International Journal of Computer Trends and Technology
ISSN: 2231-2803 / https://doi.org/10.14445/2231280453/IJCTT-V67I3P111
Volume 67 Issue 3, 53-64, March 2019

Original Article

A Novel Color Chaos-based Image Encryption Algorithm using Half-Pixel-Level Cross Swapping Permutation Strategy

Ruisong Ye[1], Li Liu[2]

Department of Mathematics, Shantou University, Shantou, Guangdong, 515063, China

Abstract *- A novel color chaos-based image encryption scheme with a permutation-diffusion mechanism is proposed. The permutation operation adopts a half-pixel-level interchange permutation strategy between different R, G, and B color channels to replace the traditional confusion operations. The pixel swapping between the higher 4-bit plane and the lower 4-bit plane of the R, G, and B channels improves the conventional permutation efficiency within the entire plain image and changes all the pixel values of R G, and B components. The multimodal skew map is applied to yield a pseudo-random gray value sequence in the diffusion operations to enhance security. Simulations have been carried out, and the results confirm the superior security of the proposed image encryption scheme.*

Keywords *- Cross was swapping permutation; Chaotic system; Generalized Cat map; Image encryption; Multimodal skew map.*

I. INTRODUCTION

With the development of information technology and the Internet, more and more multimedia information, such as color images, mp3, and video, are used in daily life. However, unauthorized access or modification to private digital information happens every day and has become a serious issue of digital information. However, recent research found that traditional cryptosystems like Data Encryption Standard (DES) and International Data Encryption Algorithm (IDEA) may not be suitable for image encryption due to some intrinsic features of images such as bulky data capacity, high redundancy, and strong correlation between pixels [1, 2].

In the last decades, chaos-based image encryption has attracted great attention from scholars thanks to chaotic systems' special characteristics, such as ergodicity, pseudo-randomness, and sensitivity to parameters and initial conditions [3-7]. Most of the existing chaos-based image encryption schemes employ a typical permutation-diffusion architecture. One encryption round includes several rounds of confusion and one round of diffusion processes. Fridrich initially proposed this architecture in 1998 [3]. In a confusion operation, two-dimensional chaos systems are usually employed to modify every pixel's location, while in the diffusion phase, the value of all the pixels is systematically changed. Generally, for an ideal cryptosystem, some basic requirements should be satisfied. For example, it should be sensitive to cipher keys; the keyspace should be large enough to resist brute-force attack; the permutation and diffusion processes should possess good statistical properties to frustrate statistical attack, differential attack, known-plaintext attack, chosen-plaintext attack, etc. However, the traditional permutation-diffusion architecture with fixed key streams is blamed for one big drawback. In their papers [8, 9], Li et al. pointed out that the permutation and diffusion processes become independent if one homogeneous plain image with an identical pixel gray value is encrypted. As a result, such a kind of image encryption scheme can be attacked by the following steps: (1) a homogeneous image with an identical pixel gray value is applied to remove the permutation effect; (2) the key streams of the diffusion process can be obtained by known-plaintext, chosen-plaintext or chosen-ciphertext attacks; (3) the remaining cipher-image can be then regarded as the output of a kind of permutation-only cipher, which has been shown insecure and can be cryptanalysis successfully. Image encryption schemes with conventional permutation-diffusion architecture have been analyzed or shown to suffer from security drawbacks [10-16].

To overcome the shortcomings such as small key space and weakly secure permutation-diffusion mechanism in chaos-based ciphers, many researchers investigated novel chaos-based cryptosystems with improved chaotic maps, large keyspaces, and good permutation-diffusion mechanisms, etc. In [17], Ye proposed a novel image encryption scheme with an efficient permutation-diffusion mechanism. In both the permutation and diffusion stages, generalized Arnold maps with real number control parameters are applied to generate pseudo-random sequences and enlarge the keyspace greatly. A two-way diffusion process is executed to improve the security of the

diffusion function. The cipher constructed in [17] shows good security and performance, including huge keyspace, efficient resistance against statistical attack, differential attack, known-plaintext, and chosen-plaintext attack. Zhou et al. adopted new chaotic systems by integrating the tent, logistic, and sine maps into one single system to produce the pseudo-random sequence [18, 19]. The intertwining logistic map and reversible cellular automata were utilized to design a novel image encryption scheme presented by Wang in [20]. This encryption scheme performs at bit level considering higher four bits of each pixel value. Some other novel image encryption schemes using bit-level permutation operation were proposed recently to improve the security issue of chaos-based image encryption schemes.

Each gray pixel value is usually decomposed into 8-bit planes for 256 gray-scale images for the bit-level permutation. The bit-level permutation not only rearranges the pixel positions but also modifies the gray pixel values [21, 22]. Therefore certain diffusion effect has been achieved in the permutation process. Zhang et al. proposed a novel image encryption scheme using lightweight bit-level confusion and cascade cross circular diffusion [23] using superior characteristics of bit-level operation and the intrinsic bit features of images. They also applied an expand-and-shrink strategy at the bit-level to shuffle the image with a reconstructed interpermuting plane [24]. All the proposed image encryption schemes perform better than the traditional permutation-diffusion structure operating at the pixel level. However, there exists one flaw in all bit-level-based image encryption schemes. Although the bit-level confusion operations can change the gray pixel values, they consume much execution time to get the eight-bit planes.

This paper proposes a novel color chaos-based image encryption algorithm using a half-pixel-level interchange permutation strategy. We first extract the higher 4-bit plane part RH/GH/BH and the lower 4-bit plane part RL/GL/BL from the R channel, G channel, and B channel. And then interchange the pixel values between RH and GL, the values between GH and BL, and BH and RL, respectively. A generalized Arnold map controls the interchange positions. The interchange strategy between the higher and lower plane parts will obtain two merits. It improves the conventional permutation efficiency within the plain image and changes all the pixel values of the entire image. To enlarge the keyspace and improve the diffusion effect, we adopt a multimodal skew tent map to generate a pseudo-random intensity sequence used to modify the pixels' color intensity values sequentially. Experiments are carried out thoroughly with detailed analyses, including key space analysis, key sensitivity analysis, statistical analysis, differential attack analysis, etc. All the results demonstrate that the proposed image encryption scheme possesses a large keyspace to efficiently frustrate brute-force attacks and other common various kinds of attacks.

