithm, an encryption algorithm, and a decryption algorithm. These algorithms use primitive operations and it uses *lg*(*x*) to denote the base-two logarithm of *x* as shown in Table 3.

3.1 Key expansion scheme

The key expansion consists of three algorithmic steps: conversion, initialization, and mixing as shown in Fig. 2.

The conversion of the proposed scheme is practically identical to the conversion of RC6 [27], copy user's secret key *K* [*0*…*b*-1] into an array *L* [*0*….*c*-1] with words *c*=[*b*/*u*], where *u*=*w*/8 is the number of bytes/word. This operation is done in a natural manner, using *u* consecutive key bytes of *K* to fill up each successive word in *L*, low-order byte to high-order

Table 2 Summery of BEA- *w*/*r*/*b* parameters

Parameters	Definition	Values
w	Word size in bits	16,32,64
r	Number of rounds	0,1,2,….,255
b	No. of bytes in secret key	0,1,2,….,255

Table 3 Primitive operations of the proposed algorithm

Notation	Meaning
X+Y	Two's complement addition of words
X−Y	Two's complement subtraction of words
X⊕ Y	Bit-wise exclusive –OR of words
X<<<Y	The cyclic rotation of word X left by Y bits
X >>>Y	The cyclic rotation of word X right by Y bits
X* Y	Multiplication modulo 2^w

byte. In the initialization, we adopt the chaotic map, Eq (2), to generate chaotic sequences and then XORed it with the pseudorandom bits generated by the cyclic elliptic curve to initialize the expanded key table, array *S*. Finally, the user's secret key is mixed over the array *S* and *L*.

We use two functions *Map*1 (*x*) and *Map*2 (*x*). *Map*1 (*x*) maps a byte to [0, 1] interval, *Map*2 (*x*) maps the [0, 1] interval to a word, respectively. Namely, the function *Map*1 (x) maps an integer between 0 and 255 in the key space domain into a real number in the interval [0, 1] in the chaotic map domain. Then, the second *Map*2 (x) maps the domain of the chaotic map, the interval [0, 1], back into [0, 255]. Further, we use chop (*x*) to return *x* with the integer part, init_pad (*K*[*b*]) to calculate the initial pad from the user supplied secret key. Chaotic (*x*, iterations) means to evaluate the chaotic map starting from *x*, iteration times and finally *RSCEC* [i] is the random sequence of cyclic elliptic curve points. We note that, the key-expansion function has a certain amount of "one-wayness" to make it not so easy to determine *K* from *S*. In what follows, the key expansion algorithm is described as the following steps.

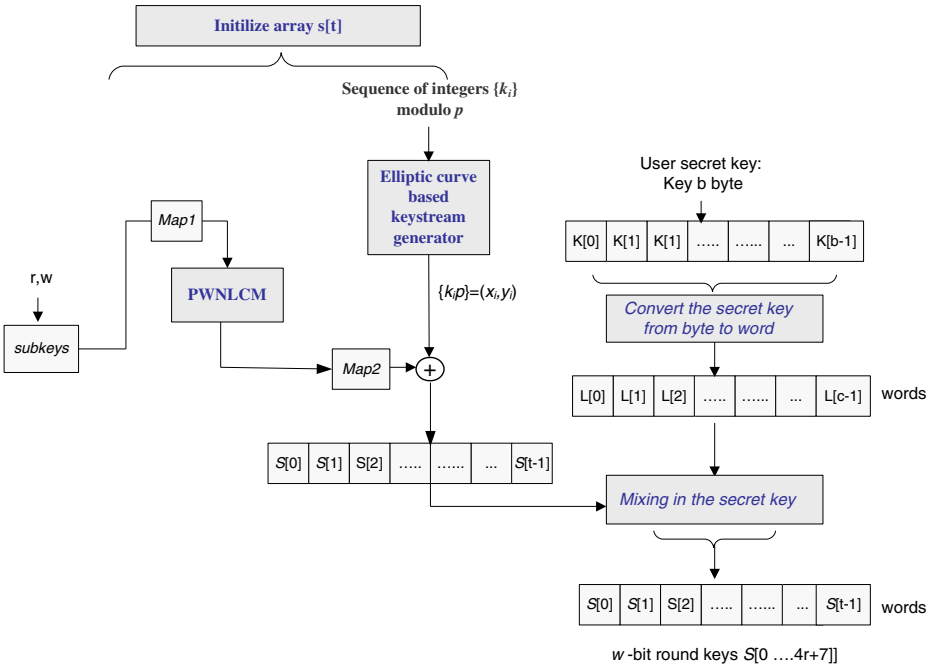


Fig. 2 The key expansion scheme

Algorithm: key expansion (b, r, $x_1(0)$, λ_1)

INPUT: user secret key (key b byte), number of rounds r, initial value $x_1(0)$, control parameter λ_1

OUTPUT: w-bit round keys S[0,…,4r+7]

Step 1 for (i=b−1,L[c−1]=0;i!=−1;i−−)
L[i/u]← (L[i/u]<<<8)+K[i];
Step 2 for(IV [0]=init_pad, i=1;i<t;i++)
IV [i] ← Chaotic (chop (*Map*1 (K[subkey] +pad), K [next_subkey (Subkey)]) +IV [i-1];
subkey ← next_subkey (subkey);
for(i=0;i< t;i++)
C[i] ← *Map*2 (IV[*i*]);
S[*i*] ←*C*[*i*] ⊕ *RSCEC*[i];
Step 3 for(X=Y=i=j=k=0;k<t;k++,i=(i+1) mod t, j=(j+1) mod c)

$$X \leftarrow S[i] \leftarrow (S[i] + X + Y) <<< \lg w;$$
$$X \leftarrow L[j] \leftarrow (L[j] + X + Y) <<< (X + Y);$$

3.2 Encryption scheme

The proposed encryption consists of eight w-bit registers P_i (i=1,2,…,8), which contain the initial input plainimage as well as the output cipherimage at the end of the encryption process. In the inner encryption process, the following quadratic function, Eq. (3), is used four times within each round, which is different from RC6 that is used only two times:

$$f(x) = x(2x + 1)\ (\mathrm{mod} 2^w) \tag{3}$$

The high-order bits of the above function are used to determine the rotation amounts used. The particular choice of this transformation function is followed by a left rotation by lg(w) bit positions (e.g. in the case of w=32, lg(w)=5). The transformed values of P_i (i=2, 4, 6, 8) are used to modify the registers P_i (i=1, 3, 5, 7), increasing the nonlinearity of the algorithm while not losing any entropy (since the transformation is a permutation). The fixed rotation by lg(w) bits plays an important role in complicating both linear and differential cryptanalysis. The proposed encryption is described below and an illustration is given in Fig. 3.

Algorithm: Encryption (P, *r*, S[0,…,4*r*+7])

INPUT: the plainimage P, number of rounds r, w-bit round keys S[0,…,4r+7]

OUTPUT: 256-bit cipherimage C stored in P_i (i=1,2,…,8)

Step 1 $P_2 \leftarrow P_2 + S[0]$, $P_4 \leftarrow P_4 + S[1]$, $P_6 \leftarrow P_6 + S[2]$, $P_8 \leftarrow P_8 + S[3]$
Step 2 for (i=1;i<=r;i++)

$$a \leftarrow (P_2(2\ P_2 + 1)) <<< \lg w; \qquad b \leftarrow (P_4(2\ P_4 + 1)) <<< \lg w;$$
$$c \leftarrow (P_6(2\ P_6 + 1)) <<< \lg w; \qquad d \leftarrow (P_8(2\ P_8 + 1)) <<< \lg w;$$
$$P_1 \leftarrow (P_1 \oplus a) <<< b + S[4i]; \qquad P_3 \leftarrow (P_3 \oplus b) <<< a + S[4i + 1];$$
$$P_5 \leftarrow (P_5 \oplus c) <<< d + S[4i + 2]; \quad P_7 \leftarrow (P_7 \oplus d) <<< c + S[4i + 3];$$

temp←P_1; $P_1 \leftarrow P_2$; $P_2 \leftarrow P_3$; $P_3 \leftarrow P_4$; $P_4 \leftarrow P_5$; $P_5 \leftarrow P_6$; $P_8 \leftarrow P_8$; $P_8 \leftarrow P_7$; $P_7 \leftarrow$ temp;

Step 3 $P_1 \leftarrow P_1 + S[4r+4]$; $P_3 \leftarrow P_3 + S[4r+5]$; $P_5 \leftarrow P_5 + S[4r+6]$; $P_7 \leftarrow P_7 + S[4r+7]$);

