Multimed Tools Appl (2018) 77:8911–8938
DOI 10.1007/s11042-017-4786-7

A novel image encryption scheme based on logistic map and dynatomic modular curve

Bo Li[1,2] · Xiaofeng Liao[1] · Yan Jiang[3]

Received: 26 May 2016 / Revised: 2 March 2017 / Accepted: 28 April 2017 /
Published online: 31 May 2017

Abstract A new image encryption and decryption algorithm based on chaotic map and dynatomic modular curve is proposed in this paper. Firstly, the definition of dynatomic modular curve and its periodic points are introduced, and a property of the dynatomic modular curve is proved. Secondly, the relationship between the Logistic map and the dynatomic modular curve is discussed. Finally, the encryption algorithm which is composed of permutation of pixels and substitution is given. In order to eliminate sufficiently the relation between adjacent pixels in the image, pixel values of the original image are sorted as index function, which derives from Logistic map and dynatomic modular curve. And XOR operation is performed between the scrambled pixel sequence and projective transformation sequence. Simulation experiments and nonparametric hypothesis test demonstrate that the proposed algorithm is secure to resist different types of attacks and it can be applied to real-time encryption.

Keywords Logistic map · Dynatomic modular curve · Projective transformation · Chi-Square test · K-S test

1 Introduction

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resource(e.g., network, servers, storage, applications)that can be rapidly provisioned and released with minimal management effort or service provider interaction. Attracted by these appealing features, both individuals and enterprises are actively outsourcing their data to the cloud. But, outsourcing sensitive

✉ Xiaofeng Liao
xfliao@cqu.edu.cn

1 College of Electronic and Information Engineering, Southwest University, Chongqing 400715, China

2 Department of Mathematics, Logistical Engineering University, Chongqing 401331, China

3 Department of Mathematics, Chong Qing Energy College, Chongqing 402262, China

information (such as e-mails, personal health records, company finance data, government documents, and secret images etc.) to remote servers will bring privacy concerns. The general approach of preserving privacy data is to encrypt it before outsourcing [10]. From a security point of view, this process mainly contains two aspects. On the one hand, the inverted sort index of the file needs to be encrypted [5, 26]. On the other hand, the file to be uploaded needs to be encrypted [18, 19]. Some encryption algorithms have been proposed to solve the latter problem, such as AES and DES algorithm. But, comparing these conventional encryption algorithms, chaos-based ones have suggested more secure and fast encryption methods [15, 24].

The first chaos-based encryption algorithm was proposed in 1989 [12]. Since then more and more researchers have investigated and analyzed many kinds of chaos-based encryption algorithms. The improvement of encryption algorithm mainly includes security and computational cost. In terms of security, some researchers were working to eliminate sufficiently the relativity of adjacent pixel in images [8, 13, 17, 27, 29, 30]. In addition, other researchers were studying how to increase the size of the key space to ensure the security of the encryption algorithm [7, 20, 28]. Moreover, some people even put forward the method of keeping secret communication from the angle of pulse synchronization [2, 22, 23, 25]. In terms of computational cost, a real-time and fast encryption method was presented based on orthonormal matrices [3]. The algorithm not only considers the statistical attack, but also pays more attention to the speed of encryption. In [21], a two stage combinational approach for image encryption was proposed. This algorithm need only one password for both stages and it had a low computational complexity. The above mentioned schemes have made a great contribution to the security and computational cost. However, there are few algorithms which can simultaneously take into account the two aspects. In [28], an image encryption scheme was designed based on 2D hyper-chaotic system. They claimed that the algorithm had high security. However, the computational cost of using two-dimensional hyper chaotic system was relatively high. If 1-D chaotic system be used in this algorithm, the speed of operation would be improved. But, the security could not be guaranteed. Moreover, the encryption scheme based on chaotic hiding and modulation of 1-D chaotic systems was decrypted by multi-step nonlinear prediction method. So, it is a great challenge to design a low dimensional chaotic encryption scheme with security and low computational cost. Fortunately, some people started to study this problem. In [14], the author proposed a new image encryption algorithm based on parameter-varied Logistic map. This method could resist the attack of phase space reconstruction, and it has a low computational complexity. However, the difference between some data which derived from Logistic map would be very small for different parameters. It may cause a larger calculation error and reduce the security of the algorithm. In order to make up for these deficiencies and expand the scope of the data, this paper proposes a new image encryption scheme based on Logistic map and dynatomic modular curve (DMC). In stage of diffusion, the original pixel values are sorted as index function which derives from the logistical map and the DMC. In stage of confusion, the XOR operation between the scrambled pixel sequence and projective transformation sequence is performed. The simulation experiments show that the proposed algorithm has a low computational complexity, and sensitivity to the key, and it has a good property of resistance statistical attack, differential attack, and malicious attack.

The advantages of the proposed scheme are summarized as follows:

(1) The DMC is the first time to apply in image encryption. Not only it can help us improve the calculation precision by increasing infinity point and infinity line, but also can expand the scope of chaotic data.
(2) The relationship between the Logistic map and the DMC is discussed, and it is used to encrypt and decrypt image.
(3) Chi-square and Kolmogorov-Smirnov test are applied to verify the distribution of the encrypted image pixels and correlation coefficients in performance analysis. They can help us test the security of the encryption algorithm from the view of statistical analysis.
(4) The key space is extended by introducing the adjustment parameter and amplification parameter.

The rest of this paper is organized as follows. In section 2, in order to find the relationship between the Logistic map and the DMC, the definition of the DMC and the type of period are introduced, and a property of the DMC is proved. In section 3, the relationship between the Logistic map and the DMC is discussed, and its role in image encryption is summarized. In section 4, image encryption and decryption algorithm based on the Logistic map and the DMC are given. In section 5, some images which selected from the USC-SIPI image database are used to test effectiveness of the proposed algorithm. And the Chi-square test is used to verify the histogram distribution of the encrypted image pixels. Performance analysis of the proposed algorithm is described in section 6. It includes statistical analysis, adjacent pixels correlation analysis and simulation, *NPCR* and *UACI* calculation and simulation, key sensitivity test, information entropy calculation and algorithm intensity analysis. In this section, the distribution of correlation coefficient of encrypted image is verified by K-S test, and the experimental results are compared with other algorithms from correlation coefficient, key sensitivity and computational complexity. Section 7 summaries the main innovative points of the proposed algorithm and lists some research problems in the future.

2 Dynatomic modular curve

Firstly, we introduce the definition of dynatomic polynomial and its periodic point in projective space. And then, the conception of the DMC is put forward. Finally, a property theorem of the DMC is proved.

Definition 1 [22] $\Phi_n^*(z)$ is called the n -th *dynatomic polynomial* and is given by the following formula,

$$\Phi_n^*(z) = \prod_{k|n} \left(\phi^k(z) - z\right)^{\mu\left(\frac{n}{k}\right)}.$$

Here μ is the Mobius function. μ is defined by $\mu(1) = 1$ and

$$\mu\left(p_1^{e_1} p_2^{e_2} \cdots p_r^{e_r}\right) = \begin{cases} (-1)^r, & \text{if } e_1 = e_2 = \cdots = e^r = 1, \\ 0, & \text{if any } e_i \geq 2. \end{cases}$$

$\phi^k(z)$is the k -th iteration of ϕ. If we let $\phi(z) \in K[z]$ be a polynomial, where $K[z]$ is a polynomial set on perfect field K. And consider that

$$\phi(z) = \phi_c(z) = -z^2 + c$$

Some expressions of $\Phi_n^*(z)$ can be obtained,

$$\Phi_1^*(z) = \Phi_1^*(c,z) = \prod_{k|1} \left(\phi^k(z) - z\right)^{\mu\left(\frac{1}{k}\right)} = \phi(z) - z = -z^2 - z + c, \tag{1}$$

$$\begin{aligned} \Phi_2^*(z) = \Phi_2^*(c,z) &= \prod_{k|2} \left(\phi^k(z) - z\right)^{\mu\left(\frac{2}{k}\right)} = (\phi(z) - z)^{\mu(2)} \left(\phi^2(z) - z\right)^{\mu(1)} \\ &= (\phi(z) - z)^{-1} \left(\phi^2(z) - z\right) = z^2 - z + (1 - c) \ , \end{aligned} \tag{2}$$

$$\begin{aligned} \Phi_3^*(z) = \Phi_3^*(c,z) &= \prod_{k|3} \left(\phi^k(z) - z\right)^{\mu\left(\frac{3}{k}\right)} = (\phi(z) - z)^{\mu(3)} \left(\phi^3(z) - z\right)^{\mu(1)} \\ &= z^6 - z^5 + (1 - 3c)z^4 + (2c - 1)z^3 + \left(3c^2 - 3c + 1\right)z^2 \\ &+ \left(-c^2 + 2c - 1\right)z - \left(c^3 - 2c^2 + c - 1\right). \end{aligned} \tag{3}$$

Similarly, other expressions of $\Phi_n^*(z)$ $(n = 4, 5, \cdots)$ can be obtained. Let $\Phi_n(P) = \phi^n(P) - P$, we can make the following definition.

Definition 2 Let $\phi(z) \in K[z]$ be a rational map and let $P \in \mathbb{P}^1$ be a periodic point for ϕ, where $\mathbf{P}^1$ represents 1-dimensional projective space.

(1) P has *period n* if $\Phi_n(P) = 0$.
(2) P has *formal period n* if $\Phi_n^*(P) = 0$.
(3) P has *primitive* (*or exact*) *period n* if $\Phi_n(P) = 0$ and $\Phi_m(P) \neq 0$ for all $m < n$.

We set the notation

$$\begin{aligned} &\mathrm{Per}_n(\phi) = \left\{P \in \mathbb{P}^1 : \ \Phi_n(P) = 0\right\}, \\ &\mathrm{Per}_n{}^*(\phi) = \left\{P \in \mathbb{P}^1 : \Phi_n{}^*(P) = 0\right\}, \\ &\mathrm{Per}_n^{**}(\phi) = \left\{P \in \mathbb{P}^1 : \Phi_n(P) = 0 \ \text{ and } \ \Phi_m(P) \neq 0 \ \text{ for all } 1 \leq m < n\right\}. \end{aligned}$$

Thus, $\mathrm{Per}_n(\phi)$ is the set of points of period n, $\mathrm{Per}_n^*(\phi)$ is the set of points of formal period n, and $\mathrm{Per}_n^{**}(\phi)$ is the set of points of primitive (or exact) period n. This definition will be used to prove proposition theorm1. Beyond that, we also need the definition of the DMC.