The rest of the paper is organized as follows. Section II briefly introduces the logistic map and multimodal skew tent map with M tents. Section III introduces interpermuting planes. Section IV proposes a novel image encryption scheme composed of one permutation process and one diffusion process based on a generalized Arnold map and multimodal skew tent map. The corresponding decryption process is also presented in Section IV. The security and performance of the proposed image encryption algorithm are evaluated via detailed analyses and experiments in Section V. Section VI draws some conclusions.

II. CHAOTIC MAPS

A. Logistic map

In the proposed algorithm, a logistic map is used to generate the control parameters of the generalized Arnold map, which are used to perform the interchange permutation between higher-bit plane parts and lower-bit plane parts. The Logistic map can be presented in Eq. (1) [27, 28].

$$x_{n+1} = \lambda x_n (1 - x_n), \tag{1}$$

where $x_n \in (0,1)$ $\lambda \in (0,4]$ and. It is well known that when $\lambda \in (3.9,4]$ the logistic map undergoes chaotic and the pseudo-random sequence between 0 and 1 can be obtained.

B. The multimodal skew tent map

The unimodal skew tent map is defined by Eq. (2)

$$x_{n+1} = \begin{cases} x_n / a, & \text{if } x_n \in [0, a], \\ (1 - x_n)/(1 - a), & \text{if } x_n \in (a, 1], \end{cases} \tag{2}$$

where $x \in [0,1]$ is the system's state, and $a \in (0,1)$ what is the system parameter? There exist some good dynamical features in the skew tent map [29]. We generalize the unimodal skew tent map (2) to the multimodal skew tent map $T : [0,1] \to [0,1]$ defined by

$$x_{n+1} = \begin{cases} (x_n - a_{2i})/(a_{2i+1} - a_{2i}), & \text{if } x_n \in [a_{2i}, a_{2i+1}], \\ (a_{2i+2} - x_n)/(a_{2i+2} - a_{2i+1}), & \text{if } x_n \in (a_{2i+1}, a_{2i+2}], \end{cases} \tag{3}$$

where $i = 0, \cdots, M-1$, $0 = a_0 < a_1 < \cdots < a_{2M-1} < a_{2M} = 1$ and M is referred to the number of tents. See Fig. 1(a) for the case of $M = 3$, a = [0, 0.16, 0.3, 0.51, 0.68, 0.78, 1.0] .

The orbit of $x_0 = 0.367$ generated by system (3) is $\{x_k = T^k(x_0), k = 0,1,\cdots\}$, shown in Fig. 3(a) for a = [0, 0.16, 0.3, 0.51, 0.68, 0.78,1.0], $M = 3$. Its waveform is quite irregular, implying the system's chaotic feature. To illustrate the distribution of the orbit points $\{x_k : k = 0,1,\cdots,20000\}$, we plot the histogram in Fig. 3(b). It can be seen that the points of the orbit spread more or less evenly over the unit

interval. Multimodal skew tent map possesses desirable auto-correlation and cross-correlation features as well. The trajectory is applied to calculate the correlation coefficients, shown in Figs. 3(c)-(d) respectively. The orbits of and calculate the cross-correlation coefficients $x_0 = 0.367$ $y_0 = 0.368$. The control parameter $a_1, \cdots, a_{2M-1}$ and the initial condition x_0 can be regarded as cipher keys if the multimodal skew tent map is applied to design image encryption schemes.

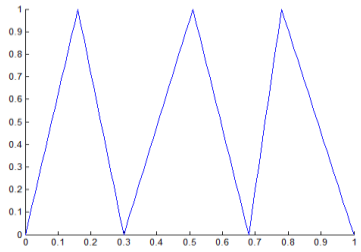
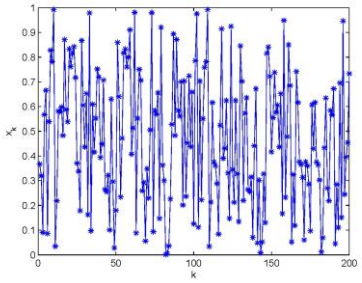
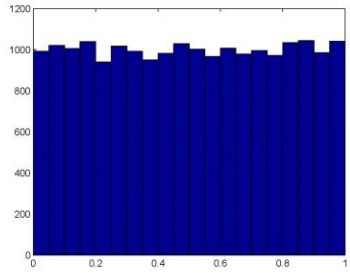


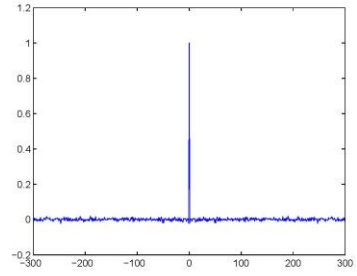
Fig. 1 The diagram of a multimodal skew tent map.



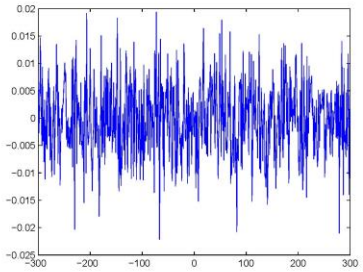
(a)



(b)



(c)



(d)

Fig. 2. Orbits derived from the considered multimodal skew tent map with $a = [0\ 0.16\ 0.3\ 0.51\ 0.68\ 0.78\ 1.0]$. (a) The chaotic orbit of $x_0 = 0.367$; (b) Histogram of a typical orbit of length 20000; **(**c) The auto-correlation; (d)The cross-correlation

The probability density $\rho(x)$ for multimodal skew tent map on [0, 1] is given by [30]:

$$\rho_0(x) = \begin{cases} 1, & \text{if } x \in (0,1), \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

This fact has been illustrated numerically in Fig. 3(b). The existence and unique value of the Lyapunov exponent also follow from the following theorem. It has been shown that for the multimodal skew tent map (3) with the constant probability density $\rho(x) \equiv 1$, the Lyapunov exponent of (3) is (see [30] for more details)

$\lambda = -p_1 \ln p_1 - p_2 \ln p_2 - \cdots - p_{2M-1} \ln p_{2M-1} - p_{2M} \ln p_{2M}$.