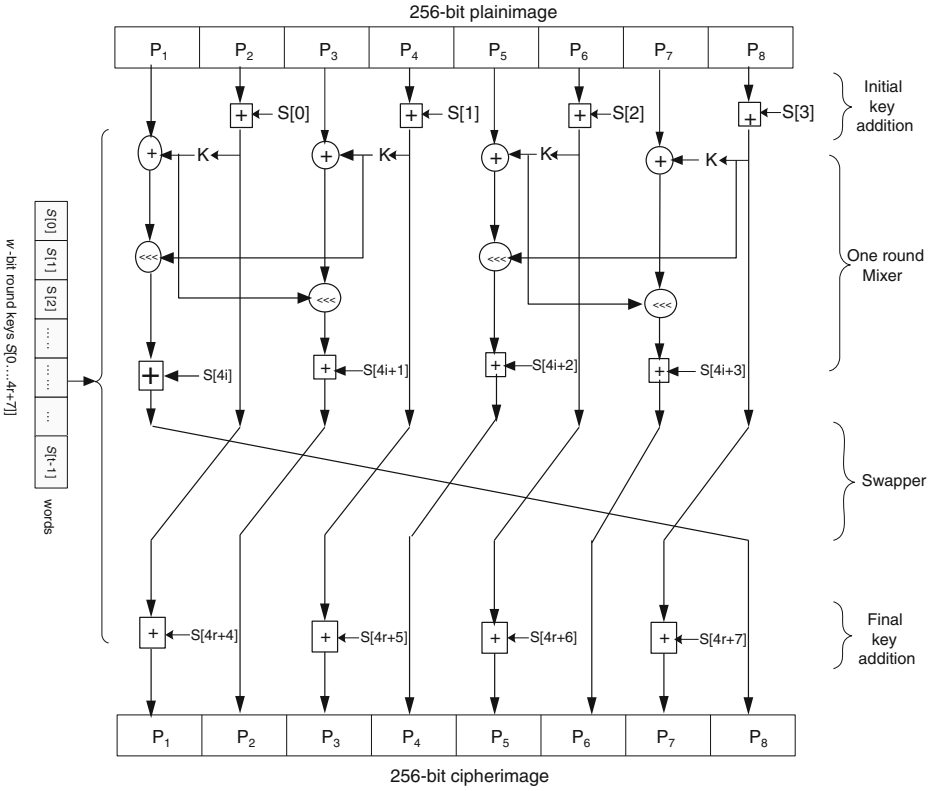


Fig. 3 Diagram of the encryption scheme. Here, $K \leftarrow (P_i(2P_i + 1)) <<< \lg w$

3.3 Decryption scheme

Decryption is the converse of encryption. At the receiver side, using the same round transformations and the same *w*, *r*, and *b*, the decryption can easily derive from the encryption routine as shown in Fig. 4 and through the following steps:

Algorithm: Decryption (C, *r*, S[0,…,4*r*+7]**)**

INPUT: the cipherimage C, number of rounds *r*, *w*-bit round keys S[0,…,4r+7]

OUTPUT: 256-bit plainimage P stored in P_i (i=1,2,…,8)

Step 1 $P_1 \leftarrow P_1$- S[4r+4]; $P_3 \leftarrow P_3$ -S[4r+5]; $P_5 \leftarrow P_5$ -S[4r+6]; $P_7 \leftarrow P_7$ -S[4r+7];

Step 2 for (i=r;i>0;i−−)

temp$\leftarrow P_7$; $P_7 \leftarrow P_8$; $P_8 \leftarrow P_6$; $P_6 \leftarrow P_5$; $P_5 \leftarrow P_4$; $P_4 \leftarrow P_3$; $P_3 \leftarrow P_2$; $P_2 \leftarrow P_1$; $P_1 \leftarrow$ temp;

$$a \leftarrow (P_2(2\ P_2 + 1)) <<< \lg w; \qquad b \leftarrow (P_4(2\ P_4 + 1)) <<< \lg w;$$
$$c \leftarrow (P_6(2\ P_6 + 1)) <<< \lg w; \qquad d \leftarrow (P_8(2\ P_8 + 1)) <<< \lg w;$$
$$P_7 \leftarrow ((P_7 - S[4i + 3] >>> c) \oplus d; \quad P_5 \leftarrow (P_5 - S[4i + 2] >>> d) \oplus c;$$
$$P_3 \leftarrow (P_3 - S[4i + 1] >>> a) \oplus b; \quad P_1 \leftarrow (P_1 - S[4i] >>> b) \oplus a;$$

Step 3 $P_8 \leftarrow P_8$-S[3]; $P_6 \leftarrow P_6$-S[2]; $P_4 \leftarrow P_4$-S[1]; $P_2 \leftarrow P_2$-S[0];

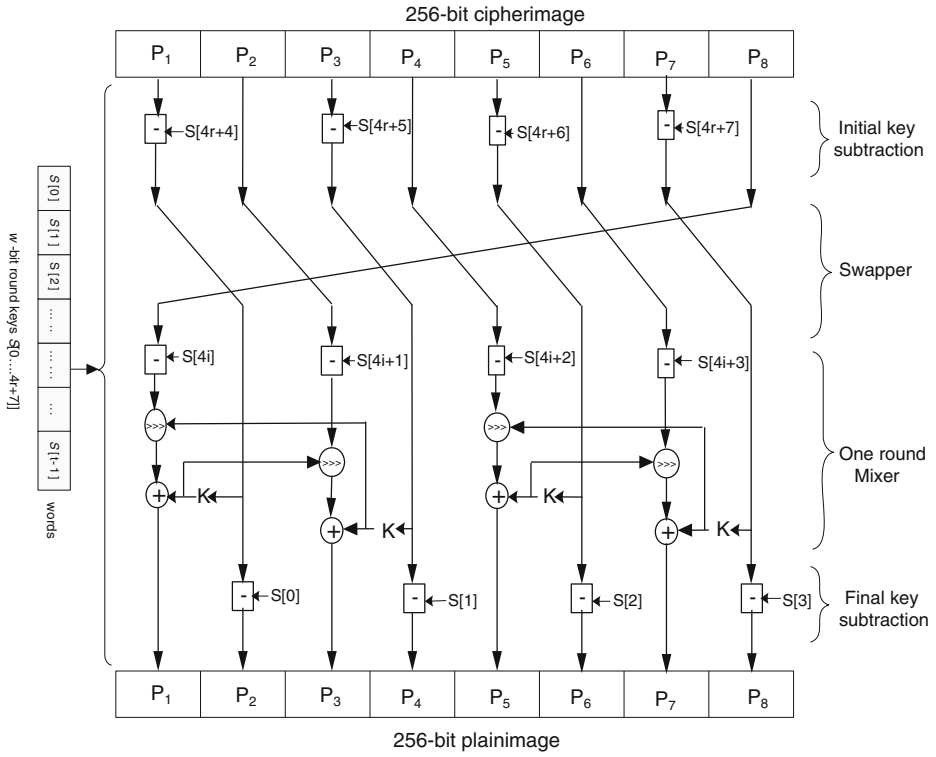


Fig. 4 Diagram of the decryption scheme. Here, $K \leftarrow (P_i(2P_i + 1)) <<< \lg w$

4 Performance and security analysis

4.1 Visual testing

A number of images, from the USC-SIPI image database [32], are encrypted by the proposed method, and a visual test is performed. Figures 5 and 6 show the application of the proposed algorithm for digital images. By comparing the original and the encrypted images in Figs. 5 and 6, there is no visual information observed in the encrypted image, and the encrypted images are non-recognizable in appearance, disorder, and are unsystematic. We cannot obtain any useful information from it, which reveals the confidentiality of the proposed scheme for digital images.

In fact, visual inspection is not enough for judging the quality of encrypted images. Thus, other measuring techniques, objective metrics, are considered to evaluate the degree of encryption quantitatively. The following illustrates different objective metrics that are used to show the performance of the proposed algorithm against competitive algorithms.