Replacing c with y in Eq. (1), (2), (3) and so on, we can obtain affine curve in affine space. That is,

Definition 3 The *dynatomic modular curve* $Y_1(n) \subset \mathbf{A}^2$ is the affine curve defined by the equation

$$\Phi_n{}^*(y,z) = 0.$$

The normalization of the projective closure of $Y_1(n)$ is denoted by $X_1(n)$.Where $\mathbf{A}^2$ represents 2-dimensional affine space.

Indeed, $Y_1(n)$ is a curve in affine space, which satisfies the equation $\Phi_1{}^*(y, z) = 0$. (y, z) represents the coordinates of the curve $Y_1(n)$. When (y, z) is replaced by homogeneous coordinates $\left(\frac{y}{w}, \frac{y}{w}\right)$, we can get projective curve $X_1(n)$ in projective space (w is a constant). That is,

$$X_1(n): \quad \Phi_n{}^*(y, z, w) = 0.$$

$X_1(n)$ is a curve in projective space, and there is a close correspondence between $X_1(n)$ and $Y_1(n)$. Such as, affine curve $Y_1(1)$ satisfies $\Phi_1{}^*(y, z) = 0$, that is, $-z^2 - z + y = 0$. And, projective curve $X_1(1)$ satisfies $\Phi_1{}^*(y, z, w) = 0$, that is, $z^2 + zw - yw = 0$. Similarly, the affine curve $Y_1(2)$ satisfies $z^2 - z + (1 - y) = 0$. And projective curve $X_1(2)$ satisfies $z^2 - zw - yw + w^2 = 0$. In [21], the author has shown that $X_1(1)$and $X_1(2)$ are rational curve. Indeed, $X_1(3)$ is a rational curve too.

Proposition 1 *Let dynatomic modular curve* $Y_1(3)$ *satisfies* $\Phi_3{}^*(y, z) = 0$. $X_1(3)$ *is the normalization of the projective closure of* $Y_1(n)$. *So the* $X_1(3)$ *is a rational curve.*

Proof. In order to parameterize $X_1(3)$, suppose that $\phi(z) = Az^2 - Bz + C$ is any quadratic polynomial with a periodic point of primitive period 3. Note that, as long as the field K does not have characteristic 2, the any quadratic polynomial $\phi(z) = Az^2 - Bz + C$ can be put into the form $(-z^2 + c)$ as the following method. Let $f(z) = \frac{B-2z}{2A}$, which results to $f^{-1}(z) = \frac{B}{2} - Az$. Thus

$$\phi^f(z) = \left(f^{-1} \circ \phi \circ f\right)(z) = -z^2 + \left(\frac{B^2}{4} - AC + \frac{B}{2}\right). \tag{4}$$

Then, conjugating by a linear map $z \mapsto \alpha z + \beta$, we may assume that the given 3-cycle has the form $0 \rightarrow 1 \rightarrow t \rightarrow 0$for some t.This gives the equations

$$\begin{cases} \phi(0) = C = 1\ , \\ \phi(1) = A - B + C = t\ , \\ \phi(t) = At^2 - Bt + C = 0\ . \end{cases}$$

Solving for A, B, C in terms of t yields

$$A = \frac{t^2 - t + 1}{t - t^2}, \quad B = \frac{t^3 - t^2 + 1}{t - t^2}, \quad C = 1.$$

Put A, B and C into the expressions of $\phi^f(z)$ and $f^{-1}(z)$, we obtain

$$\phi^f(z) = -z^2 + \frac{t^6 - 4t^5 + 9t^4 - 8t^3 + 4t^2 - 2t + 1}{4t^4 - 8t^3 + 4t^2},$$
$$f^{-1}(0) = \frac{t^3 - t^2 + 1}{2t - 2t^2}.$$

It shows that for every value of $t \notin \{0, 1\}$, the point

$$P^* = \left(\frac{t^6 - 4t^5 + 9t^4 - 8t^3 + 4t^2 - 2t + 1}{4t^4 - 8t^3 + 4t^2}, \frac{t^6 - 4t^5 + 9t^4 - 8t^3 + 4t^2 - 2t + 1}{4t^4 - 8t^3 + 4t^2}\right),$$

is a solution to the equation

$$\Phi_3^*(y,z) = z^6 - z^5 + (1-3y)z^4 + (2y-1)z^3 + \left(3y^2-3y+1\right)z^2 + \left(-y^2+2y-1\right)z - \left(y^3-2y^2+y-1\right) = 0\,.$$

It also can be verified by using a computer. That is to say, $P^* \in \mathrm{Per}_3^*(\phi)$ and it is formal period 3. We have thus constructed a nonconstant rational map

$$\mathbb{P}^1 \to X_1(3),\quad t \mapsto \left(\frac{t^6-4t^5+9t^4-8t^3+4t^2-2t+1}{4t^4-8t^3+4t^2}, \frac{t^3-t^2+1}{2t-2t^2}\right). \tag{5}$$

That is, rational map can be obtained for $t \notin \{0, 1\}$ from (5). Thus, $X_1(3)$ is a rational curve.

Moreover, $X_1(3)$ is birational to $\mathbb{P}^1$ based on Lüroth's theorem [6]. For it, we can make further explanation. Assuming that (c, b) be a root of Φ_3^*, and set $g(z) = (-b^2 + c - b)z + b$. So, g sends 0 to b and 1 to $\phi(b)$. Then

$$g^{-1}(z) = \frac{z-b}{-b^2-b+c},$$

and

$$\phi^g(z) = \left(g^{-1}\circ\phi\circ g\right)(z) = -\left(-b^2-b+c\right)z^2 - 2bz + 1. \tag{6}$$

It can be computed that $\phi^g(0) = 1$, $\phi^g(1) = b^2 - b - c + 1$ from (6), and

$$\phi^g\left(b^2-b-c+1\right) = b^6 - b^5 + (1-3c)b^4 + (2c-1)b^3 + \left(3c^2-3c+1\right)b^2 - \left(c^2-2c+1\right)b - \left(c^3-2c^2+c-1\right)\,.$$

Since (c, b) is a root of Φ_3^*, that is to say $\Phi_3^*(c, b) = 0$. Putting (c, b) into Φ_3^*, we may obtain

$$\phi^g\left(b^2-b-c+1\right) = \Phi_3^*(c,b) = 0.$$

Thus we can get the 3-cycle for ϕ^g, that is, $0 \to 1 \to b^2 - b - c + 1 \to 0$. Compared with the above 3-cycle, it gives the map

$$X_1(3) \to \mathbb{P}^1,\quad (c,b) \mapsto b^2-b-c+1, \tag{7}$$

which is inverse to (5), this can also be checked directly with a computer yet. For example, let

$$y(t) = \frac{t^6-4t^5+9t^4-8t^3+4t^2-2t+1}{4t^4-8t^3+4t^2},\quad z(t) = \frac{t^3-t^2+1}{2t-2t^2}.$$

When $t = 0.35$, the correspondence $y = 5.3964$, $z = 2.7083$. At this time,$z^2 - z - y + 1 = 0.35 = t$.

Since (7) is the inverse mapping of (5), there is a one to one mapping between $\mathbf{P}^1$ and $X_1(3)$. This mapping can be used to encrypt and decrypt private data. This scheme has the following two advantages.

(a) Increasing infinity point and infinity line can reduce effectively calculation error, and improve the calculation accuracy.

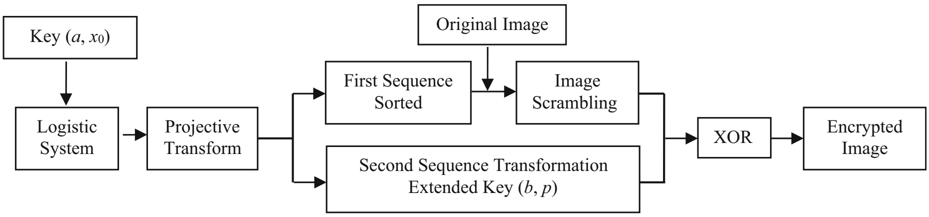


Fig. 1 Block diagram of the encryption algorithm

For example, let $t_1 = 0.000012$. When reserved four decimal places, it will produce larger rounding error by direct calculation in affine space. However, in projective space, we can let $w = 0.00001$. Then we can obtain $t_1{}^* = 1.2$, which will reduce calculation error if we use this value to calculate.

(b) It can help us expand or compress the scope of data by choosing an appropriate value of w.

For example, given arbitrarily $t_2 \in (0, 1)$, we may choose $w = 0.01$, and it makes $t_{2*} \in (0, 100)$. Similarly, if we choose $w = 100$, the scope of t will be compressed to $(0, 0.01)$. This property can be used to adjust the scope of chaotic attractor according to the requirement of the problem.

However, due to the existence of periodic points, it is not enough to use the above properties to design the encryption algorithm. In order to get rid of periodic points, we must build the relationship between the Logistic map and the DMC.

3 The relationship between logistic map and dynatomic modular curve

In this section, the relationship between the Logistic map and the DMC is discussed. In addition, the periodic points are analyzed, in order to ensure the security of the proposed algorithm.

Now we see that the expression of $\phi(z) = -z^2 + c$ play an important role for dynatomic polynomial $\Phi_n^*(z)$. It has close relationship for Logistic map.

$$z(n+1) = az(n)\cdot[1-z(n)], a\in[0,4], z(n)\in(0,1) \tag{8}$$

Given some parameters and initial values, some sequences can be obtained from (8), which would be used in the next section. Equation (8) can be described by $\psi(z) = a(z - z^2)$, and it can be transformed into the form $-z^2 + c$ based on the process of proposition1. Such as, let $h(z) = \frac{1}{2} + \frac{1}{2}$, thus $h^{-1}(z) = -\frac{a}{2} + az$.