λ It is always larger than zero, implying the dynamic system is always chaotic. For $M = 3, a = [0\ 0.16\ 0.3\ 0.51\ 0.68\ 0.78\ 1.0]$, we obtain

$p_1 = 0.16, p_2 = 0.14, p_3 = 0.21, p_4 = 0.17, p_5 = 0.1, p_6 = 0.22.$

So $\lambda = 1.7608$. It is usually larger than the Lyapunov exponent for the unimodal skew tent map (2). As a matter of fact, for the unimodal skew tent map (1), the largest Lyapunov exponent $\ln 2 = 0.6931$ occurs in the extreme case $a = 0.5$. It implies that the multimodal skew tent map (3) is in a stronger sense chaotic and, therefore, can perform better data mixing, which makes it a better choice for designing encryption schemes than the unimodal skew tent map. The unimodal skew tent map is widely applied to generate pseudo-random sequences in chaos-based image encryption schemes. Multimodal skew tent map has been shown to possess good chaotic natures, such as pseudo-randomness, ergodicity, and desirable auto-correlation and cross-correlation features [30]. We apply a multimodal skew tent map to enlarge the cipher keyspace as it has more choices of control parameters.

III. THE INTERPERMUTING PLANES

In traditional chaos-based image encryption algorithms, two steps are performed alternatively, as shown in Fig. 3. In the confusion phase, two-dimensional chaotic maps are usually employed as pseudo-random number generators, and a mapping rule from sequential positions to pseudo-random positions is defined. The pixels are mapped from a plain image to a pre-defined new blank plane, which becomes the confused image after the confusion operation. The two planes are the plain image and the blank plane, and the mapping operation is performed in a one-way manner, as shown in Fig.4.

In Fig. 4, A and B represent two adjacent pixels in the plain image, while A' B' and are the two corresponding random pixels in the blank plane. In a traditional confusion phase, one pixel (e.g., pixel A in Fig. 4(a)) is mapped from the original plain image to a random position in the pre-defined blank plane (e.g., the pixel A'in Fig. 4(b)). Once all the pixels in the plain image have been mapped to the blank plane, the newly obtained plane becomes the confused image, and the confusion operation finishes. In contrast, in confusion operations based on interpermuting planes, the two planes are both parts of the plain image, and confusion operations are performed bi-directional. The two random positions in the two interpermuting planes are exchanged rather than simply moving one pixel from the original plain image to a pseudo-random position in the blank plane, as shown in Fig. 5.

The two interpermuting planes can be obtained by combining different bit planes for a gray image or different color channels for an RGB image. A two-dimensional chaotic map is used to define the mapping of a pixel from its regular position (x, y) to a new pseudo-random position (x', y'). In other words, when confusion is applied by using interpermuting planes, the two interpermuting planes are both parts of the plain image, and the confusion involving these two planes is performed by exchanging the pixel located at position (x, y) in-plane 1 and the pixel located at position (x', y') in-plane 2 as depicted diagrammatically in Fig.5. Referring to Fig.5, if plane1 and plane 2 are any two of the color channels of an RGB image, the pixels A, B, A'and B'are all 8-bit pixels. If plane 1 and plane 2 are obtained by combining different bit planes of a gray image or one-color channel for an RGB image, interpermuting plane 1 would contain the higher 4-bit planes. Interpermuting plane 2 would contain the lower 4-bit planes of the gray image or some certain color channel component, in which case A, B, A' and B' should be special units that contain 4-bit information [25].

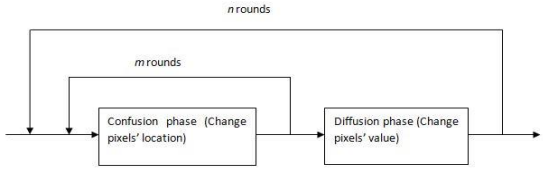


Figure 3. The traditional architecture of chaos-based image encryption.

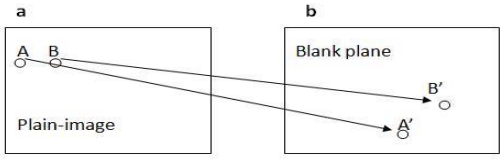


Figure 4. Traditional confusion operation.

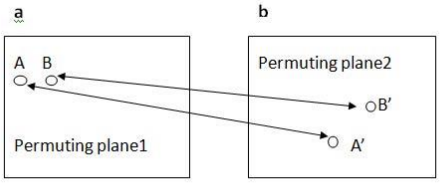


Fig. 5 Interpermuting-plane confusion operation.

The swapping operation between the two interpermuting planes is then performed as follows. In Fig. 5, taking pixel A, for example, A's coordinate (x, y) is. The corresponding random position is then calculated using a two-dimensional chaotic map and determined to be the position (x', y'). First, pixel A is moved from position (x, y) in-plane 1 to (x', y') in-plane 2, similar to traditional confusion operations. Meanwhile, the pixel located (x', y') in the in-plane 2 is moved back to the position (x, y) 1 in the plane. When one random position is calculated, two pixels (or two units that contain bit information of different bit planes) are permuted between the two planes. The pixel in plane 1 is confused using an ordinary two-dimensional chaotic map and moved from its regular position to a random one.

Meanwhile, the pixel in plane 2 is moved from the random position in-plane 2 back to the ordinary position in-plane1, which is the effect of confusion using a reverse two-dimensional chaotic map. The reverse chaotic map is applied first in [4]. Therefore, the confusion approach proposed in this paper can be considered a combination of two known confusion techniques, i.e., those based on ordinary and reverse two-dimensional chaotic maps, but with the additional use of two interpermuting planes for swapping purposes.

There are two calculation steps in the confusion phase. The first step calculates the new position of the pixel (the most time-consuming step), and the second step moves the pixel from one memory address to the other [23]. In the proposed confusion operation based

on interpermuting planes, when one random position is calculated, ordinary and reverse random mapping are used as a swapping strategy to simultaneously permute the two pixels between the two interpermuting planes, which can save around half of the confusion time.

The success of the newly proposed confusion scheme would, therefore, to a large extent, depend on finding a way to generate two interpermuting planes from the existing image. For an image with 256 gray levels, the two interpermuting planes could be obtained by dividing the 8-bit planes into two parts. For example, the higher 4-bit plane part and the lower 4-bit plane part could either be treated as interpermuting plane 1 or plane 2, respectively, or the four odd-numbered bit planes and the four even-numbered bit planes could be treated as the two interpermuting planes, respectively. For an RGB color image, there are six naturally interpermuting planes, three higher 4-bit planes, and three lower 4-bit planes of the R, G, and B channels, respectively. Alternatively, the different interpermuting planes could also be obtained by using a distinct combination of different bit planes or color channels.