4.2 Key space analysis

The key space is the total number of different keys that can be used in the encryption. For a secure image encryption, the key space should be large enough to make brute force attacks infeasible [30]. The proposed algorithm is flexible, with a moderately large key space, which is estimated from the following parameters:

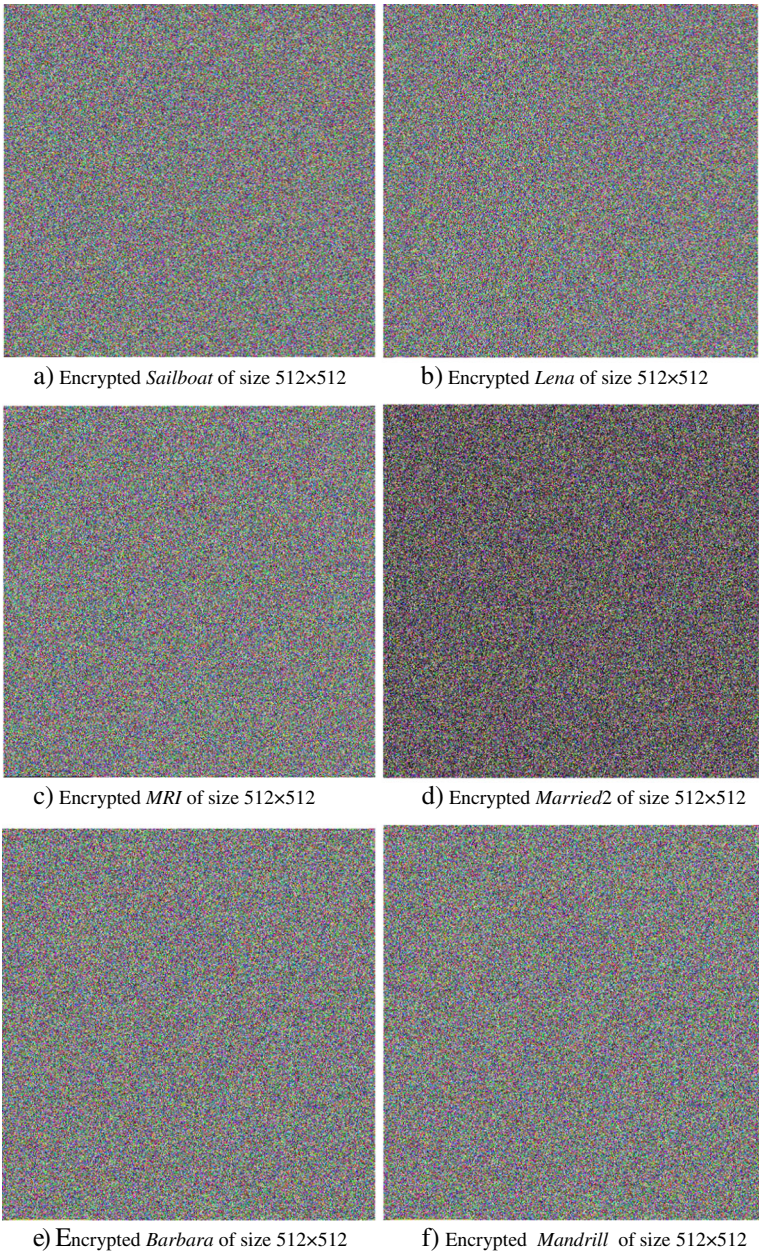
a) *Sailboat* of size 512×512 b) *Lena* of size 512×512

c) *MRI* of size 512×512 d) *Married*2 of size 512×512

e) *Barbara* of size 512×512 f) *Mandrill* of size 512×512

Fig. 5 Original plain images

1. Number of stages in the shift register over GF(p),
2. Initial contents of shift registers,
3. Possible elliptic curves and base point,
4. The secret key of the chaotic map (if the precision is 10^{-14}, the size of the key space for initial condition and control parameter is 2^{93}),
5. The external secret user's key of 256-bit.



a) Encrypted *Sailboat* of size 512×512 b) Encrypted *Lena* of size 512×512

c) Encrypted *MRI* of size 512×512 d) Encrypted *Married2* of size 512×512

e) Encrypted *Barbara* of size 512×512 f) Encrypted *Mandrill* of size 512×512

Fig. 6 Encrypted images using the proposed algorithm

Thus, the key space *KS is* the total number of different keys which can be computed as in Eq. (4).

$$KS = 2^{256} \times 2^{93} \times n \times (p^n - 1) \times (\alpha(p^n - 1)/n) \times 2(2^m - 1) \times \alpha(M) \tag{4}$$

Where n is the number of stages of LFSR over $GF(p)$, (p^n-1) is the number of possible initial values of the LFSR. $(\alpha(p^n-1)\ n)$ is the number of possible feedback coefficients. $2(2^m-1)$ is the number of distinct elliptic curves over $GF(2^m)$. $\alpha(M)$ is the number of base points in the cyclic elliptic curve having largest order M where M is the order of the cyclic elliptic curve and α is the Euler's Totient Function.

For a given cyclic elliptic curve,

$$KS = 2^{256} \times 2^{93} \times n \times (p^n - 1) \times (\alpha(p^n - 1)/n) \times \alpha(M) \tag{5}$$

It is to be noted that unless all the above elements of the key space KS are known to the attacker, decryption using brute force attack is difficult.

Table 4 shows the key space comparison between the proposed encryption algorithm and the recent algorithms. It is worth noting that, the key space and desired key size can be obtained by proper choice of 'n' depending on the level of security required. For example, if we consider the number of shift register stages 'n' is 6, the proposed algorithm would be approximately 2^{477} which implies a large key space.

4.3 Statistical analysis

It is well known that the statistical analysis on cipherimage is of crucial importance for any encryption algorithm. Actually, an ideal cipher should frustrate powerful attacks based on statistical analysis.

Statistical analysis has been performed to show the resistance of the proposed algorithm against statistical attacks. This is shown by a test on the histograms of the cipherimages and on the correlations of the adjacent pixels in the ciphered image.

4.3.1 Histogram analysis

An image histogram illustrates how pixels in an image are distributed by plotting the number of pixels at each color intensity level [5, 9, 10, 16, 18, 20, 22]. A good image encryption scheme should always generate a cipherimage of the uniform histogram for any plainimage. We have calculated and analyzed the histograms of many encrypted images, Figs. 6 (a–f), as well as each original image, Figs. 5 (a–f), that have widely different content.

The histograms of the cipherimages shown in Figs. 7 (a–f) are uniform, significantly different from that of the original images shown in Figs. 8 (a–f), and bear no statistical resemblance to the plainimage. It is clear that the histograms of the encrypted images are fairly uniform and significantly different from the respective histograms of the original images and hence does not provide any clue to employ any statistical attack on the proposed image encryption algorithm.

Table 4 Key space size of the proposed algorithm compared to existing works

Encryption scheme	key space size
Chen et al. [5]	2^{128}
Pareek et al. [20]	2^{80}
Gao et al. [9]	2^{230}
Sathyanarayana et al.[29]	$KS = n \times (p^n - 1) \times (\alpha(p^n - 1)/n) \times \alpha(M) \approx 2^{128}$
Proposed algorithm	$KS = 2^{256} \times 2^{93} \times n \times (p^n - 1) \times (\alpha(p^n - 1)/n) \times \alpha(M) \approx 2^{477}$

4.3.2 Correlation between adjacent pixels in plainimages and cipherimages

For an ordinary image having definite visual content, each pixel is highly correlated with its adjacent pixels. An ideal encryption technique should produce the cipher images with no such correlation in the adjacent pixels (correlation coefficient≈0) [5, 18]. The visual testing of the correlation of adjacent pixels can be done by plotting the distribution of the adjacent pixels in the plain image and its corresponding cipher image. The correlation coefficient between two adjacent pixels in an image is determined as in Eq. (6).