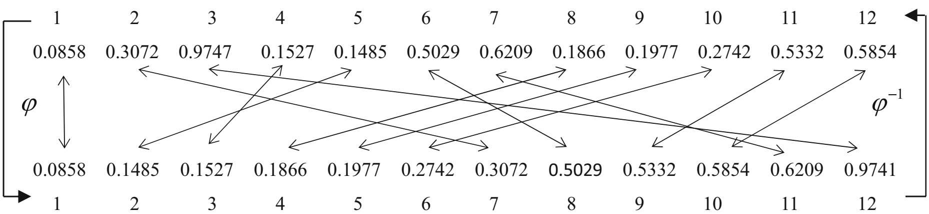
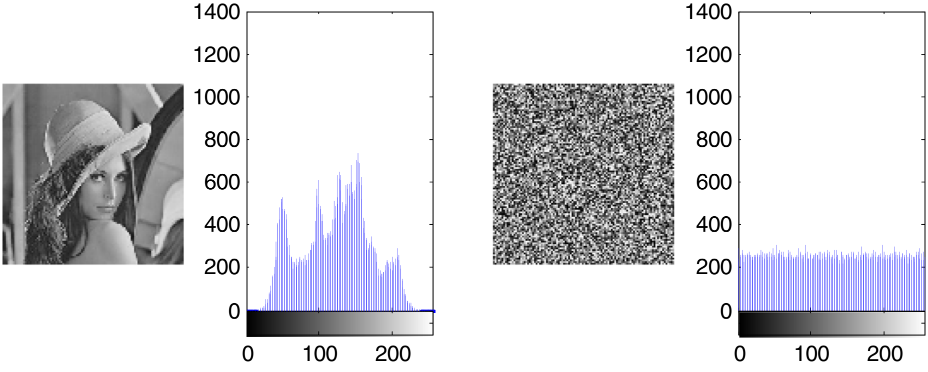


Fig. 2 Index transform diagram between original and ascending order



(a) Original Lena image and its histogram (b) Encrypted Lena image and its histogram

Fig. 3 The histograms of the original and encrypted gray image (256 × 256)

Then

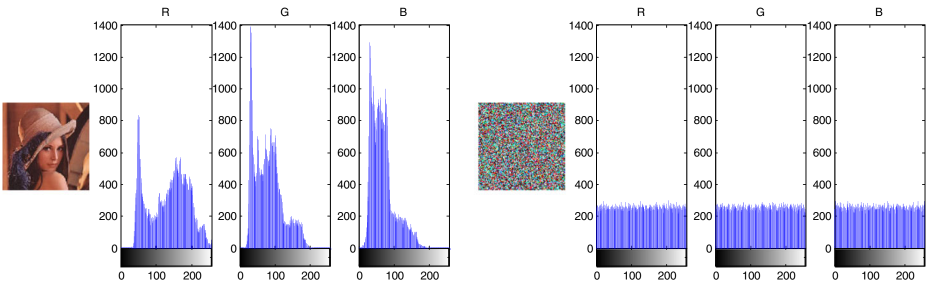
$$\psi^h(z) = \left(h^{-1}\circ\psi\circ h\right)(z) = -z^2 + \left(\frac{a^2}{4} - \frac{a}{2}\right).$$

The expression of $\psi^h(z)$ is consistent with (4). Therefore, we can analyze Logistic map as the method in section 2. That is to say, when a and t satisfies the equation

$$\frac{a^2}{4} - \frac{a}{2} = \frac{t^6 - 4t^5 + 9t^4 - 8t^3 + 4t^2 - 2t + 1}{4t^4 - 8t^3 + 4t^2}, \tag{9}$$

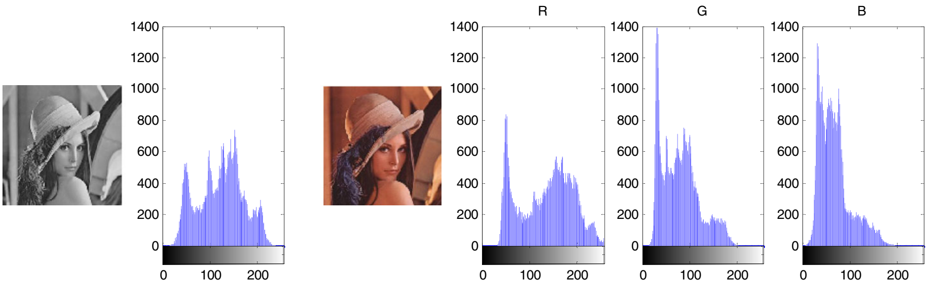
we can obtain rational map (5) and (7). That is, Logistic map can be considered a special kind of the DMC based on Eq. (9).

Let $a = 3.8355$, then $t = -0.7356$ from (9). It makes $(1.76, 0.0238)$ is the solution to $\Phi_3^*(y, z) = 0$. So, 0.0238 is the formal period 3 point of ψ. According to Li-Yorke theorm, ψ can produce chaos when $a = 3.8355$. Thus, the chaotic state of the Logistic map is verified from the perspective of the DMC. Since $0.0238 \in (0, 1)$, we should get rid of this periodic point in encryption algorithm. Therefore, for arbitrary parameters $a \in [0, 4]$, we need remove the periodic point which belongs to $(0, 1)$ to improve the security of the algorithm. But, when $a = 4$, there



(a) Original Lena image and its histogram (b) Encrypted Lena image and its histogram

Fig. 4 The histograms of the original and encrypted color image (256 × 256)



(a) Decrypted gray image and its histogram (256 × 256) (b) Decrypted color image and its histogram (256 × 256)

Fig. 5 The histograms of the decrypted images

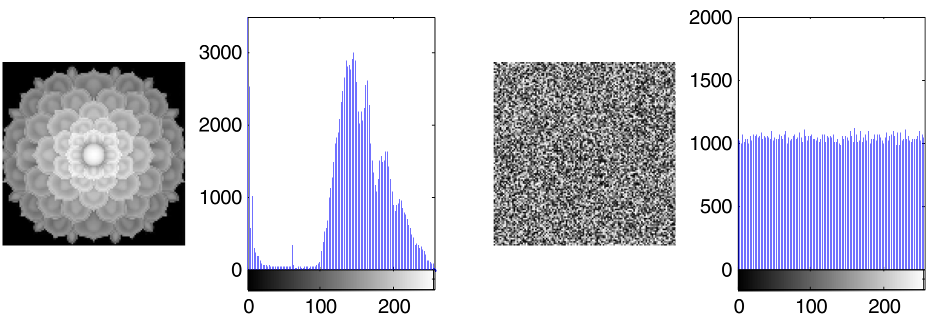
exists $t = 2.8794$, which makes (2, 1.5321) is the solution to the equation $\Phi_3^*(y, z) = 0$. So, 1.5321 is also the formal period 3 point of ψ. Since $1.5321 \notin (0, 1)$ in Logistic map, we can use sequences to encrypt the image directly, instead of eliminating it in advance. Anyhow, the processes of removing the periodic points need to be discussed in design of algorithm.

In order to obtain more secure encrypted image, not only we should get rid of those periodic points, but also ensure that sequence after transformation is far from them. Therefore, an adjustment coefficient b is introduced into the encryption algorithm.

4 Image encryption based on chaos and dynatomic modular curve

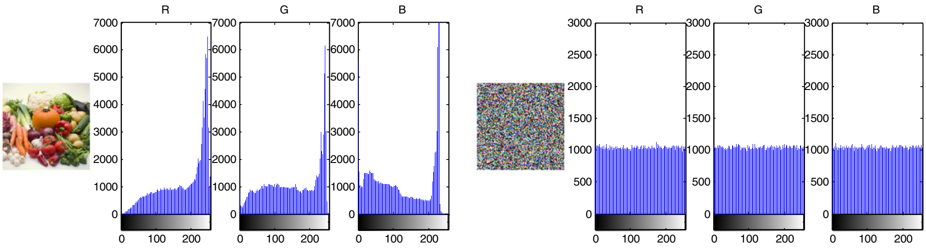
4.1 Image encryption flowchart

From Fig. 1, we can see that the proposed algorithm mainly adopts Logistic map, and the most important step is projective transform. After it, we can obtain two sequences. One is used to scramble the original image pixels and another is used to replace pixel values.



(a) Original filtration image and its histogram (b) Encrypted filtration image and its histogram

Fig. 6 The histograms of the original and encrypted gray image



(a) Original vegetables image and its histogram (b) Encrypted vegetables image and its histogram

Fig. 7 The histograms of the original and decrypted color image (512 × 512)

4.2 Encryption and decryption algorithm

According to the flowchart, the process of the proposed algorithm can be summarized as follows:

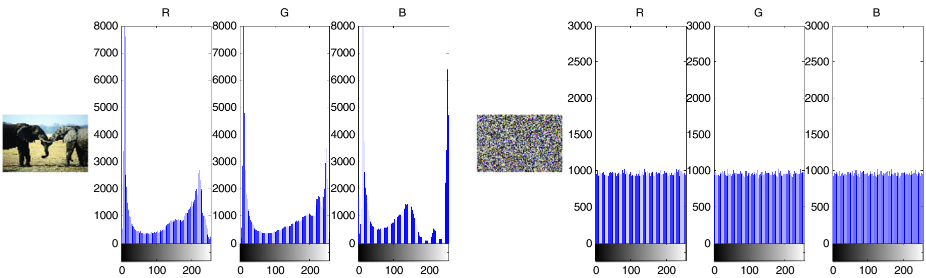
Step 1: Set effective parameter and initial value, and one sequence $T = \{t_1, t_2, \cdots, t_N\}$ can be generated from the Logistic map. $N = m \times k$ is the length of the sequence, where m is the number of rows of the image matrix and k is the number of columns of the image matrix.

Step 2: Projective transform. Suppose $t_i \in T$, $i = 1, 2, \cdots, N$.Using the projective transform (5), t_i is changed into (y_i, z_i), y_i forms a set Y. And another set $Z = \{z_i^{'}\}$ can be obtained by the following function.

$$z_i^{'} = \lfloor 10^p \cdot (bz_i - \lfloor bz_i \rfloor) \rfloor \bmod 256 \quad (10)$$

where b is called adjustment coefficient and p is called amplification parameter. $z_i (i = 1, 2, \cdots, N)$ can be converted to some other elements which belong to $[0, 255]$ by Eq. (10).

Step 3: Image scrambling. Assume that the original image matrix $X = \{x_{ij} | i = 1, 2, \cdots, m; j = 1, 2, \cdots, k\}$.



(a) Original elephant image and its histogram (b) Encrypted elephant image and its histogram

Fig. 8 The histograms of the original and encrypted color image (600 × 400)

Table 1 The results of Chi-square test for gray image ($\alpha = 0.05$)

Image	Chi-square value	df	Progressive significance
Lean	256.953	255	0.454
Filtration	251.080	255	0.558

Then the elements of Y are sorted by the ascending order. In fact, this rearrangement is a kind of transformation. We can rearrange original image pixels according to this transformation function. The pixel set after scrambling is denoted as $X' = \left\{x'_{ij} | i = 1, 2, \cdots, m; j = 1, 2, \cdots, k\right\}$.

Step 4: Pixel replacement. Performing the operation of XOR between the sequence Z and the scrambled image sequence X' as follows:

$$y_{ij} = z'_i \oplus x'_{ij}. \tag{11}$$

Then, we obtain a set $Y' = \{y_{ij} | i = 1, 2, \cdots, m; j = 1, 2, \cdots, k\}$, which is the pixel set of the encrypted image.

The decryption is the inverse process of the encryption. Firstly, do the operations as step 1 to 2 in the encryption algorithm. Secondly, the XOR operation is performed between the encrypted image data and the sequence Z. Finally, the scrambling operation is performed by using of index inverse transform function and reconstructs the original image.