IV. THE PROCESS OF THE IMAGE ENCRYPTION SCHEME

In this section, the proposed encryption algorithm is proposed. The flowchart of the encryption process is shown in Fig. 6. The encryption algorithm consists of two processes: permutation and diffusion. We read an 8-bit color plane image P with size $H \times W$. In this paper, we restrict the plain images with equal height H and width W ; that is, $H = W$ The plain color image is modeled by a three-dimensional matrix sized $H \times W \times 3$ whose elements belong to the integers between 0 and 255, denoting the intensity values of different color channel components.

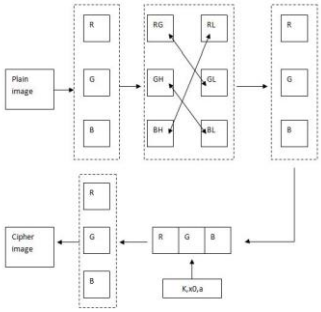


Fig. 6 Flowchart of the encryption algorithm.

A. Permutation process

In the permutation process, we first extract the higher 4-bit planes and the lower 4-bit planes from the R, G, and B channels and denote them as interpermuting planes RH, RL, GH, GL, BH, and BL, respectively. And then, we will exchange the pixels between two interpermuting planes using a generalized Arnold map, defined by Eq. (5).

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & p \\ q & 1+pq \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \bmod H, \qquad (5)$$

where $p, q \in 0,1,2,...,H-1$, p and q are the control parameters of generalized Arnold map and function "x mod H represents the remainder after x divided by H . (x, y) is the original position of the higher plane while (x', y') is the pseudo-random position governed by the generalized Arnold map. Exchanging the pixel value at the location (x, y) of the higher 4-bit plane part extracted from one color channel with the pixel value at the location (x', y') of the lower 4-bit plane part from another color channel. The detailed permutation process is depicted as follows.

Step 1. Generate six control parameters $p(i)$ $q(i)$ ($i = 1,2,3$) by Eq. (1). λ x_0 Are the parameters as initial secret keys.

Step 2. Set the higher 4-bit plane part of the R channel to be interpermuting plane 1 and the lower 4-bit plane of the G channel to be interpermuting plane 2.

Step 3. For each pixel in plane 1, a random position (x', y') is calculated by Eq. (5) with parameters $p(1), q(1)$. Then exchange the two pixels by locating (x, y) them in interpermuting plane 1 and (x', y') interpermuting plane 2 until all the pixels have been confused. We finish the interchange permutation between RH and GL.

Step 4. Define the higher 4-bit plane of the G channel as interpermuting plane 1 and the lower 4-bit plane of the B channel as interpermuting plane 2, and repeat Step 3 with parameters $p(1), q(1)$ replaced by parameters $p(2), q(2)$. We then finish the interchange permutation between GH and BL.

Step 5. Define the higher 4-bit plane of B channel as interpermuting plane 1 and the lower 4-bit plane of R channel as interpermuting plane 2, and repeat Step 3 with parameters $p(1), q(1)$ replaced by parameters $p(3), q(3)$. We then finish the interchange permutation between BH and RL.

Step 6. Integrate the exchanged matrices pairs to be three permutated color components R, G, B :

$$R(i, j) = RL(i, j) + RH(i, j) \times 16,$$
$$G(i, j) = GL(i, j) + GH(i, j) \times 16,$$
$$B(i, j) = BL(i, j) + BH(i, j) \times 16, i, j = 0,1,\cdots, H-1.$$

B. Diffusion process

In the diffusion phase, the values of all the pixels are systematically modified. First, the confused R, G, and B channels are transformed into a vector. The R, G, and B channels are transformed into three vectors, respectively, and each of them has $H \times W$ numbers. After that, the confusing red, green, and blue color component matrices R, G, B are integrated into a

vector V with length $3HW$. The diffusion process is described as follows.

Step 1. Set the values of M the control parameters $a_i (i = 1, ..., 2M-1)$ and the initial condition y_0. Iterate the multimodal skew tent map (4) 100 times and reject the transient 100 points to avoid the harmful effect. We reset y_0 to be y_{100}. Let $n = 1$ $s = 1$,

Step 2. Iterate (4) for s rounds with initial y_0 to get y. The keystream element $k(n)$ is calculated by

$$k(n) = \mathrm{mod}(\mathit{floor}(y \times 10^{14}), 256). \quad (6)$$

The intensity values are modified sequentially according to Eq. (7) $V(n)$ $c(n)$ $c(n-1)$. The intensity values of the currently operated pixel in the confused vector V, output cipher-pixel, and previous cipher-pixel, respectively. We note that an initial value seed $c(0)$ is required for a well-defined calculation.

$$c(n) = V(n) \oplus \mathrm{mod}(k(n) + c(n-1), 256). \quad (7)$$

Step 3. Compute s $s = 1 + \mathit{mod}(c(n), 2)$ by. Let $n = n+1$ and reset y_0 to be y. Return to Step 2 until n it reaches $3HW+1$.

Step 4. Convert the resulted vector c to one color cipher image with height H and width W.

The complete diffusion process is composed of Steps 1 to Step 4. After the diffusion process, the yielded vector c is transformed into three two-dimensional arrays modeling the cipher-image's red, green, and blue color components.

C. The process of image decryption scheme

The decryption procedure is the reverse process of the encryption algorithm, and the flowchart of the decryption process is shown in Fig. 7. The entire decryption procedure is outlined as follows.

Step 1. Extract the R, G, and B channels of the cipher-image, transform them into three length HW vectors, then integrate them into one vector c with length $3HW$. Set the values of M the control parameters $a_i (i = 1, ..., 2M-1)$ and the initial condition y_0. Iterate the multimodal skew tent map (4) 100 times and reject the transient 100 points to avoid the harmful effect. We reset y_0 to be y_{100}. Let $n = 1$ $s = 1$, Us do the loop consisting of Step 2-Step 3, in which an initial value $c(0)$ is required and can be regarded as one part of the cipher keys.

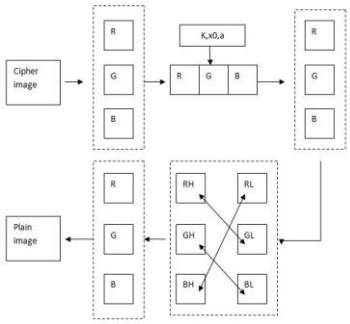


Fig. 7 Flowchart of the decryption algorithm.