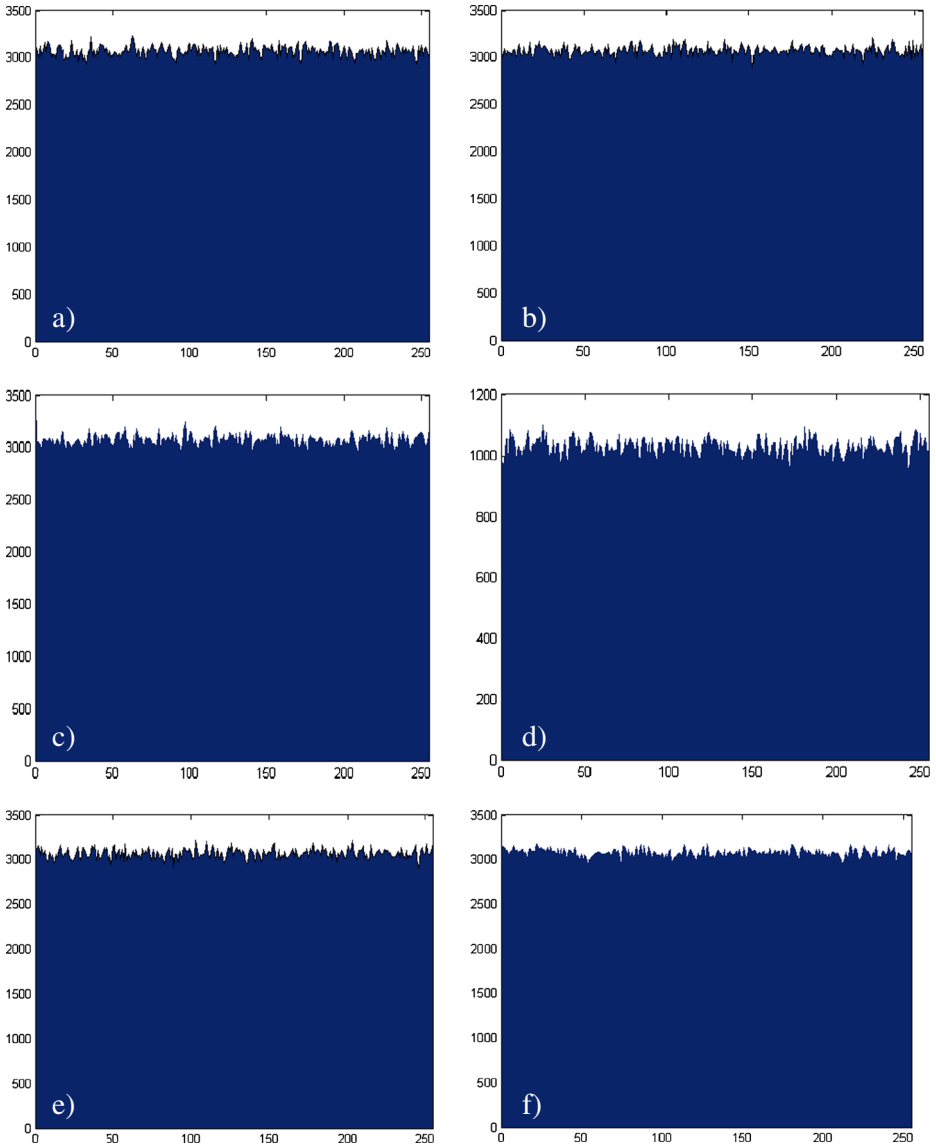


Fig. 7 Histograms of the encrypted images

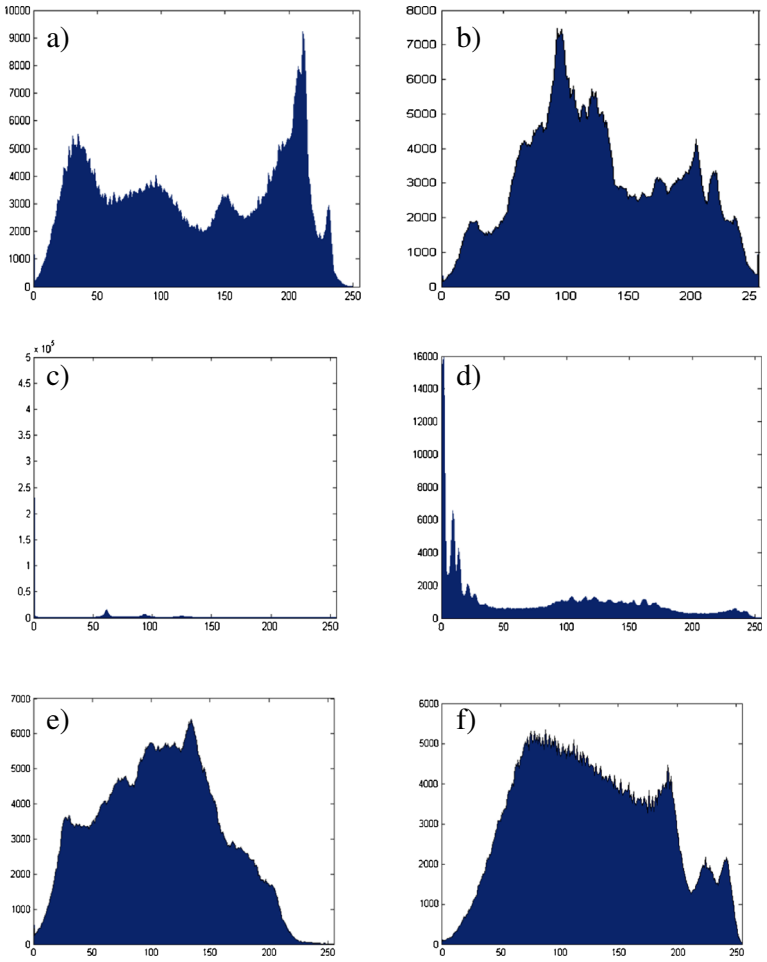


Fig. 8 Histograms of the original images

$$\gamma_{xy} = \frac{\left(\sum_{i=1}^{N}(x_i - mean(x))(y_i - mean(y))\right)}{\left(\left(\sum_{i=1}^{N}(x_i - mean(x))^2\right)^{1/2} * \left(\sum_{i=1}^{N}(y_i - mean(y))^2\right)^{1/2}\right)} \tag{6}$$

where x and y are gray values of two adjacent pixels in the image. The correlation coefficients of the adjacent pixels are calculated and listed in Table 5. The corresponding distribution for the vertical, horizontal and diagonal directions are shown in Figs. 9, 10 and 11. These figures demonstrate that the encryption algorithm has covered up all the plain image characters images and shows good performance with a balanced 0–1 ratio.

4.4 Entropy analysis

Entropy is a statistical measure of randomness in information theory. To measure the entropy H (s) of a source S, we have:

$$H(s) = -\sum_{i=0}^{M} P(s_i)\ \log_2 P(s_i), \tag{7}$$

where M is the total number of symbols $s_i \in S$; $p(s_i)$ is the probability of occurrence of symbol s_i and log denote the base 2 logarithm so that the entropy is expressed in bits. Let us suppose that the source emits 2^8 symbols with equal probability, i.e., $m = \{m_0, m_1, ..., m_{255}\}$ after evaluating Eq. (7), we obtain its entropy $H(m)$=8, corresponding to a truly random source. Actually, given that a practical information source seldom generates random messages, in general its entropy value is smaller than the ideal one. However, when the messages are encrypted, their entropy should ideally be 8. If the output of such a cipher emits symbols with entropy less than 8, there exists a certain degree of predictability, which threatens its security.

The entropy values for 100 plainimages of size 512×512 and corresponding cipherimages are given in Fig. 12. The average entropy value for 100 cipherimages is 7.9997≈8 . This implies that the information leakage in the proposed encryption process is negligible and the encryption scheme is secure against the entropy-based attack.

4.5 Sensitivity analysis

A good image encryption procedure should be sensitive with respect to both the secret key and plain image [3, 17]. The change of a single bit in either the secret key or plainimage should produce a completely different encrypted image.

4.5.1 Key sensitivity

In the proposed algorithm, an incremental change in key; even of the order of (Δ=) 10^{-10}, result in a completely unrecognizable decrypted image. A typical key sensitivity test has been performed using the following steps:

i. Original images of size 512×512 are encrypted by using the test user's secret *key*1 "12345678901234567890123456789012345678901234567890123456789012345678901230" and with the control parameter λ_1=1.97.
ii. The user's secret key is changed slightly to be *key*2 "12345678901234567890123456789012345678901234567890123456789012345678901231" and is used to encrypt the same image.
iii. The control parameter is changed slightly to be λ_1=1.97 0000000001 and used to encrypt the same image.
iv. The two cipherimages in ((i) and (ii)) are compared pixel-by-pixel.

The test results for key sensitivity are shown in Figs. 13, 14, 15 and 16. We can see that if a tiny change in the key; even of the order of (Δ=) 10^{-10}, results in a completely unrecognizable image.

Table 5 Correlation coefficient of two adjacent pixels in plainimages and cipherimages on *lena* image of size 512×512

Test images	Horizontal	Vertical	Diagonal
Original image	0.995810248	0.990848889	0.987584908
Encrypted image	0.000224045	0.006131844	0.000420505

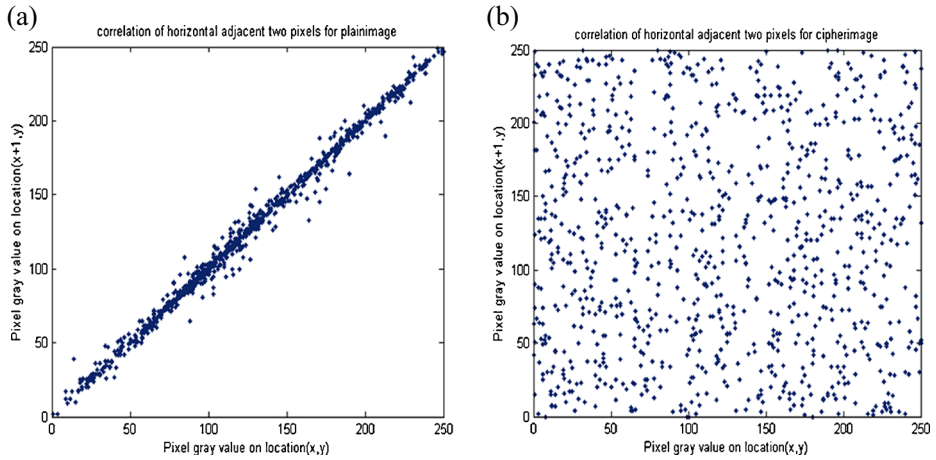


Fig. 9 Two horizontally adjacent pixels correlation in original image/encrypted image, respectively

4.5.2 Plainimage sensitivity

To test the sensitivity to the plainimage, we modify the pixel value (*PV*) at grid (50, 50) of the original image (d) by adding one, i.e.,

$$PV\left(s'_{50,50}\right) = \left(\left(PV\left(s_{50,50}\right)\right) + 1\right) \mod 256 \tag{8}$$

The results of the plainimage sensitivity are given in Figs. 13, 14, 15 and 16 (f), which show the encrypted image with *key*1 and control parameter λ_1=1.97 when only one pixel changed in Figs. 13, 14, 15 and 16 (a).