Figure 2 shows the index transform function and its inverse function for some data.

From Fig. 2, the index of the sorted elements has a corresponding relationship with the original elements. Assume φ represents the function between the indexes, we can obtain that

$$\begin{aligned} &\varphi(1) = 1, \varphi(2) = 7, \varphi(3) = 12, \varphi(4) = 3, \varphi(5) = 2, \varphi(6) = 8, \\ &\varphi(7) = 11, \varphi(8) = 4, \varphi(9) = 5, \varphi(10) = 6, \varphi(11) = 9, \varphi(12) = 10. \end{aligned}$$

And

$$\begin{aligned} &\varphi^{-1}(1) = 1, \varphi^{-1}(2) = 5, \varphi^{-1}(3) = 4, \varphi^{-1}(4) = 8, \varphi^{-1}(5) = 9, \varphi^{-1}(6) = 10, \\ &\varphi^{-1}(7) = 2, \varphi^{-1}(8) = 6, \varphi^{-1}(9) = 11, \varphi^{-1}(10) = 12, \varphi^{-1}(11) = 7, \varphi^{-1}(12) = 3. \end{aligned}$$

This algorithm can make full use of the advantages of the DMC to improve the security of the encrypted image, and it can reduce calculation error. These conclusions can be verified by the following experiments.

Table 2 The results of Chi-square test for color image ($\alpha = 0.05$)

Image	R		G		B	
	C-s value	Sig.(two-sides)	C-s value	Sig.(two-sides)	C-s value	Sig.(two-sides)
Lena	219.219	0.949	241.313	0.722	251.078	0.558
Vegetables	203.036	0.993	256.203	0.462	234.500	0.817
Elephant	262.582	0.459	253.195	0.538	248.079	0.592

Table 3 Computational time of five different images

Image	Lena(gray)	Lena(color)	Filtration	Vegetables	Elephant
Size	256 × 256	256 × 256	512 × 512	512 × 512	600 × 400
Encryption time(s)	0.3579	0.3753	0.9587	1.1312	1.0279
Decryption time(s)	0.3839	0.3935	1.1410	1.1871	1.1091

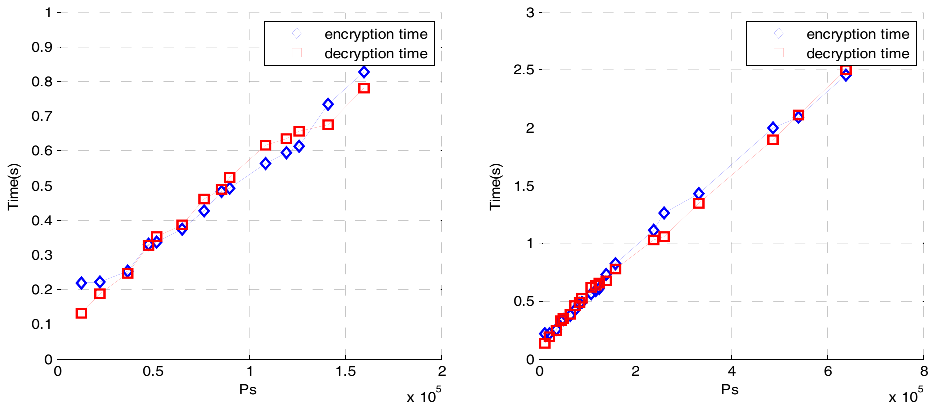
5 Experimental results

Five images are taken for testing effectiveness of the proposed algorithm and they are two gray images (256 × 256,512 × 512) and three color images (256 × 256,512 × 512,600 × 400). R represents red, G represents green, and B represents blue in color images. The histograms of the original image and encrypted image are shown in Figs. 3, 4, 6, 7, and 8. The histograms of decrypted image are shown in Fig. 5. Choose parameters $a = 3.8355$, $p = 10$, $b = 2.345$ and initial value$z(0) = 0.58$.

From Figs. 3, 4, 6, 7, and 8, histograms of the original image and the encrypted image are very different. The histogram of encrypted image is uniform distribution. Indeed, it can be verified by Chi-square test for each image. Here, we take the gray image as an example to verify this conclusion.

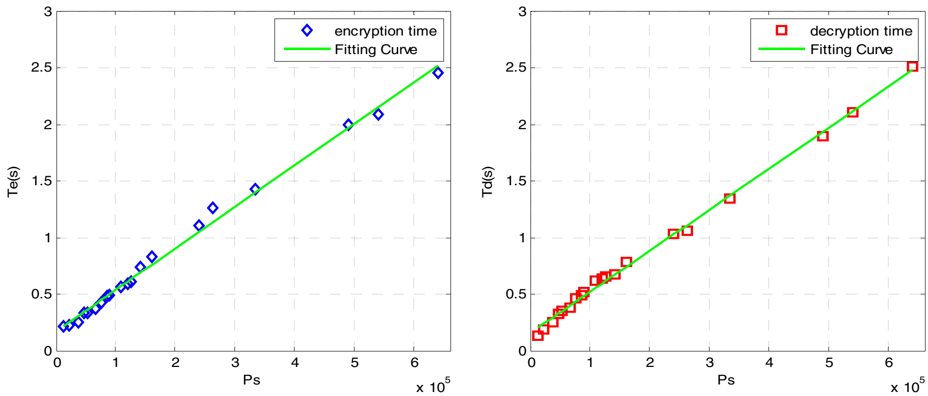
H_0: Encrypted pixel values of the gray image obey uniform distribution.
H_1: Encrypted pixel values of the gray image not obey uniform distribution.

Assume that n_1 represents the number of samples, $X_i(i = 1, 2, \cdots, n_1)$ represent random variable of encrypted gray values, and all samples are divided into m_1 groups, $f_j(j = 1, 2, \cdots, m_1)$ represents



(a) P_s -T_e and P_s -T_d of fourteen images (b) P_s -T_e and P_s -T_d of all images

Fig. 9 Encryption and decryption time of different images



(a) P_s-T_e fitting curve (b) P_s-T_d fitting curve

Fig. 10 The fitting curve of encryption and decryption time for different images

frequency of group j. Let $\overline{X}$ represents the mean value, θ and ω are parameters of this distribution. Then, the estimated value of θ and ω can be obtained by the point estimate method.

$$\hat{\theta} = \overline{x} - \sqrt{\frac{3}{n_1}\sum_{i=1}^{n_1}\left(x_i - \overline{x}\right)^2} = 0,$$
$$\hat{\omega} = \overline{x} + \sqrt{\frac{3}{n_1}\sum_{i=1}^{n_1}\left(x_i - \overline{x}\right)^2} = 256.$$

The probability $p_j\{X=j\}, j=1,2,\cdots,m_1$ can be calculated as uniform distribution which parameters have been known. Finally, we can make a decision through comparing the value of

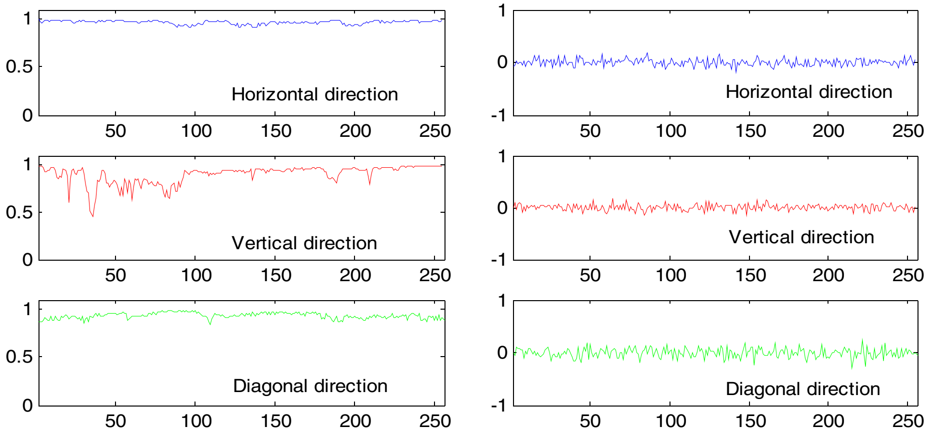
$$\chi^2 = \sum_{j=1}^{m_1}\frac{\left(f_j - n_1 p_j\right)^2}{n_1 p_j},$$

with $\chi^2(97)$. Results are listed in Table 1 which derives from SPSS statistics 19.

Table 1 demonstrates the progressive significance $p^* > 0.05$. So, we can't reject H_0.That is to say, encrypted pixel values of the gray image obey uniform distribution. As is known to all, the color image is composed of three pixel matrixes, which are R matrix, G matrix and B matrix. Each pixel matrix can be encrypted and decrypted according to the same

Table 4 Average correlation coefficients of different images

Image	Original-image			Encrypted-image		
	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Lena(gray)	0.968579	0.902024	0.936888	0.001398	0.007793	0.006573
Lena(color)	0.970634	0.944835	0.925606	0.002443	0.002921	0.002067
Filtration	0.984714	0.991562	0.990413	0.002438	0.000097	0.002054
Vegetables	0.975181	0.984653	0.985581	0.002061	0.003577	0.003893
Elephant	0.929295	0.964738	0.975612	0.002859	0.001403	0.001208



(a) Correlation coefficients of the original image (b) Correlation coefficients of the encrypted image

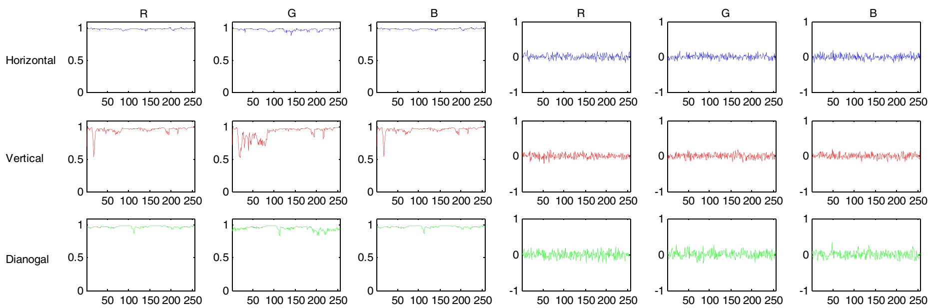
Fig. 11 All the correlation coefficients of gray image (Lena)

processing method of gray image. Then, encrypted image can be got by merging them. So, the proposed scheme is feasible to color image. Moreover, it can also be verified that encrypted pixel values of the color image obey uniform distribution through Chi-square test. Results are listed in Table 2.

Table 2 shows that each progressive significance value P^* is far greater than 0.05. Moreover, all values of progressive significance are greater than 0.454. That is to say, the proposed encryption algorithm works better for color image.