Step 2. Iterate (4) for s rounds with initial y_0 to get y. The current keystream element $k(n)$ is calculated by Eq. (8).

$$k(n) = \mathrm{mod}(\mathit{floor}(y \times 10^{14}), 256). \quad (8)$$

The confused vector V is obtained according to Eq. (9), where $V(n)$ $k(n)$ $c(n)$ $c(n-1)$, are the currently operated pixel in the confused vector, temporal keystream element, output cipher-pixel, and previous cipher-pixel, respectively.

$$V(n) = c(n) \oplus \mathrm{mod}(k(n) + c(n-1), 256). \quad (9)$$

Step 3. Compute s $s = 1 + \mathit{mod}(c(n), 2)$ by. Let $n = n+1$ and reset y_0 to be y. Return to Step 2 until n it reaches $3HW+1$.

Step 4. Extract the R, G, and B channels' pixel values from the confused vector V orderly and convert them into two-dimensional matrices denoted as R, G, and B, respectively.

Step 5. Generate six chaotic values $p(i)$ $q(i), i = 1, 2, 3$ by Eq. (1) with the cipher keys λ x_0,

Step 6. Set the higher 4-bit plane of R to be interpermuting plane 1 and the lower 4-bit plane of G to be interpermuting plane 2. For each pixel in plane 1, a random position (x', y') is calculated by Eq. (5) with $p(1), q(1)$. Then exchange the two pixels at locating (x, y) in interpermuting plane 1 and (x', y') interpermuting plane 2 until all the pixels have been exchanged.

Step 7. Define the higher 4-bit plane of G as interpermuting plane 1 and the lower 4-bit plane of B as interpermuting plane 2; repeat the same operation in Step 6 with control parameters $p(1), q(1)$ $p(2), q(2)$.

Step 8. Define the higher 4-bit plane of B as interpermuting plane 1 and the lower 4-bit plane of R as interpermuting plane 2; repeat the same operation in Step 6 with control parameters $p(1), q(1)$ $p(3), q(3)$.

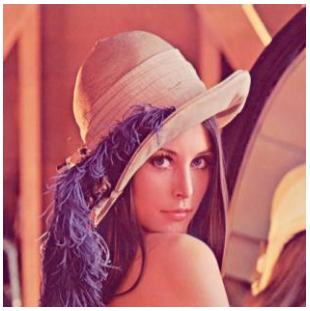
Step 9. Form the yielded R, G, and B channels' two-dimensional matrices to be three-dimensional matrices modeling the plain color image.

V. PERFORMANCE ANALYSIS

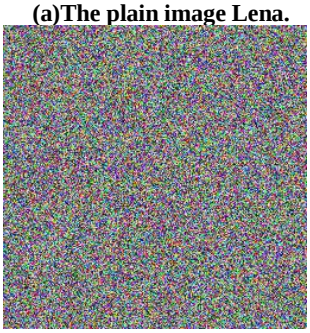
According to the basic principle of cryptology [31], a good encryption scheme requires sensitivity to cipher keys, i.e., the ciphertext should closely correlate with the keys. An ideal encryption scheme should have a large keyspace to make brute-force attacks infeasible; it should also resist various attacks like statistical attacks, differential attacks, etc. In this section, some security analyses have been performed on the proposed image encryption scheme, including the most important ones like key space analysis, statistical analysis, and differential analysis. All the analyses show that the proposed image encryption scheme is highly secure.

A. Experimental results

We use MATLAB R2010b to run the encryption and decryption process in a computer with 1.70 GHz CPU, 4 GB memory, and Microsoft Windows 8 operating system. All the dates in this article are obtained under this circumstance. The plain image we choose in the simulation is the color image Lena.bmp of size 256×256 . The cipher keys are λ = 3.999 , $c(0)$ = 78 , x_0 = 0.36 , y_0 = 0.375 , a =[0 , 0.16 , 0.3 , 0.51, 0.68 , 0.78 , 1.0]. Fig.8 shows the encryption result.



(a)The plain image Lena.



(b)The cipher-image of Lena.

Fig. 8 Encryption result.

B. Statistical analysis

It is well known that the statistical property is enormously vital, and an ideal image algorithm should be robust against any statistical attacks. Histogram and correlation of two adjacent pixels are two important indicators of statistical analysis.

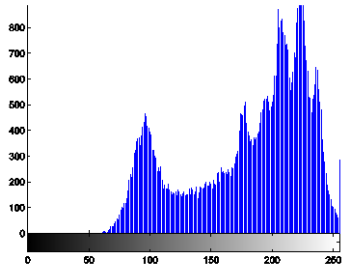
Histogram. Histograms of plain-image and cipher-image are plotted, through which we can intuitively see the number of pixels of each value. A good image algorithm should make the histogram of the cipher-image as flat as possible. The histograms of Lena and its cipher-image are shown in Fig.9. Fig.9(a)-(c) are the histograms of R,G,B components of plain-image Lena; Fig.9(d)-(f) are the histograms of R,G,B components of cipher-image.

Correlation of adjacent pixels. Generally speaking, the two adjacent pixels of a plain image would come near each other. Adjacent pixels from the plain image of Lena and its cipher image are selected in the horizontal direction, vertical direction, and diagonal direction, respectively, and the correlation coefficients r_{xy} of each pair are calculated using the following equations [32] :

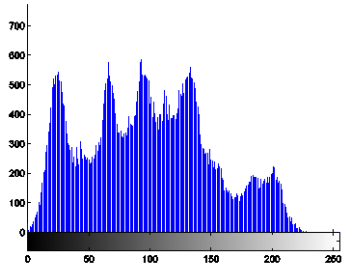
$$cov(x,y) = E(x-E(x))(y-E(y)),$$

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)D(y)}} \qquad (10)$$

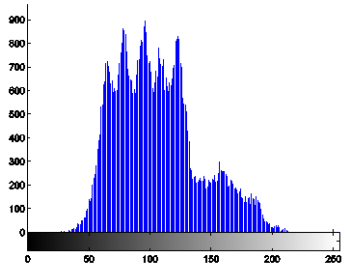
where x and y are values of the two adjacent pixels in the image $E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i$ $D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2$.