4.6 Differential attack

In general, the differential attack means finding out a meaningful relationship between the plainimage and the cipherimage (chosen-plaintext attack) by making a minor change (e.g.,

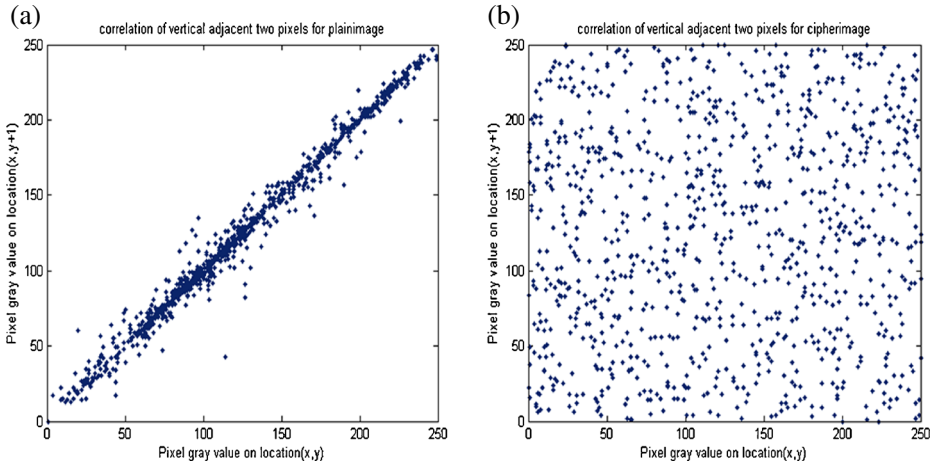


Fig. 10 Two vertically adjacent pixels correlation in original image/encrypted image, respectively

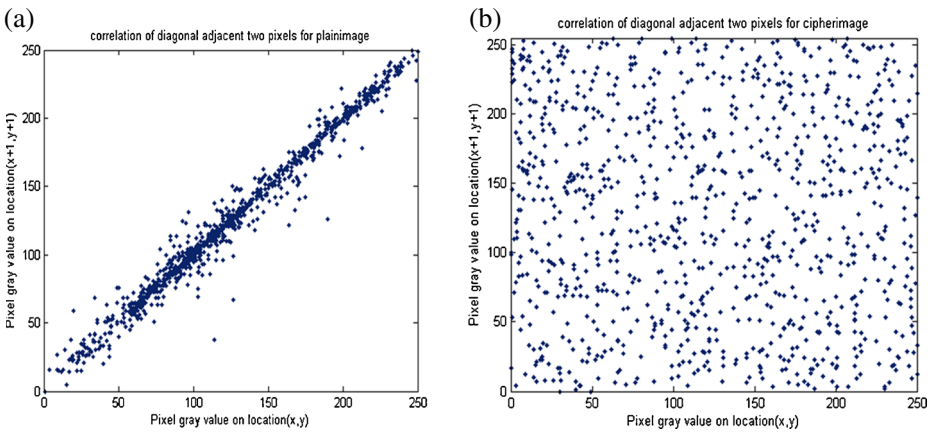


Fig. 11 Two diagonally adjacent pixels correlation in original image/encrypted image, respectively

modify only one pixel) in the encrypted image, and then observe the change of the result [2, 4, 23, 31, 34]. To resist the differential attack, a minor change in the plainimage should cause a significant change in the cipherimage. To quantitatively test the influence of a one-pixel change on a cipherimage, two common measures are used, i.e., number of pixels change rate (NPCR) and unified average changing intensity (UACI), they can be defined as:

$$NPCR = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} f(i,j)}{m \times n} \times 100\%, \tag{9}$$

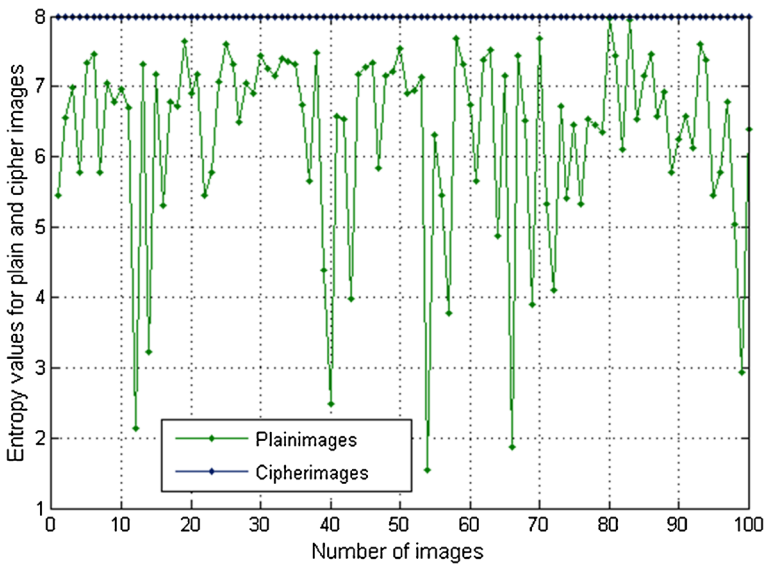


Fig. 12 Entropy value for 100 images of size 512×512 and corresponding cipherimages

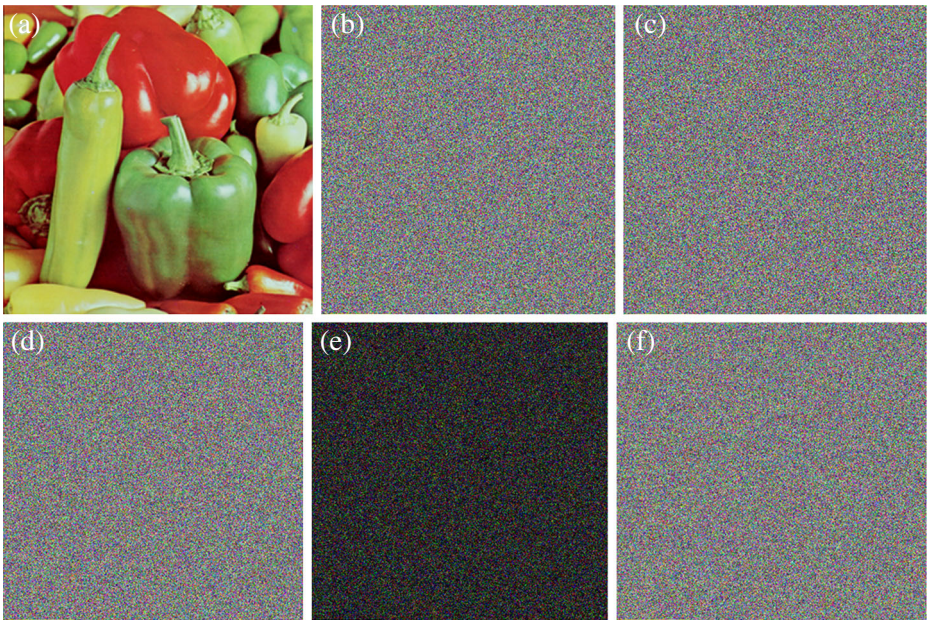


Fig. 13 Results of sensitivity experiment with respect to the key and the plainimage. (**a**) is the original 'Pepper' image, (**b**) is the encrypted image of (**a**) with *key1* and control parameter λ_1=1.97, (**c**) is the encrypted image of (**a**) with *key2*, (**d**) is the encrypted image of (**a**) with slight change in the control parameter λ=1.970000000001, (**e**) difference image between the two cipherimages (Figs. 13(c) and (d)), (**f**) is the encrypted image with *key1* and control parameter λ_1=1.97 when only one pixel changed in (**a**)

$$UACI = \frac{\left[\sum_{i=1}^{m}\sum_{j=1}^{n}|f'(i,j) - f''(i,j)|\right]/225}{m \times n} \times 100\% \tag{10}$$

where f' and f'' are two images with the same size $m \times n$. m and n are width and height of the image. Define a bipolar array, f, with the same size as images f' and f''. Then, $f(i,j)$ is determined by $f'(i, j)$ and $f''(i, j)$, namely, if $f'(i, j)=f''(i, j)$ then $f(i,j)=1$; otherwise, $f(i,j)=0$.