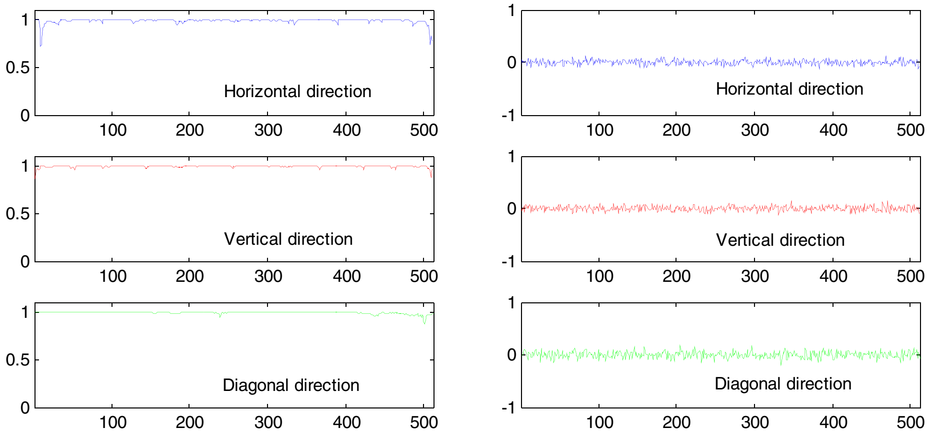
The computational time of these images is listed in Table 3. In addition, other seventeen color images which have different size are selected from the USC-SIPI image database. Encryption and decryption time of all images are shown in Fig. 9. And all the algorithms are calculated by MATLAB R2013a on the same computer with Inter Xeon E5-2630v2/ 16G, DDR3.

Let T_e represents encryption time, T_d represents decryption time, and P_s represents the size of images. Thus, T_e and T_d of different images which size belongs to (10,000, 160,000) are shown in Fig. 9a. T_e and T_d of different images which size belongs to



(a) Correlation coefficients of the original image (b) Correlation coefficients of the encrypted image

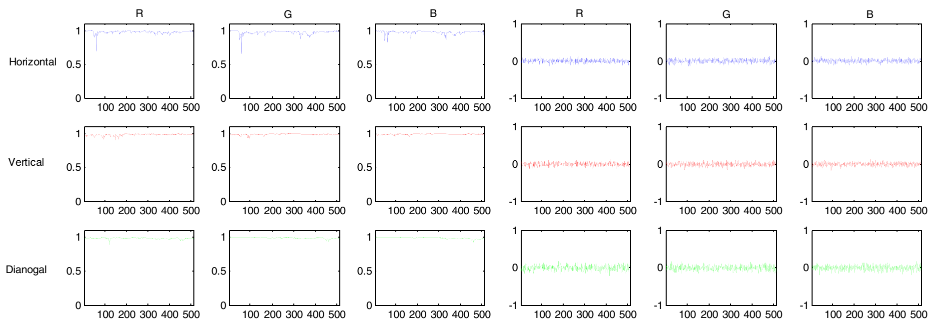
Fig. 12 All the correlation coefficients of color image (Lena)



(a) Correlation coefficients of the original image (b) Correlation coefficients of the encrypted image

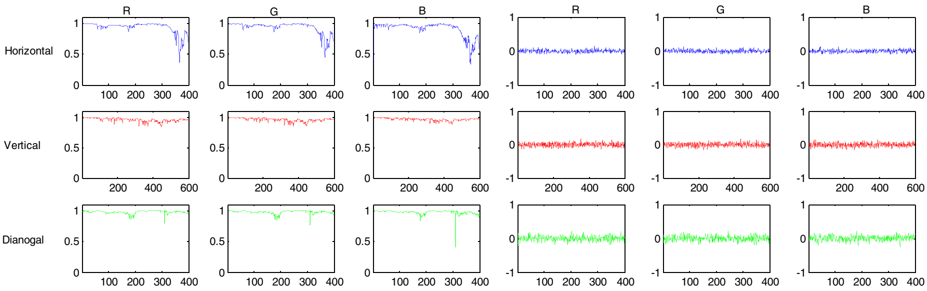
Fig. 13 All the correlation coefficients of gray image (Filtration)

(10,000, 640,000) are shown in Fig. 9b. Figure 9 indicate that T_e and T_d of different images increase with the size of images. Specially, for different size of images, the shortest $T_e = 0.2196s$ and $T_d = 0.1335s$, and the longest $T_e = 2.4534s$ and $T_d = 2.5047s$. Moreover, the relationship between T_e and P_s can be approximately represented by $T_e = 3.6714 \times 10^{-6} P_s + 0.1705$, which derives from fitting (Fig. 10a). Similarly, the relationship between T_d and P_s can be approximately represented by $T_d = 3.6271 \times 10^{-6} P_s + 0.1586$ (Fig. 10b). Thus, the encryption and decryption time of all images in the USC-SIPI image database can be estimated through the above two linear function. For example, a color peppers (512×512) image is selected to verify it, and its file name is 4.2.07 in the USC-SIPI image database. When $P_s = 262144$, we can obtain that $T_e = 1.1329s$ and $T_d = 1.1094s$.They are very close to the experimental results in Table 3. Therefore, the proposed algorithm is applicable to any image of this database. And because of it, we use the above five images as the representative to analysis the performance of the proposed algorithm in the next section.



(a) Correlation coefficients of the original image (b) Correlation coefficients of the encrypted image

Fig. 14 All the correlation coefficients of color image (Vegetables)



(a) Correlation coefficients of the original image (b) Correlation coefficients of the encrypted image

Fig. 15 All the correlation coefficients of color image (Elephant)

6 Performance analysis

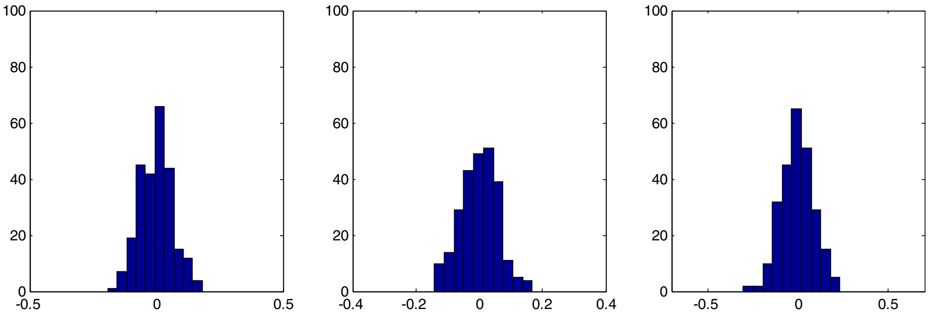
In this section, the performances of the encryption algorithm are measured by calculating the correlation coefficient, the information entropy, the values of *NPCR* and *UACI*. Moreover, the key space, the key sensitivity and the algorithm intensity are discussed.

6.1 Correlation coefficient

Correlation coefficient measures the dependence of two adjacent variables at a certain direction. The closer this value is zero the less correlation exists between two adjacent. Conversely, the value is to 1. The two variables are not relevant and unpredictable when correlation coefficient is close to 0. The calculation formula of correlation coefficient is as follows [9].

$$r=\frac{n\left(\sum_{i=1}^{n}x_iy_i\right)-\left(\sum_{i=1}^{n}x_i\right)\left(\sum_{i=1}^{n}y_i\right)}{\sqrt{\left[n\left(\sum_{i=1}^{n}x_i^2\right)-\left(\sum_{i=1}^{n}x_i\right)^2\right]\left[n\left(\sum_{i=1}^{n}y_i^2\right)-\left(\sum_{i=1}^{n}y_i\right)^2\right]}} \tag{12}$$

where $n\left(\sum_{i=1}^{n}x_iy_i\right)-\left(\sum_{i=1}^{n}x_i\right)\left(\sum_{i=1}^{n}y_i\right)$ represents the sample variation, $\left[n\left(\sum_{i=1}^{n}x_i^2\right)-\left(\sum_{i=1}^{n}x_i\right)^2\right]$



(a) Horizontal direction (b) Vertical direction (c) Diagonal direction

Fig. 16 correlation coefficient's frequency distribution of encrypted gray image (Lena)

Table 5 Single sample K-S test for correlation coefficient of the encrypted images

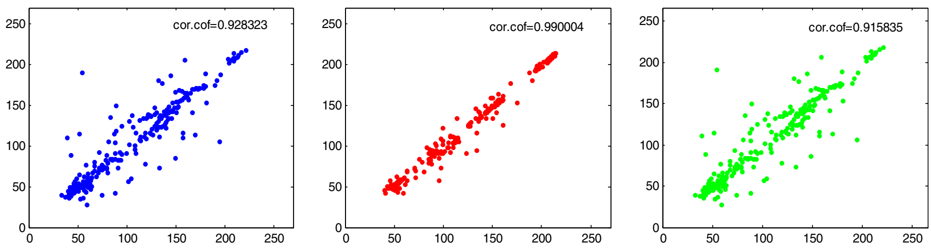
Image	Horizontal			Vertical			Diagonal		
	Mean	STDEV	Sig.(two-sides)	Mean	STDEV	Sig.(two-sides)	Mean	STDEV	Sig.(two-sides)
Lena(gray)	0.0014	0.0644	0.8191	−0.0078	0.0646	0.6543	0.0025	0.0838	0.9645
Lena(color)	0.0015	0.0432	0.6850	0.0021	0.0658	0.7870	0.0043	0.0913	0.7543
Filtration	−0.0024	0.4437	0.6251	−0.0001	0.0471	0.7134	−0.0021	0.0625	0.8337
Vegetables	0.0029	0.0412	0.8567	0.0014	0.0500	0.9078	0.0019	0.0741	0.9608
Elephant	0.0020	0.3005	0.6697	0.0036	0.0440	0.8567	0.0039	0.06187	0.6977

and $[n\left(\sum_{i=1}^{n} y_i^2\right) - \left(\sum_{i=1}^{n} y_i\right)^2]$ are the sample standard variation of X_j and $Y_j(j = 1, 2, \cdots, m)$, respectively.

All correlation coefficients of adjacent pixel of the original and encrypted image are calculated from the horizontal, vertical and diagonal directions. For example, let $X_j = \{x_{j1}, x_{j2}\cdots, x_{jn}\}$ represents one pixel vector of the gray image, the adjacent pixel vector of X_j is $Y_k = \{y_{k1}, y_{k2}\cdots, y_{kn}\}$, which satisfies $k = j + 1$, $j = 1, 2, \cdots, m - 1$. According to (12), the correlation coefficient of X_j and Y_k can be calculated. Finally, the correlation coefficient of the gray image in a fixed direction is obtained by averaging. For color images, the correlation coefficients of R, G and B matrices are calculated respectively. And then their average value which represents the correlation coefficients of the color image is calculated. Thus, the number of correlation coefficient varies with the size of the image. Some computational results are listed in Table 4, which only includes average values. Each value of correlation coefficients is shown in Figs. 11, 12, 13, 14, and 15.