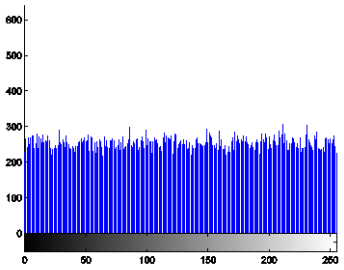
(a)



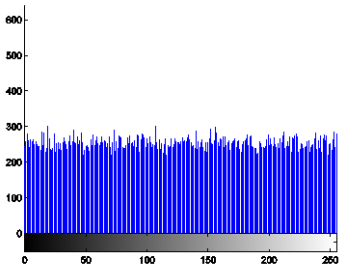
(b)



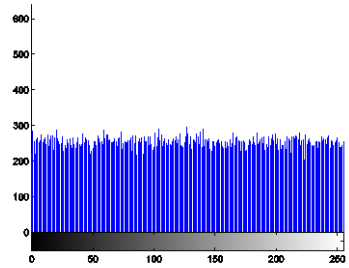
(c)



(d)



(e)



(f)

Figure 9. (a)-(c): histograms for R, G, B components of Lena; (d)-(f): histograms for R, G, B components of cipher-image.

The correlations between adjacent pixels of plain images and cipher images are given in Table 1. From Table 1, we can find no detectable correlations between the plain image and its corresponding cipher image. The correlation coefficients of adjacent pixels are significantly deduced in our proposed scheme, and the proposed encryption scheme shows perfect correlation performance.

Table 1. Correlation coefficients of adjacent pixels in the plain and cipher images.

Correlation between adjacent pixels	Plain image			Cipher image		
	Red	Green	Blue	Red	Green	Blue
Horizontal	0.9460	0.9465	0.9046	0.0037	0.0006	0.0001
Vertical	0.9720	0.9729	0.9465	-0.0041	0.0002	0.0012
Diagonal	0.9212	0.9236	0.8677	0.0004	0.0006	0.0008

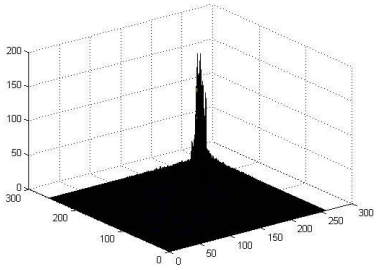
Furthermore, we introduce a new statistic index to reflect the effect of the cipher-image, which is the co-occurrence histogram [33], and we depict it as follows. The co-occurrence histogram in the horizontal direction is defined by Eq.(11).

$$co_1(i,j)=\sum_{x=1}^{n-1}\sum_{y=1}^{n}\delta(g(x,y)-i)\delta(g(x+1,y)-j), i,j=0,1,...,255. \quad (11)$$

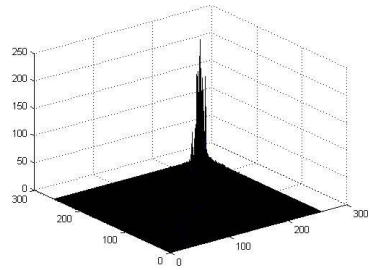
The co-occurrence histogram in the vertical direction is defined by Eq.(12).

$$co_2(i,j)=\sum_{x=1}^{n}\sum_{y=1}^{n-1}\delta(g(x,y)-i)\delta(g(x,y+1)-j), i,j=0,1,...,255. \quad (12)$$

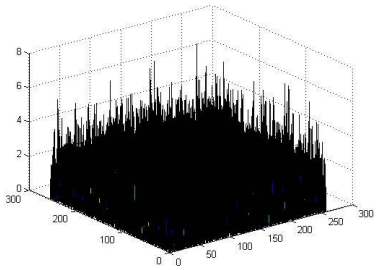
where $g(x,y)$ is the pixel value at the location (x,y) ? If $x = y$,then $\delta(x,y)=1$, otherwise, $\delta(x,y)=0$. The detailed co-occurrence histograms of the plain-image and cipher-image are shown in Fig.10.



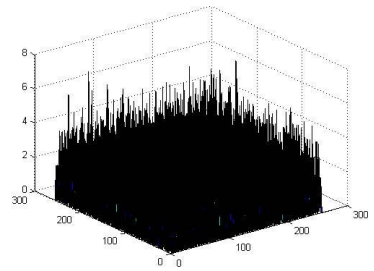
(a)



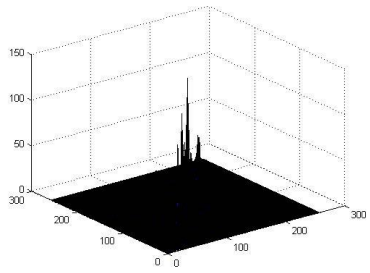
(b)



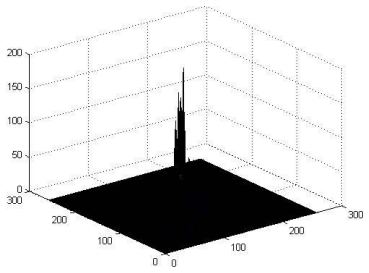
(c)



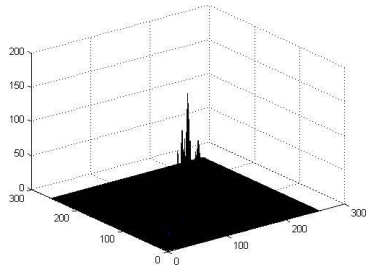
(d)



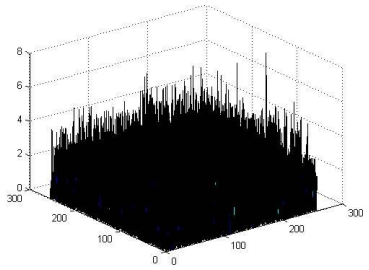
(e)



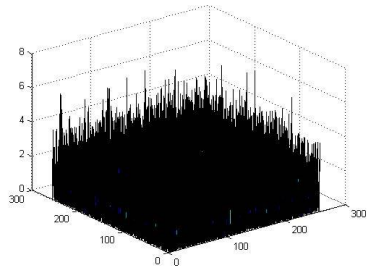
(j)



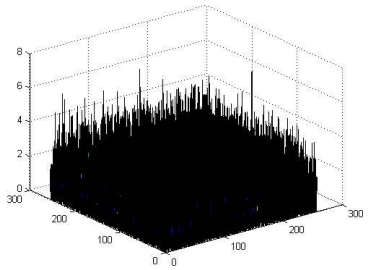
(f)



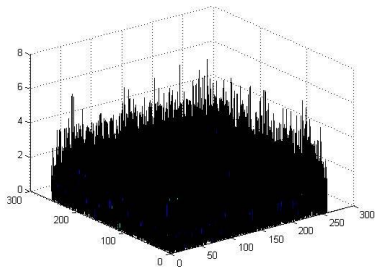
(k)



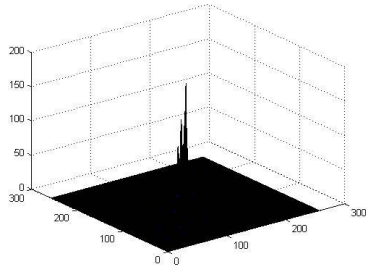
(g)



(l)



(h)



(i)

Figure 10. Co-occurrence histograms of plain image and cipher image. (a), (b), (e), (f), (i), (j) are the vertical and horizontal co-occurrence histograms of R, G, and B components of Lena, respectively; (c), (d), (g), (h), (k), (l) are the vertical and horizontal co-occurrence histograms of R, G, B components of cipher-image respectively.