Here, we test NPCR and UACI for different images [32] as shown in Figs. 17 and 18. The percentage of pixels changed in the encrypted image is grater than 99 % for NPCR and is grater than 33 % for UACI even with one-bit difference in the plainimages. This result shows that the proposed algorithm has a strong ability to resist a differential attack.

4.7 Computational complexity and speed performance

The running time of an encryption algorithm is determined by many factors such as programming language, programming skill, execution environment, etc. In the following, we discuss the efficiency of the proposed algorithm from both speed performance and the computational complexity.

4.7.1 Speed performance

The running time of an encryption algorithm is determined by many factors such as programming language, programming skill, execution environment, etc. In what follows,

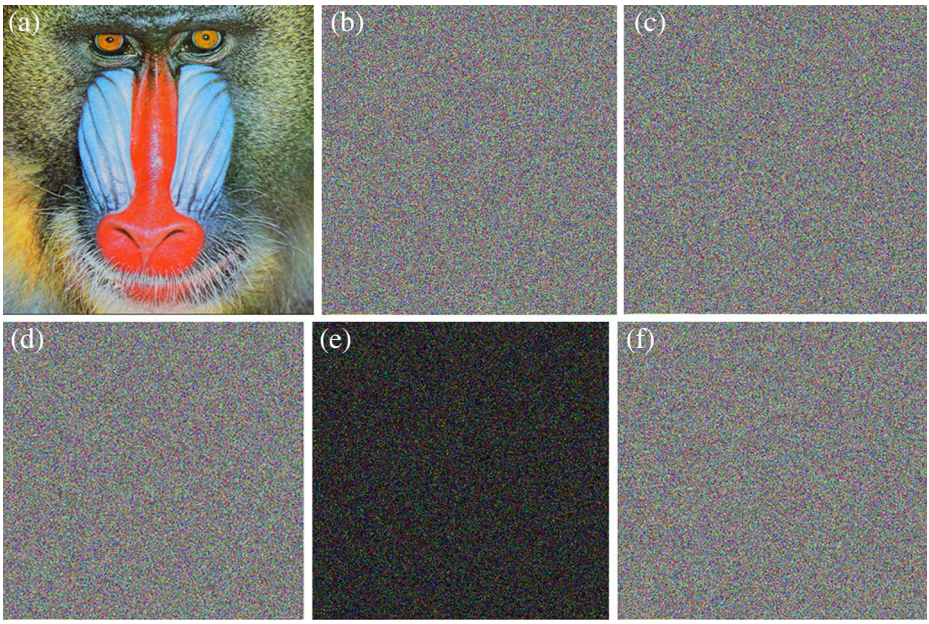


Fig. 14 Results of sensitivity experiment with respect to the key and the plain image. (**a**) is the original 'Mandrill' image, (**b**) is the encrypted image of (**a**) with *key1* and control parameter λ_1=1.97, (**c**) is the encrypted image of (**a**) with *key*2, (**d**) is the encrypted image of (**a**) with slight change in the control parameter λ=1.970000000001, (**e**) difference image between the two cipherimages (Figs. 14(c) and (d)), (**f**) is the encrypted image with *key*1 and control parameter λ_1=1.97 when only one pixel changed in (**a**)

we discuss the efficiency of the proposed algorithm from both encryption time and the number of rounds.

Running time Here, we have implemented the proposed algorithm and other algorithms kept at the same number of rounds r=2, word size in bits w=32 and key length in bytes b=16. We have used the Visual C++ compiler on a computer of Dual-Core CPU 2.7 GHz and 1.99 GB of RAM. The operating system used is Windows XP SP2. Thus, as far as the running time is concerned, our algorithm is acceptable, as shown in Table 6.

Encryption rounds Since the number of encryption rounds affects the computational complexity and the speed of the encryption algorithm, performance evaluations should be performed after each encryption round. The iterations stop only when all the performance requirements are satisfied. For security purpose, NPCR and UACI should be greater than 99 % and 33 %, respectively. From Table 7, we can see that only two rounds of encryption is enough to receive a perfect cipherimage with high performance for the proposed algorithm.

4.7.2 Encryption operations and inner loops

The proposed algorithm used familiar primitive operations that are efficiently implemented on modern processors such as cyclic rotations, Bit-wise exclusive-OR, addition, subtraction, and integer multiplication. As can be seen in Section 3, the inner loops of the encryption and key expansion are simple and compact in terms of complexity. In addition, the complexity of the derived sequence of cyclic elliptic curve points in the proposed algorithm depends on the

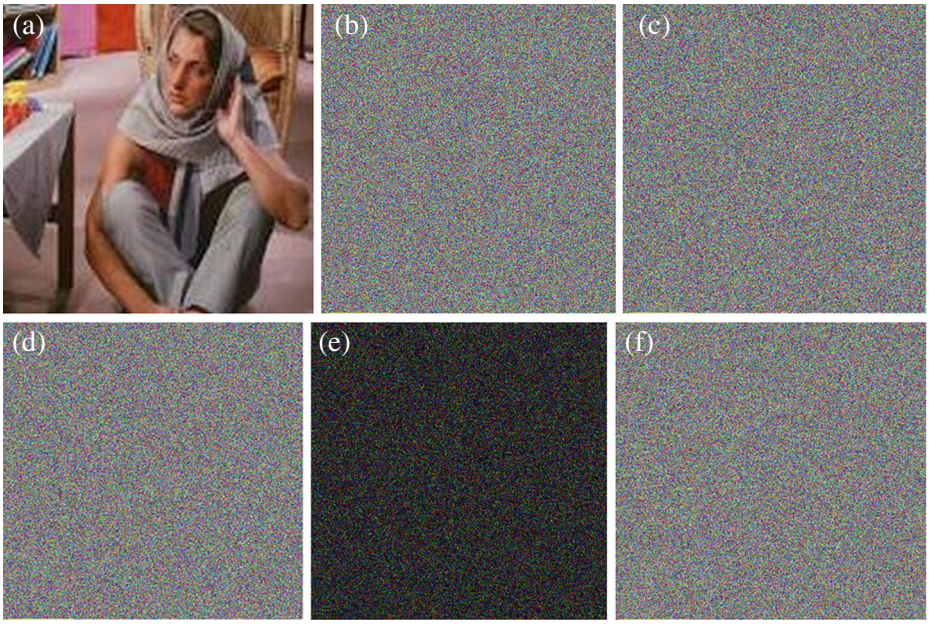


Fig. 15 Results of sensitivity experiment with respect to the key and the plain image. (**a**) is the original 'Barbara' image, (**b**) is the encrypted image of (**a**) with *key1* and control parameter λ_1=1.97, (**c**) is the encrypted image of (**a**) with *key*2, (**d**) is the encrypted image of (**a**) with slight change in the control parameter λ=1.970000000001, (**e**) difference image between the two cipherimages (Figs. 15 (c) and (d)), (**f**) is the encrypted image with *key*1 and control parameter λ_1=1.97 when only one pixel changed in (**a**)

number of stages of the shift register 'n', which is linear and equal to n. However, the complexity can be increased by increasing the number of stages of the shift register 'n'. In this paper, the number of shift register stages 'n' is 6.

5 Discussion and overall performance

In this section, we discuss the most important findings of the paper.

5.1 Confusion

The histogram and correlation analysis of adjacent pixels both indicate that the proposed algorithm possesses a good property of confusion. This mainly results from the pseudo-randomness of the cyclic elliptic curve and chaotic system as well as primitive operations in the key schedule. They work together to introduce the random-like effect to the cipher image.