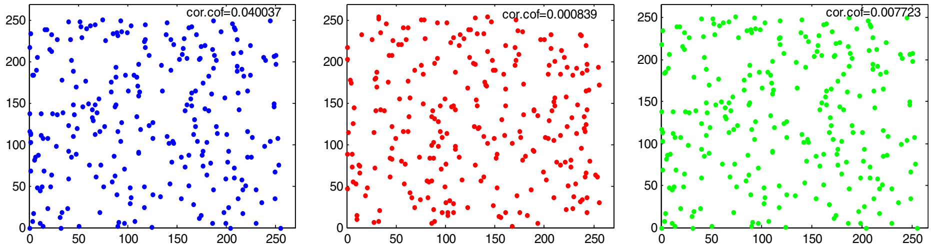
Table 4 shows that correlation coefficients of the different original images in horizontal, vertical and diagonal directions are close to 1. However, they are close to 0 after encryption. More specifically, the correlation coefficient of the original Filtration image is 0.991562 in vertical direction, but it drops to 0.000097 by encrypting. Similar results can be obtained in horizontal and diagonal direction.

The adjacent pixels of the original image have a significant correlation from Figs. 11, 12, 13, 14, and 15a. However, the correlation disappeared after encryption. Moreover, all correlation coefficients of the encrypted image evenly are distributed in the small range of zero from Figs. 11, 12, 13, 14, and 15b. In order to more accurately determine the distribution type of the correlation coefficients in the encrypted image, we describe the frequency charts in different directions in Fig. 16.



(a) Horizontal direction (b) Vertical direction (c) Diagonal direction

Fig. 17 The distribution of two adjacent pixel vectors for the original image (Lena)

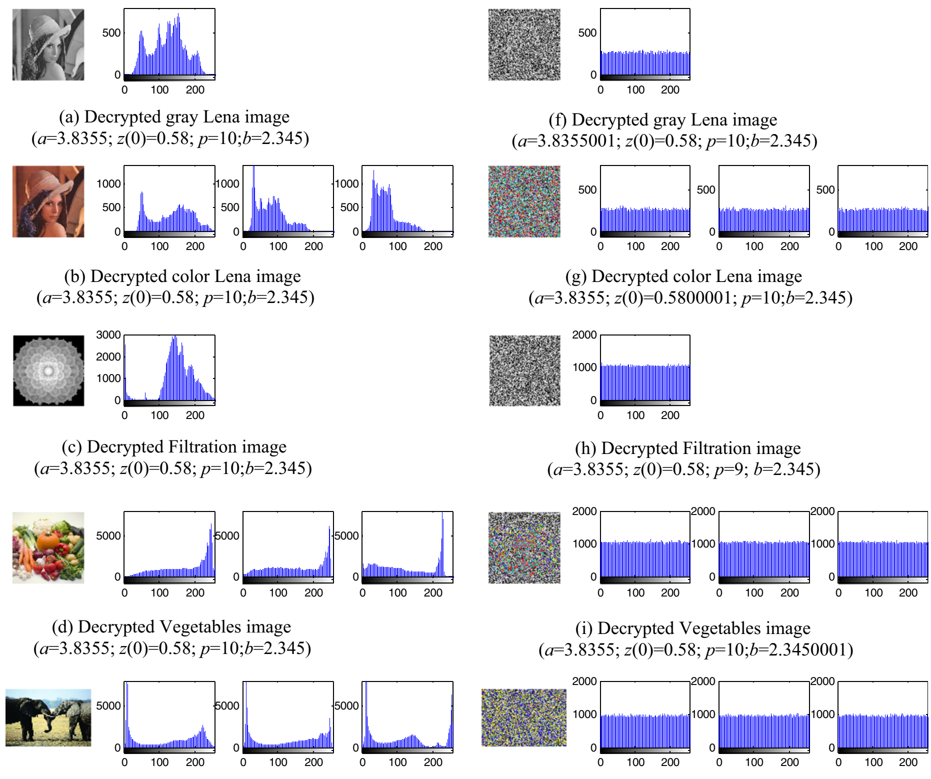


(a) Horizontal direction (b) Vertical direction (c) Diagonal direction

Fig. 18 The distribution of two adjacent pixel vectors for the encrypted image (Lena)

Figure 16 shows that the correlation coefficients of the encrypted image obey normal distribution, and this conclusion can be verified by single sample K-S test. The process of this test is given as follows.

H_0: Correlation coefficients of the encrypted image obey normal distribution.
H_1: Correlation coefficients of the encrypted image don't obey normal distribution.



(a) Decrypted gray Lena image (a=3.8355; z(0)=0.58; p=10;b=2.345)

(f) Decrypted gray Lena image (a=3.8355001; z(0)=0.58; p=10;b=2.345)

(b) Decrypted color Lena image (a=3.8355; z(0)=0.58; p=10;b=2.345)

(g) Decrypted color Lena image (a=3.8355; z(0)=0.5800001; p=10;b=2.345)

(c) Decrypted Filtration image (a=3.8355; z(0)=0.58; p=10;b=2.345)

(h) Decrypted Filtration image (a=3.8355; z(0)=0.58; p=9; b=2.345)

(d) Decrypted Vegetables image (a=3.8355; z(0)=0.58; p=10;b=2.345)

(i) Decrypted Vegetables image (a=3.8355; z(0)=0.58; p=10;b=2.3450001)

(e) Decrypted Elephant image (a=3.8355; z(0)=0.58; p=10;b=2.345)

(j) Decrypted Elephant image (a=3.8355001; z(0)=0.5800001; p=9;b=2.3450001)

Fig. 19 Key sensitivity test for different images

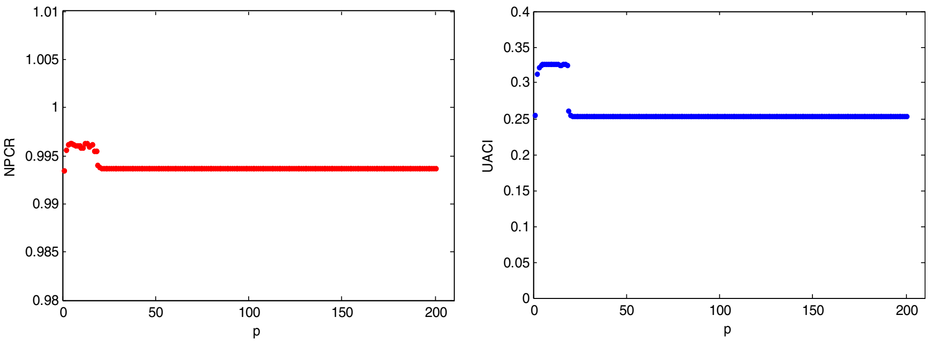


Fig. 20 Simulation of *NPCR* and *UACI* with different *p*

Let x_i represents sample, $F_0(x)$ represents theory distribution function, and $F_{n_2}(x)$ represents sample cumulative frequency function. And let

$$D = \max|F_{n_2}(x) - F_0(x)|.$$

In order to obtain $F_0(x)$, we must estimate μ and σ, which are parameters of normal distribution in H_0. They can be obtained by the following formulas,

$$\hat{\mu} = \frac{1}{n_2}\sum_{i=1}^{n_2} x_i = \overline{x}, \quad \hat{\sigma}^2 = \frac{1}{n_2}\sum_{i=1}^{n_2}\left(x_i - \overline{x}\right)^2.$$

And, $F_{n_2}(x) = F/n_2$, where F represents cumulative frequency, and n_2 represents sample size. When $D > D(n_2, \alpha)$ (α is significance level, here $\alpha = 0.05$), reject H_0. Otherwise, accept H_0.

SPSS is used to test correlation coefficient's distribution. The results of K-S test are listed in Table 5.

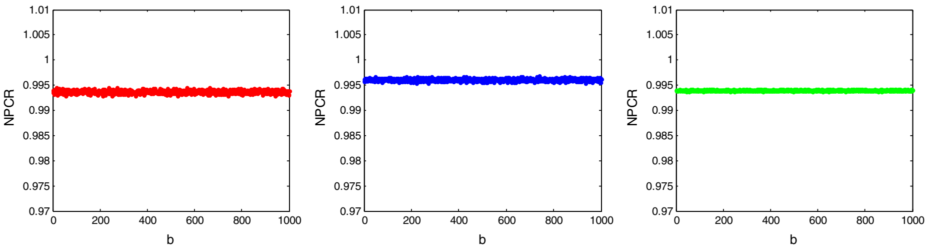
From Table 5, the maximum value of progressive significant is 0.9645 and the minimum value is 0.6251. Each value of progressive significant is greater than 0.05. Moreover, the mean value and standard deviation are close to zero. The results indicate that the correlation coefficients of the encrypted image obey normal distribution.

In addition, we randomly select two adjacent pixel vectors in the original image and the encrypted image. And the distribution of these pixel values is shown in Figs. 17 and 18.

Before encryption, the distribution of two adjacent pixel vectors is close to a straight line. However, the distribution is not regular after encryption. Therefore, it is difficult for the attacker to analyze the distribution law of the image under such an irregular distribution.

Table 6 Values of *NPCR* and *UACI* with different *p*

p	1	2	3	4	5	6	7	8	9	10
NPCR(%)	99.4202	99.6002	99.6063	99.6063	99.6078	99.6017	99.6674	99.5941	99.5911	99.6033
UACI(%)	25.6828	31.6668	32.0038	32.8772	32.2740	32.7265	32.6198	32.7495	32.6196	32.6173
p	11	12	13	14	15	16	17	18	19	20
NPCR(%)	99.6475	99.6124	99.6368	99.6216	99.5911	99.6216	99.6246	99.5987	99.3790	99.3805
UACI(%)	32.6496	32.7576	32.6500	32.5905	32.7043	32.5732	32.5856	32.5895	26.0909	25.6699



(a) p=1 (b) p=10 (c) p=19

Fig. 21 Simulation of *NPCR* with different b and p

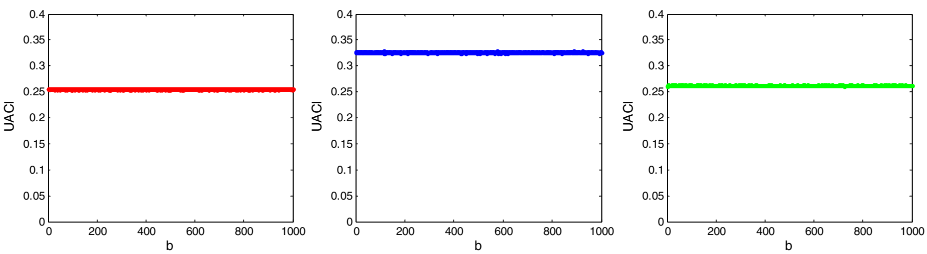
6.2 Resistance statistical analysis

We have done statistical analysis for the proposed algorithm in section 5. The results show that the histogram of encrypted image is uniform distribution. That is to say, the frequency of each pixel value is very close after encryption. The result is quite different from the distribution of the original image. It makes the attacker cannot obtain the statistical law of the encrypted image, which is a precondition for breaking the code. Moreover, Correlation coefficient analysis demonstrates that all correlation coefficients of different encrypted images are close to zero in section 6.1. It makes the attacker cannot predict the original image by analyzing the statistical characteristics of the encrypted image. So, the proposed algorithm can resist statistical attack.