Besides, the information entropy Eq. (13) correlated with co-occurrence is introduced to measure the uniformity level of the two-dimension histogram. The distribution of the pixel space will be more uniform when the value of information entropy is bigger. The simulation results are shown in Table 2.

$$H(co_l) = -\sum_{i=0}^{255}\sum_{j=0}^{255} \frac{co_l(i,j)}{n(n-1)} \ln\left(\frac{co_l(i,j)}{n(n-1)}\right), l = 1, 2. \quad (13)$$

Table 2. Co-occurrence histogram entropy.

Information entropy	Plain image			Cipher image		
	Red	Green	Blue	Red	Green	Blue
Horizontal	8.416	8.722	8.271	10.515	10.511	10.514
Vertical	8.088	8.340	7.954	10.511	10.515	10.515

Information entropy analysis. In [34], entropy was proposed by Shannon to measure the randomness and unpredictability of an information source quantitatively. The mathematical formula for the

entropy of a message source is defined in Eq. (14), where s the source N is the number of bits to represent the symbol s_i $P(s_i)$ and is the probability of the symbol s_i .

$$H(s) = -\sum_{i=0}^{2^N-1} P(s_i)\log_2 P(s_i). \quad (14)$$

For a purely random source emitting 2^N symbols, the entropy is N . Therefore, the upper-bound entropy of an effective cipher-image with 256 Gray levels is 8. Such expected value will be achieved when the cipher-image is uniformly distributed, i.e., the image has a complete flat histogram.

The results of the information entropy analysis for R, G, and B channels of the plain-image and cipher-image are listed in Table 3. The results illustrate that the entropies of the cipher image are very close to the upper-bound value 8. Therefore, we can conclude that there is little possibility of eavesdropping and our encryption scheme has high robustness against entropy attacks.

Table 3. Entropy values of a plain image and its cipher image.

	Plain image			Cipher image		
	Red	Green	Blue	Red	Green	Blue
Entropy	7.2763	7.5834	7.0160	7.9971	7.9971	7.9976

C. Differential attack analysis.

The number Pixel Change Rate(NPCR) and Unified Average Changing Intensity(UACI) are usually used to measure the sensitivity of the cryptosystem to a slight modification of the plain image. In an ideal situation, a slight modification of the plain image will lead to a completely different cipher image, indicating its resistance to differential attack. Otherwise, it would have been possible to obtain the correlation between the plain image and the cipher image by a series of attacks of this nature. NPCR and UACI are defined by Eq. (15).

$$NPCR = \frac{\sum_{i,j} D(i,j)}{W \times H} \times 100\%.$$

$$UACI = \frac{1}{W \times H}\left[\sum_{i,j} \frac{|c_1(i,j) - c_2(i,j)|}{255}\right] \times 100. \quad (15)$$

where c_1 c_2 and are two images of the same size $W \times H$. If $c_1(i,j) \neq c_2(i,j)$, then $D(i,j) = 1$ otherwise, $D(i,j) = 0$

We randomly select 3 positions from R, G, and B channels with the pixel gray added 1, respectively, to test the influence of one-pixel change on the whole cipher image. We get the values of NPCR and UACI each time, then take the average of them. Table 4 gives NPCR and UACI performances of the proposed image encryption scheme. It shows clearly that the proposed scheme reaches very good NPCR and UACI performance when encrypted in just one round. The results show that the proposed image encryption method is sensitive to plaintext, which is important to resist differential attacks.

Table 4. Differential attack analysis.

	Red	Green	Blue	average
NPCR	99.5987	99.6185	99.5972	99.6048
UACI	33.5033	33.4798	33.4164	33.4665

D. Keyspace analysis

The keyspace is the total number of different keys used in a cryptosystem. In [35], it is suggested that the keyspace of a chaos-based image cryptosystem should be better larger than 2^{100} . As to the proposed scheme, there are two keys in the permutation phase, say λ x_0 and. The multimodal skew tent map's initial value y_0 $c(0)$ and control parameter $a_i (i = 1,...,2M-1)$ serve as the proposed cryptosystem's primary key in the diffusion phase. The keyspace wholly depends on the encryption processes, denoted as *Key - P* . According to the IEEE floating-point standard [36], the computational precision of the 64 -bit double-precision number is about 10^{-15} . In our proposed algorithm, the range λ is within 10^{15} x_0 . Because can be anyone among those 10^{16} possible values within $(0,1)$, and so as $x00$ and a_i ($i = 1,...,2M-1$) . As to H gray-level image, the valid values of $c(-1)$ H is. For the case $M = 3$, we can take an example, $\lambda = 3.999$ $y_0 = 0.375$ x_0 =0.36, a =[0 0.16 0.3 0.51 0.68 0.78 , 1.0], $c(0)$ =78, then the keyspace of the proposed cryptosystem is

$$Key - P = 10^{15} \times 10^{16} \times 10^{16} \times 10^{16} \times 10^{16} \times 10^{16} \times 10^{16} \times 10^{16} \times 256 \approx 2^{429} ,$$

Which satisfies the security requirement suggested in [35] and is large enough to resist brute-force attacks.

E. Key sensitivity analysis

Key sensitivity of an image cryptosystem can be observed in two aspects: (i) completely different cipher images should be produced when slightly different keys are applied to encrypt the same plain image; (ii) the cipher image cannot be correctly decrypted even tiny mismatch exists in decryption keys [37]. Concerning the symmetrical characteristic of the secret key, we typically test the sensitivity λ x_0 $c(0)$ y_0 $a(2)$ $a(5)$ to avoid redundancy. The plain image is respectively encrypted with one master cipher and six cipher keys which have only a slight change in any one of six parts of the master cipher key. The following cipher keys are used to perform the simulation.