5.2 Diffusion

The inner loop of the encryption and decryption is based on data-dependent rotations as well as integer multiplication, which is very effective primitive "diffusion". The diffusion effect is increased due to the heavy dependence on rotations, the quadratic function that speed up the avalanche of change between rounds, and fixed bit shifting by five bits, which complicates

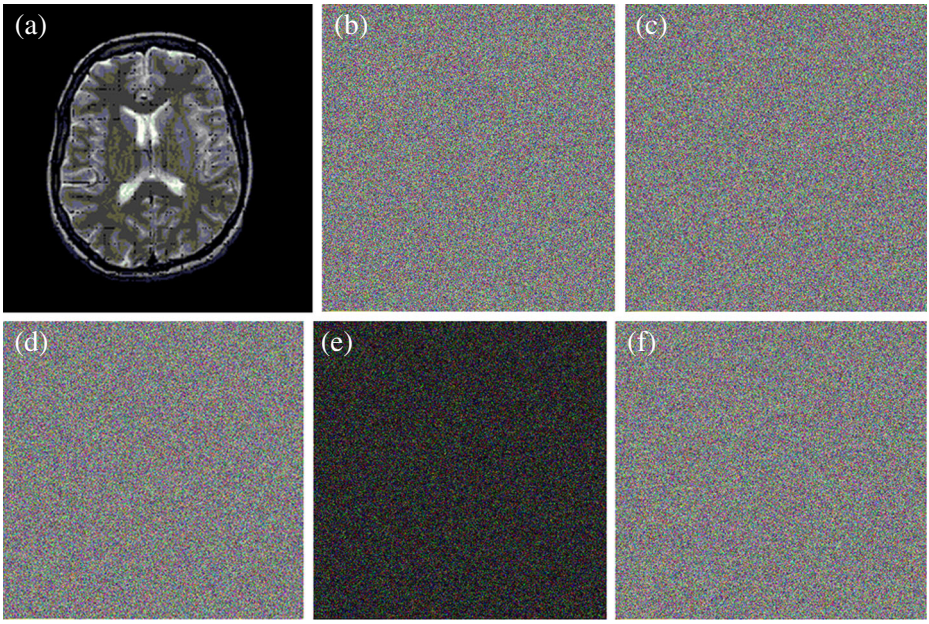


Fig. 16 Results of sensitivity experiment with respect to the key and the plain image. (**a**) is the original 'MRI' image, (**b**) is the encrypted image of (**a**) with *key1* and control parameter $\lambda_1=1.97$, (**c**) is the encrypted image of (**a**) with *key*2, (**d**) is the encrypted image of (**a**) with slight change in the control parameter $\lambda=1.970000000001$, (**e**) difference image between the two cipherimages (Figs. 16 (c) and (d)), (**f**) is the encrypted image with *key*1 and control parameter $\lambda_1=1.97$ when only one pixel changed in (**a**)

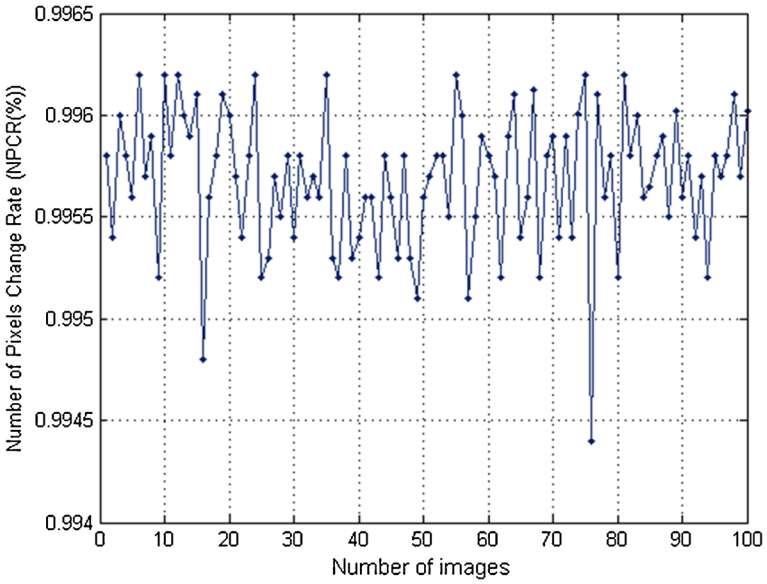


Fig. 17 NPCR for 100 different images of size 256×256

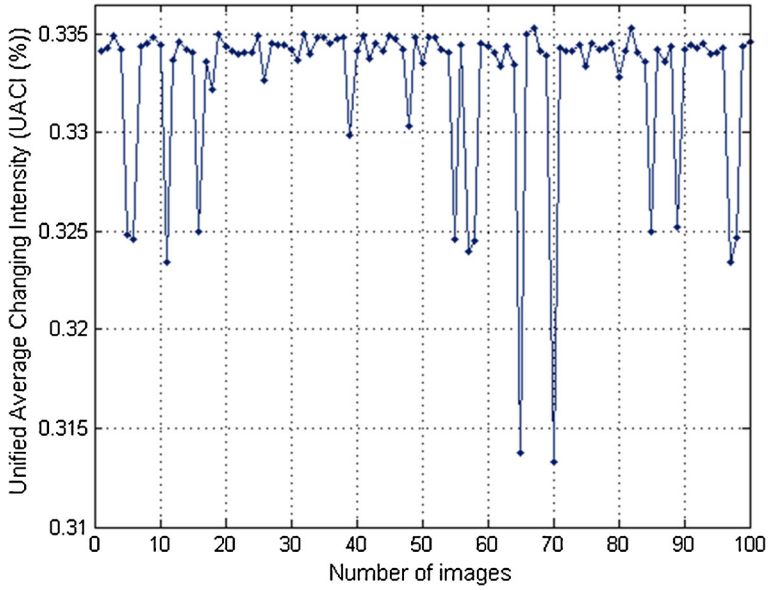


Fig. 18 UACI for 100 different images of size 256×256

advanced cryptanalytic attacks. This allows the proposed algorithm to run with fewer rounds of encryption and decryption at increased security.

5.3 Resistance to brute force attack

Regarding brute-force attack, the security of the proposed algorithm depends on the size of the key space. As mentioned above, the total length of our key is 477 bits (suppose the number of shift register stages n is 6), which is very safe for ordinary business applications.

5.4 Resistance to differential attacks

From aforementioned analysis and numerical experiments, the proposed algorithm has fairly uniform histograms, and low correlation between pixels in cipherimages. In addition, the conducted simulations demonstrated that the entropy, key sensitivity, number of pixel change rate (NPCR), and unified average changing intensity (UACI) can satisfy the performance requirements for the confidentiality of digital images. Thus, the proposed algorithm can perform outstandingly well against differential attacks such as chosen-plaintext and known-plaintext attacks.

Table 6 Test of the encryption speed of Lena testpat image of size 512×512 in pixels and 768 KB

Encryption scheme	Speed	System characteristics	Platform
RC5	0.046 s	Pentium (R) 2.7 GHz	Visual C++
RC6	0.039 s	Pentium (R) 2.7 GHz	Visual C++
Ref. [2]	0.046 s	Pentium (R) 2.7 GHz	Visual C++
Ref. [3]	0.033 s	Pentium (R) 2.7 GHz	Visual C++
Proposed scheme	0.035 s	Pentium (R) 2.7 GHz	Visual C++

Table 7 Number of rounds to achieve NPCR >0.996 and UACI >0.334

Algorithm	Number of rounds
Proposed algorithm	2
RC6	20
Ref. [34]	6
Ref. [35]	3

5.5 Overall security and performance

The proposed algorithm possesses the following advantages:

- Incorporation of the cyclic elliptic curve with chaotic system as well as the cryptographic primitive operations, which strengthen the round keys for encryption and enlarge the key space required to resist the brute force attack.
- The proposed algorithm acts on 256-bit input/output blocks within eight 32-bit registers.
- Several simple steps in the round increase the rate of diffusion: integer multiplication, the quadratic equation, and fixed bit shifting by five bits, which is a secure way against both linear and differential attacks.
- Extensive use of data-dependent rotations that greatly increases the diffusion achieved per round.
- The proposed algorithm runs with fewer rounds (2 rounds) at increased security and does not use look-up tables during the encryption/decryption unlike other algorithms.
- The operations used during the encryption process are efficiently implemented on modern processors.
- The proposed algorithm has high key and plainimage sensitivity together with a large key space ($> 2^{477}$, depend on the number of shift register stages), which is very safe for ordinary business applications.
- The proposed algorithm is faster than competitive algorithms.
- Simple structure, which permits a compiler to produce well-optimized code, results in better performance without hand-optimizations.
- Like RC5 and RC6 algorithms, the proposed algorithm provides a great amount of flexibility with regards to the number of rounds r, the size of the encryption key b and the word size of the basic computational unit w.

6 Concluding remarks

A new encryption algorithm for still visual data based on a cyclic elliptic curve and chaotic system is introduced, which operates on 256-bit plainimage/cipherimage blocks. In the proposed algorithm, random sequences of cyclic elliptic curve points are mixed with the chaotic system as well as primitive operations to generate round keys for encryption. In the encryption process, effective confusion and diffusion are introduced based on a quadratic transformation function and cryptographic primitive operations, which made the encryption more secure with less computation overhead. Experiments conducted show that the entropy, key and plainimage sensitivity, number of pixel change rate (NPCR), and unified average changing intensity (UACI) can satisfy the performance requirements for the confidentiality of digital images. In addition, the proposed algorithm outperforms the competing algorithms in terms of speed performance. In the future, we plan to consider the frequency domain-based encryption.