6.3 Key sensitivity

A good encryption algorithm should be very sensitive to the key. A slight variation of the key should result in totally different image in the reconstructing process. Figure 19a–e show some decrypted images with the correct key. Figure 19f–j show some decrypted images with different keys which have a slightly change.

In this algorithm, all the original images can be recovered when using of the correct keys ((Fig. 19a–e). However, when the parameter or the initial value of Logistic map is changed slightly, it can't obtain the original image (Fig. 19f–g). If parameter p is changed to 9 from 10, it can't obtain the original image (Fig. 19h), and if parameter b is changed to 2.3450001 from 2.345, it can't obtain the original image yet (Fig. 19i). If all keys are changed slightly, the original image is more difficult to obtain (Fig. 19j). Moreover, each histogram of decrypted



(a) p=1 (b) p=10 (c) p=19

Fig. 22 Simulation of *UACI* with different b and p

Table 7 Values of *NPCR* and *UACI* for different images

Image(s)	Lena(gray)	Lena(color)	Filtration	Vegetables	Elephant
NPCR (%)	99.6033	99.6139	99.6089	99.6122	99.6040
UACI (%)	32.6173	32.6602	33.2957	34.1178	33.6992

image obeys uniform distribution when used wrong key. So, the proposed algorithm is sensitive to the key.

6.4 NPCR and UACI

NPCR is the comparison of relational positions between original and encrypted images in order to ensure that the pixels of every level matrix can be altered. *UACI* is, on the other hand, the percentage of the average level matrix change between the relational positions of two images [9]. The following equations define *NPCR* and *UACI:*

$$\mathrm{NPCR} = \frac{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} D(i,j)}{M \times N} \times 100\ \%\ ,$$

$$\mathrm{UACI} = \frac{1}{M \times N}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} \frac{\left|A(i,j) - A_{CS}\left(i,j\right)\right|}{255} \times 100\%,$$

and

$$D(i,j) = \begin{cases} 0, & A(i,j) = A_{CS}(i,j)\ , \\ 1, & A(i,j) \neq A_{CS}(i,j)\ \ , \end{cases}$$

where A is the original image of $M \times N$ dimension, and A_{cs} is the encrypted image. M and N are width and height of the image. Taking the gray image as an example, we discuss the influence of the parameter p on the *NPCR* and *UACI*. Figure 20 shows that the values of *NPCR* and *UACI* are different for different parameter p. It has a fixed trend when $p \to \infty$. That is to say, *NPCR* and *UACI* are invariable for some p. A few precise values of *NPCR* and *UACI* are listed in Table 6.

From Table 6, there exists large difference when $p \geq 19$, which is caused by the calculation accuracy. Moreover, experiments show that each pixel value of encrypted image is equal after $p = 24$, which also consists of 8-bit integer and 16-bit decimal numbers. In this time, Eq. (10) will fail. That is, the proposed algorithm only contains scrambling process without replacement. Thus, the value of *NPCR* is 99.3790% which stay away from the ideal value. The same reason is for *UACI*. In addition, the values of *NPCR* and *UACI* are unsatisfactory when $p = 1$. Moreover, it is found that they are still close to the ideal values when p isn't an integer by

Table 8 Entropy values of different encrypted images

Image(s)	Lena(gray)	Lena(color)	Filtration	Vegetables	Elephant
Entropy	7.8571	7.8679	7.8568	7.9130	7.9583

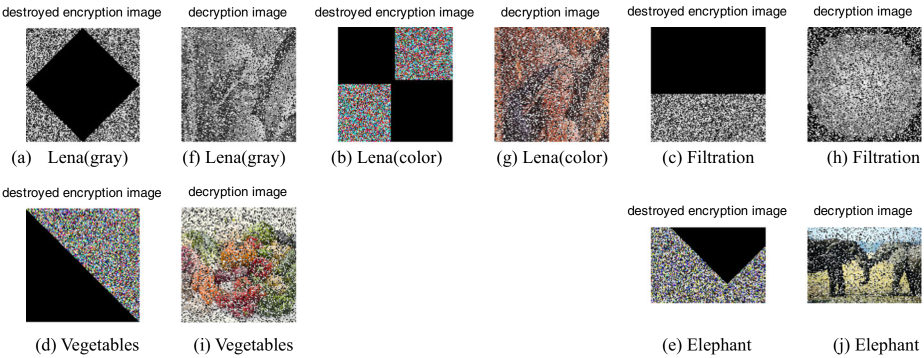


Fig. 23 Algorithm intensity test for different images

experiments, such as $p = 9.5812$. At this time, the information of the encrypted image is completely different from that of the other p. So, we should choose $p \in (2, 18)$ to achieve the best encryption effect in our computer. If our algorithm can be combined with cloud computing, the scope of p will dramatically increase.

We also discuss the influence of adjustment coefficient b on *NPCR* and *UACI* for different p. Values of *NPCR* and *UACI* are shown in Figs. 21 and 22.

When $p = 10$, the average value of *NPCR* and *UACI* is 99.6145% and 32.5906%, respectively. They have litter change for different b. We can get the same conclusion by simulation experiments when $p \in (2, 18)$ and $b \in R^*$. When $p \geq 19$, the average value of *NPCR* and *UACI* remains unchanged with different b. But, the difference between them and the ideal value becomes lager duo to the role of the parameter p. Therefore, the values of *NPCR* and *UACI* are not affected by the adjustment coefficient b. Other values of *NPCR* and *UACI* are listed in Table 7 for different images when$p = 10$, $b = 2.345$.

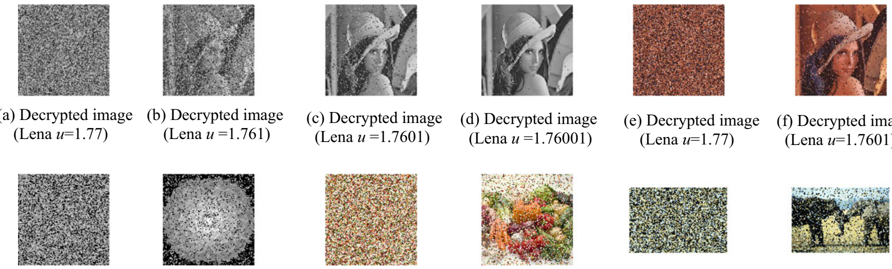
According to the principle of cryptography, a good encryption algorithm should be fully sensitive to the clear text. This sensitivity is stronger; the ability to resist differential attack will be stronger. The sensitivity of the encryption algorithm to the clear text can be characterized by the number of pixels change rate (*NPCR*) [4]. That is to say, *NPCR* is an important measure index of resisting differential attack. From Table 7, the values of *NPCR* are very close to the ideal value 99.6094% when $p = 10$ and $b = 2.345$. The results show that the information of original images has a good spread to encryption image and the proposed algorithm can resist differential attack.

6.5 Information entropy

The information entropy of image actually measures the distribution of gray value in the image. Greater information entropy represents higher uniformity of the images. Generally, an

Table 9 Comparison results of correlation coefficient for gray image (Lena)

Methods	Ours	Huang's [8]	Lin's [13]	Ye's [27]	Zhang's [30]	Zahra's [17]
Horizontal	0.0014	−0.0974	0.0242	0.0770	0.0012	−0.0018
Vertical	0.0078	−0.0707	0.0194	−0.0724	0.0156	0.0345
Diagonal	0.0066	0.0484	0.0343	−0.0615	0.1326	0.0202



(a) Decrypted image (Lena u=1.77) (b) Decrypted image (Lena u =1.761) (c) Decrypted image (Lena u =1.7601) (d) Decrypted image (Lena u =1.76001) (e) Decrypted image (Lena u=1.77) (f) Decrypted image (Lena u=1.7601)

(g) Decrypted image (Filtration u=1.77) (h) Decrypted image (Filtration u=1.7601) (i) Decrypted image (Vegetables u=1.77) (j) Decrypted image (Vegetables u=1.7601) (k) Decrypted image (Elephant u=1.77) (l) Decrypted image (Elephant u=1.7601)

Fig. 24 Testing of key sensitivity for other algorithms

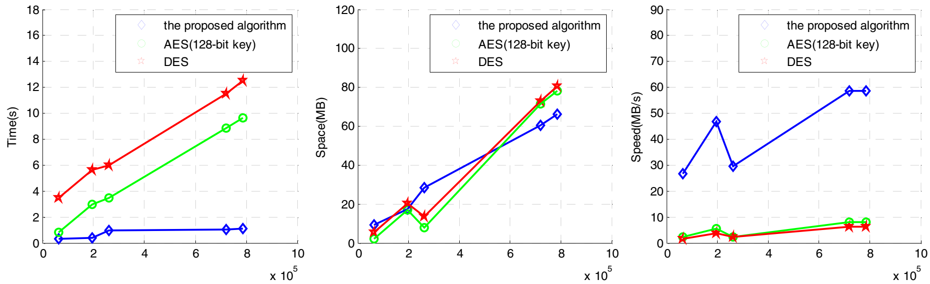
ideal value of information entropy is approaching to 8 for an image after encryption. The closer the entropy of an encryption algorithm is to 8 the less predictable, and this scheme is more secure. It is defined as follows [29]:

$$H(m) = \sum_{i=1}^{2^n-1} p(m_i)\log 2\left(\frac{1}{p(m_i)}\right),$$

where m_i is i-th gray value of the image, $p(m_i)$ represents the probability of occurrence of m_i.The information entropy of different encrypted images are calculated and the results are listed in Table 8. It shows that the entropy value of the encrypted image is close to the ideal value, especially for color images.