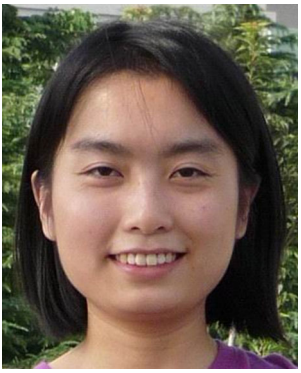
Acknowledgments The authors wish to thank all the anonymous reviewers for their suggestions to improve this paper. This work is supported by the National Natural Science Foundation of China (60832010, 61100187), the Fundamental Research Funds for the Central Universities (Grant No. HIT. NSRIF. 2010046), the China Postdoctoral Science Foundation (2011M500666) and Higher Education Commission of Egypt.

References

1. Akhavan A, Samsudin A, Akhshani A (2011) A symmetric image encryption scheme based on combination of nonlinear chaotic maps. J Frankl Inst. doi:10.1016/j.jfranklin.2011.05.001
2. Amin M, Abd El-Latif AA (2010) Efficient modified RC5 based on chaos adapted to image encryption. J Electron Imag 19(1)
3. Amin M, Faragallah OS, Abd El-Latif AA (2010) A chaotic block cipher algorithm for image cryptosystems. Comm Nonlinear Sci Numer Simulat 15:3484–3497
4. Behnia S, Akhshani A, Mahmodi H, Akhavan A (2009) Applications of tripled chaotic maps in cryptography. Chaos, Solitons Fractals 40:505–519
5. Chen G, Mao Y, Chui CK (2004) A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos, Solitons Fractals 21:749–761
6. Deepthi PP, Sathidevi PS (2009) New stream ciphers based on elliptic curve point multiplication. Comput Commun 32:25–33
7. ElGamal T (1985) A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans Inf Theory IT-31:469–472
8. Fridrich J (1998) Symmetric ciphers based on two-dimensional chaotic maps. Int J Bifurc Chaos 8(6):1259–1284
9. Gao T, Chen Z (2008) A new image encryption algorithm based on hyper-chaos. Phys Lett A 372:394–400
10. Guan Z-H, Huang F, Guan W (2005) Chaos based image encryption algorithm. Phys Lett A 346:153–157
11. Hansen T, Mullen GL (1992) Primitive polynomials over finite fields. Math Comput 59(200):639–643
12. Jafarizadeh MA, Behnia S (2002) Hierarchy of chaotic maps with an invariant measure and their compositions. J Nonlinear Math Phys 9(1):1–16
13. Li C, Li S, Asim M, Nunez J, Alvarez G, Chen G (2009) On the security defects of an image encryption scheme. Image Vis Comput 27:1371–1381
14. Li C, Li S, Lo K-T (2011) Breaking a modified substitution–diffusion image cipher based on chaotic standard and logistic maps. Comm Nonlinear Sci Numer Simulat 16:837–843
15. Lian S (2008) Multimedia content encryption: techniques and applications. Auerbach Publications, CRC Press, New York
16. Lian S (2009) Efficient image or video encryption based on spatiotemporal chaos system. Chaos, Solitons Fractals 40:2509–2519
17. Mao Y, Chen G (2005) Chaos-based image encryption. In: Bayro E (ed) Handbook of computational geometry for pattern recognition, computer vision, neuralcomputing and robotics. Springer-Verlag, Berlin, Germany, pp 231–265
18. Mao Y, Chen G, Lian S (2004) A novel fast image encryption scheme based on 3D chaotic baker maps. Int J Bifurc Chaos 14(10):3613–3624
19. Morain F (1990) Building cyclic elliptic curves modulo large primes. LNCS 547:328–336
20. Pareek NK, Patidar V, Sud KK (2006) Image encryption using chaotic logistic Map. Image Vis Comput 24:926–934
21. Patidar V, Pareek NK, Purohit G, Sud KK (2010) Modified substitution–diffusion image cipher using chaotic standard and logistic maps. Comm Nonlinear Sci Numer Simulat 15:2755–2765
22. Patidar V, Pareek NK, Sud KK (2009) A new substitution–diffusion based image cipher using chaotic standard and logistic maps. Comm Nonlinear Sci Numer Simulat 14:3056–3075
23. Pisarchik AN, Zanin M (2008) Image encryption with chaotically coupled chaotic maps. Physica D 237:2638–2648
24. Rhouma R, Belghith S (2008) Cryptanalysis of a new image encryption algorithm based on hyper-chaos. Phys Lett A 372:5973–5978
25. Rhouma R, Belghith S (2008) Cryptanalysis of a spatiotemporal chaotic image/video cryptosystem. Phys Lett A 372:5790–5794
26. Rhouma R, Solak E, Belghith S (2010) Cryptanalysis of a new substitution–diffusion based image cipher. Comm Nonlinear Sci Numer Simulat 15:1887–1892
27. Ronald L, Rivest MJB, Robshaw R, Sidney and Yin YL (1998) "The RC6^{TM} block cipher", version1.1. URL: ftp://ftp.rsasecurity.com/pub/rsalabs/rc6/rc6v11.pdf
28. Sathyanarayana SV, Aswatha Kumar M, Hari Bhat KN (2010) Random binary and non-binary sequences derived from random sequence of points on cyclic elliptic curve over finite field $GF(2^m)$ and their properties. Inf Secur J: A Glob Perspect 19(2):84–94

29. Sathyanarayana SV, Aswatha Kumar M, Hari Bhat KN (2011) Symmetric key image encryption scheme with key sequences derived from random sequence of cyclic elliptic curve points. Int J Netw Secur 12:137–150
30. Schneier B (1996) Applied cryptography—protocols, algorithms, and source code in C, 2nd edn. Wiley, Hoboken
31. Sun F, Liu S, Li Z, Lü Z (2008) A novel image encryption algorithm based on spatial chaos map. Chaos, Solitons Fractals 38:631–640
32. The USC-SIPI image database, http://sipi.usc.edu/database/database.php
33. Wang K, Pei, Zou L, Song A, He Z (2005) On the security of 3D Cat map based symmetric image encryption scheme. Phys Lett A 343(6):432–439
34. Wong K-W, Kwok BS-H, Law W-S (2008) A fast image encryption scheme based on chaotic standard map. Phys Lett A 372:2645–2652
35. Zhou Q, Wong K-W, Liao X, Xiang T, Hu Y (2008) Parallel image encryption algorithm based on discretized chaotic map. Chaos, Solitons Fractals 38:1081–1092

Ahmed A. Abd El-Latif was born in Egypt, in July 1984, received the B.Sc. degree with honor rank in Mathematics and Computer Science from Menoufia University, Egypt in 2005, and M.Sc. degree in Computer Science in 2010. He was a teaching assistant in Mathematics department from March 2007 to May 2010. He is currently pursuing the Ph.D. degree at Harbin Institute of Technology (H.I.T), Harbin, P. R. China. His areas of interests are cryptography, application of chaotic systems in multimedia content encryption, secret image sharing and biometrics. E-mail: ahmed_rahiem@yahoo.com.



Li Li received the B.S. degree in Computer Science and Technology from Harbin Institute of Technology at Weihai, Weihai, China, in 2005. She received the M.E. degree in Computer Science and Technology from Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, China, in 2007. She is currently pursuing the Ph.D. degree at Harbin Institute of Technology (H.I.T) Shenzhen Graduate School, P. R. China. Her areas of interests are media security, image processing, and network coding.

Xiamu Niu was born in China, in May 1961, received the B.S. degree and M.S. degree in Communication and Electronic Engineering from Harbin Institute of Technology (HIT), Harbin, P. R. China in 1982 and 1989 respectively, and received the Ph.D degree in Instrument Science and Technology in 2000. He was an invited scientist and staff member in Department of Security Technology for Graphics and Communication System, Fruanhofer Institute for Computer Graphics, Germany, from 2000 to 2002. He was awarded the Excellent Ph.D Dissertation of China in 2002. He now is the Professor (doctoral advisor) and Superintendent of Information Countermeasure Technique Institute HIT, Director of Information Security Technique Research Center, HIT-ShenZhen. He is SPIE member, ACM member, IEEE member, and the advanced CIE member. He has published 3 works and more than 130 papers cited by SCI and EI. His current research fields include computer information security, hiding communication, cryptography, digital watermarking, signal processing and image processing etc.