6.6 Key space

In our algorithm, the parameter value of the Logistic map $a = 3.8355$ is used as secret key. So, we need 32 bits to store this value (single float number in MATLAB). Another 32 bits is needed for storing initial value 0.58. Finally, adjustment parameter $b = 2.345$ and amplification parameter $p \in (2, 18)$ are also used as secret key. We need 64 bits to store them according to the analysis of parameter b and p in subsection 6.4. Therefore, the total number of bits used to store all the key values is 128. Thus, the cryptosystem has at least 2^{128} different combinations and this large key space is enough to resist brute force attack.



(a) P_s-T_e of different algorithms (b) P_s–S_{pa} of different algorithms (c) P_s–S_{pe} of different algorithms

Fig. 25 Comparison of computational complexity for different algorithms

6.7 Algorithm intensity analysis

A malicious attacker may destroy the encrypted image, and the legitimate receiver can decrypt the image successfully. Destroyed encryption images are shown in Fig. 23a–e, and the corresponding decrypted images are shown in Fig. 23f–j.

Figure 23 demonstrates that, even if an encrypted image is destroyed by an attacker, the legitimate receiver can decrypt the image successfully, only noise exists. Hence, the encryption algorithm can resist illegal tampering.

6.8 Comparison with some existing encryption algorithms

6.8.1 Comparison of correlation coefficient

In order to reflect the advantages of the proposed algorithm in terms of security, comparison of the correlation coefficient between original image and encrypted image is preformed in Table 9. Better results are achieved than most of the schemes mentioned in this paper, such as multi-chaotic system based schemes [8], bit level permutation based schemes [13], chaotic system based schemes [27], DNA computing based schemes [30], and secure image encryption based schemes [17].

From Table 9, the average value of correlation coefficient of the proposed algorithm is 0.0053, which is smaller than Huang's, Lin's, Ye's, Zhang's and Zahra's. This means that, the attacker will be more difficult to discover the distribution law of the encrypted image when compared with the above algorithms, and the proposed algorithm has better effect in resistance statistical attack.

6.8.2 Comparison of key sensitivity

Since the key sensitivity is an important index of the security, we also compare it with the same algorithm without the projective transformation in reference [16], which used Henon map to encrypt image. Our results are shown in Fig. 19. Their results are shown in Fig. 24.

From Fig. 24, we can't get the original image when parameter u = 1.77(the correct parameter u = 1.76 in [16]). However, we can obtain main information of the original image when the value of u is changed very small, such as u = 1.7601(Fig. 24c, f, h, j, l). It means that the main information has been leaked when differences reach 10^{-4} between the correct and the wrong key. There is no doubt that the security of algorithm is not guaranteed to resist brute force attack. But, the attacker could not obtain any information about the original image when differences reach 10^{-15} in our algorithm. The results show that the proposed algorithm has a higher security.

6.8.3 Comparison of computational complexity

Apart from the security consideration, some other issues on an image encryption scheme are also important, including the computational complexity which is composed of time complexity and space complexity, particularly for real-time Internet applications. The computational complexity of the proposed encryption algorithm depends on data generating operation, projective transform operation, image scrambling operation, and pixel replacement operation. The complexity of the first two steps for an image is O(n), and the latter two steps is O(n^2). So the total computational complexity is O(n^2). Specifically, the execution time of the above five images is faster than some well-known encryption algorithms [1, 11] in the same MATLAB R2013a platform. The results of execution time for different algorithms are shown in Fig. 25a. In addition, the occupied space is also being compared, and the results are shown in Fig. 25b.

Figure 25a shows that T_e of the proposed algorithm are faster than some other algorithms, such as AES and DES. From Fig. 25b, when the size of color image is 256 × 256, the occupied space of all algorithms is very close to 18.5642 MB. Moreover, the advantage of the proposed algorithm is further strengthened with the increase of the image size. Generally, the occupied space of our algorithm is lower than other algorithms when the size of the image is more than750 × 750.

In order to compare the running speed of the algorithms, we let $S_{pe} = S_{pa}/T_e$. Where S_{pe} represents the running speed, S_{pa} represents the occupied space. Thus, the running speed of the proposed algorithm is faster than other algorithms (Fig. 25c). Concretely, the average running speed of the proposed algorithm is 44.0836 MB/s, and it is much higher than 14.4591 MB/s which is the maximum of the AES and DES algorithm. Therefore, our algorithm is more suitable for image encryption.

7 Conclusions

In this paper, we have proposed a secure and effective encryption algorithm for images based on Logistic map and the DMC. One of the most benefits of the proposed algorithm is that we increase the diffusivity by using of projective transformation. It makes our algorithm has many good performances, which can resist statistical attack, brute force attack and differential attack.

In the future, we will continue to discuss the following problems.

(1) Research the relationship between Henon map and dynatomic modular curve.
(2) Applying our scheme to the video encryption algorithm, and evaluating its performance.

Acknowledgements This work was supported by the National Key Research and Development Program of China 2016YFB0800601.

References

1. Chen G, Mao YB, Chui CK (2004) A sysmmetric image encryption scheme based on 3D chaotic cat maps. Chaos, Solution & Fractals 12:749–761
2. Chen WH, Luo S, Zheng WX (2016) Impulsive synchronization of reaction-diffusion neural networks with mixed delays and its application to image encryption. IEEE Transactions on Neural Networks and Learning Systems 33(1):300–303
3. Chhotaray A, Biswas S, Chhotaray SK, Rath GS (2015) An image encryption technique using orthonormal matrices and chaotic maps. Proceeding of 3rd International conference on advanced computing. Networking and Informatics 9(44):355–362
4. Cong-Xu Z, Ke-Hui S (2012) Cryptanalysis and improvement of a class of hyper-chaos based image encryption algorithms. Acta Phys Sin 61(12):1–12
5. Fu Z, Sun X, Liu Q, Zhou L, Shu J (2015) Achieving efficient cloud Search services: multi-keyword ranked Search over encrypted cloud data supporting Parallel computing. IEICE Trans Commun E98-B(1):190–200
6. R. Hartshorne (1977) Algebraic geometry. Springer-Verlag, New York. GraduateTexts in Mathematics, 52

7. Hua T, Chen J, Pei D, Zhang W, Zhou N (2015) Quantum image encryption algorithm based on image correlation decomposition. Int J Theor Phys 54:526–537
8. Huang X (2012) Image encryption algorithm using chaotic chebyshev generator. Nonlinear Dyn 67:2411–2417
9. Huang CK, Liao CW, Hsu SL, Jeng YC (2013) Implementation of gray image encryption with pixel shuffling and gray-level encryption by single chaotic system. Telecommun Systems 52(2):563–571
10. Kamara S, Lauter K (2010) Cryptographic cloud storage. In: Financial cryptography and data security. Springer. pp. 136–149
11. Kwok HS, Tang WKS (2007) A fast image encryption system based on chaotic maps with finite precision representation. Chaos, Solution & Fractals 32:1518–1529
12. Li S, Zheng X (2002) Cryptanalysis of a chaotic image encryption method. Proceedings of the IEEE International Conference on Circuits and Systems 2:708–711
13. Lin T, Xingyuan W (2012) A bit-level image encryption algorithm based on spatiotemporal chaotic system and self-adaptive. Opt Commun 2012:940276
14. Liu L, Miao S (2016) A new image encryption algorithm based on logistic chaotic map with varying parameter. Spring 5(1):1–12
15. Liu H, Wang X (2010) Color image encryption based on onetime keys and robust chaotic maps. Comput Math Appl 59(10):3320–3327
16. Manjunath Prasad, K.L. Sudha (2011) Chaos image encryption using pixel shuffling. Computer Science & Information Technology 1(12):169–179
17. Parvin Z, Seyedarabi H, Shamsi M (2016) A new secure and sensitive image encryption scheme based on new substitution with chaotic function. Multimedia Tools and Applications 75(17):10631–10648
18. Qin C, Zhang X (2015) Reversible data hiding in encrypted image with privacy protection for image content. J Vis Commun Image Represent 31:154–164
19. Qin C, Chang C-C, Chiu Y-P (2014) A novel joint data-hiding and compression scheme based on SMVQ and image Inpainting. IEEE Trans Image Process 23(3):969–978
20. Rajput AS, Sharma M (2015) A novel image encryption and authentication scheme using chaotic maps. Advances in intelligent informatics. Springer International Publishing Switzerland 26:277–286
21. Sharath Kumar HS, Panduranga HT, Naveen Kumar SK (2013) A two stage combinational approach for image encryption. Advances in Computing & Information Technology.Conference paper (AISC,volume 177):843–849
22. Joseph H. Silverman (2007) The arithmetic of dynamical systems. Springer Science + Business Media, LLC.148–158
23. Sivakumar T, Venkatesan R (2016) A new image encryption method based on Knight's travel path and true random number. Journal of Information Science and Engineering, Jan 32(1):133–152
24. Wang X-Y, Feng C, Tian W (2009) A new compound mode of confusion and diffusion for block encryption of image based on chaos. Commun Nonlinear Sci Numer Simul 15(9):2479–2485
25. Wang H, Xu J-P, Sheng X-S, Zan P (2014) Discrete chaotic synchronization and It's application in image encryption. LSMS/ICSEE2014, part 2. CCIS 462:264–272
26. Xia Z, Wang X, Sun X, Wang Q (2016) A secure and dynamic multi-keyword ranked Search scheme over encrypted cloud data. IEEE Transactions On Parallel And Distributed Systems 27(2):340–352
27. Ye G, Wong K-W (2012) An efficient chaotic image encryption algorithm based on a generalized Arnold map. Nonlinear Dyn 69:2079–2087
28. Yuan H-M, Liu Y, Gong L-H, Wang J (2017) A new image cryptosystem based on 2D hyper-chaotic system. Multimedia Tools and Applications 76(6):8087–8108
29. Zhang S, Gao T (2016) An image encryption scheme based on DNA coding and permutation of hyper-image. Multimedia Tools and Applications 75(24):17157–17170
30. Zhang Q, Xue X, Wei X (2012) A novel image encryption algorithm based on DNA subsequence operation. Sci World J 2012:286741

Bo Li is the doctoral student of College of Electronic and Information Engineering, Southwest University. He received his Master's degree in China West Normal University, China, in 2008. His major study is in chaotic cryptography and application.

Xiaofeng Liao is a doctoral tutor, Senior Member IEEE. He received his PhD from the Electronic Science and Technology University, China, in 1997. His major study is artificial neural networks, wireless sensor networks, chaotic cryptography and application, and published more than 300 SCI papers. The H factor is 47 by 2015.-

Yan Jiang received her Master's degree in Southwest Forestry University, China, in 2011. Her major study is in grey system analysis and mathematics education.