Master cipher key:

Mkey(λ , x_0 $c(0)$ y_0 $a(2)$ $a(5)$);

Six slightly different keys:

Key1($\lambda + 10^{-15}$, x_0 , $c(0)$, y_0 , $a(2)$, $a(5)$);

Key2(λ , $x_0 + 10^{-16}$, $c(0)$, y_0 , $a(2)$, $a(5)$);

Key3(λ , x_0 , $c(0)+1$, y_0 , $a(2)$, $a(5)$);

Key4(λ , x_0 , $c(0)$, y_0+10^{-16} , $a(2)$, $a(5)$);

Key5(λ , x_0 , $c(0)$, y_0 , $a(2)+10^{-16}$, $a(5)$);

Key6(λ , x_0 , $c(0)$, y_0 , $a(2)$, $a(5)+10^{-16}$).

(i) For the first kind of key sensitivity analysis, the plain-image Lena is encrypted using Mkey and six slightly different keys. Then we have computed the 2D differences between the various color layers of the cipher image yielded using Mkey and six other cipher images produced using slightly different keys. The results are given in Table 5. The values indicate that all the cipher images are highly different and hence the cipher images produced by the proposed image cipher algorithm are extreme sensitivity to cipher keys.

(ii) For the second key sensitivity analysis, plain-image Lena is encrypted using Mkey, and the encrypted image is decrypted with six slightly different keys. Compared with the right decryption images, there are significant differences. The results are given in Table 6. It is clear that all the values are close 99.7000 , and the images decrypted using slightly different keys are highly different.

Table 5. Key sensitivity analysis I.

	Differences between the cipher images obtained using Mkey and					
	Skey1	Skey2	Skey3	Skey4	Skey5	Skey6
Dr	99.6292	99.3088	99.6048	99.2996	99.6475	99.6033
Drg	99.6048	99.5804	99.5834	99.6368	99.5911	99.5834
Drb	99.5834	99.6048	99.5789	99.5300	99.6078	99.5972
D	99.6292	99.6277	99.6094	99.5956	99.6109	99.5773
Dgg	99.5743	99.6216	99.6201	99.6277	99.6231	99.6094
Dgb	99.5773	99.6323	99.5972	99.5926	99.5834	99.6048
Dbr	99.5667	99.6078	99.6124	99.6582	99.6704	99.6216
Blog	99.6384	99.6475	99.6307	99.5819	99.6307	99.5773
Dbb	99.6002	99.6140	99.6213	99.6292	99.6063	99.6033

Table 6. Key sensitivity analysis II.

	Differences between decrypted images obtained using Mkey and					
	Skey1	Skey2	Skey3	Skey4	Skey5	Skey6
Dr	99.5987	99.6445	99.6399	99.5911	99.5987	99.5667
Dgg	99.6201	99.4614	99.6155	99.4232	99.6414	99.6414
Dbb	99.5682	99.1379	99.6078	99.1562	99.6140	99.6170

VI. CONCLUSION

An efficient image encryption scheme based on a half-pixel-level interchange of the pixels between the higher plane and the lower plane among R, G, and B channels is proposed in the paper. The proposed scheme can shuffle the plain image efficiently in the permutation process. An effective diffusion process is also presented to change the gray values of the whole image pixels. Security analyses, including co-occurrence histogram, keyspace analysis, key sensitivity analysis, statistical attack analysis, and differential attack analysis, are performed numerically and visually. The experimental results show that the proposed encryption scheme is secure thanks to its large keyspace and high sensitivity to the cipher keys and plain images. These satisfactory properties make the proposed scheme a potential candidate for encrypting multimedia data such as images, audio, and even videos.

ACKNOWLEDGMENT

This research is supported by China's National Natural Science Foundation (No. 11771265).

REFERENCES

[1] S. Li, G. Chen, X. Zheng, Chaos-based image encryption for digital and videos, in B. Furht, D. Kirovski(Eds), Multimedia Security Handbook. CRC Press, Florida, the United States of America, 2005, pp. 133-167(Chapter4).

[2] Y. Wang, K. W. Wong, X. F. Liao, T. Xiang, G. R. Chen, A chaos-based image encryption algorithm with variable control parameters, Chaos Solitons Fractals, 41:4(2009), 1773-1783.

[3] J. Fridrich, Symmetric ciphers based on two-dimensional chaotic maps, International Journal of Bifurcation and Chaos, 8:6(1998), 1259-1284.

[4] W. Zhang, Kwok-wo. Wong, H. Yu, Z. Zhu, An image encryption scheme using the reverse 2-dimensional chaotic map and dependent diffusion, Commun. Nonlinear Sci. Numer. Simul., 18:8(2013), 2066-2080.

[5] G. Chen, Y. Mao, Charles K. Chui, A symmetric image encryption scheme based on 3D chaotic cat maps, Chaos, Solitons and Fractals, 21:3(2004), 749-761.

[6] Y. Q. Zhang, X. Y. Wang, A new image encryption algorithm based on non-adjacent coupled map lattices, Applied Soft Computing, 26(2015), 10-20.
[7] R. Ye, M. Ge, P. Huang, H. Li, A Novel Self-adaptive Color Image Encryption Scheme, International Journal of Computer Trends and Technology, 40:1(2016), 39-44.
[8] C. Q. Li, S. J. Li, G. R. Chen, G. Chen, L. Hu, Cryptanalysis of a new signal security system for multimedia data transmission. EURASIP J. Appl. Signal Process., 8(2005), 1277-1288.
[9] S. J. Li, C. Q. Li, G. R. Chen, N. G. Bourbakis, K. T. Lo, A general quantitative cryptanalysis of permutation-only multimedia ciphers against plain-image attacks. Signal Process. Image Commun., 23(2009), 212-223.
[10] D. Xiao, X. Liao, P. Wei, Analysis and improvement of a chaos-based image encryption algorithm, Chaos, Solitons and Fractals, 40(2009), 2191–2199.
[11] E. Solak, C. Cokal, O. T. Yildiz, and T. Biyikoglu, Cryptanalysis of Friedrich's chaotic image encryption, International Journal of Bifurcation and Chaos, 20:5(2010), 1405-1413.
[12] E. Solak, R. Rhouma, and S. Belghith, Cryptanalysis of a multi-chaotic systems based image cryptosystem, Optics Communications, 283:2(2010), 232-236.
[13] J. M. Liu, Q. Qu, Cryptanalysis of a substitution-diffusion based on cipher using chaotic standard and logistic map, in Third International Symposium on Information Processing, 2010, pp. 67-69.
[14] R. Rhouma and S. Belghith, Cryptanalysis of a new image encryption algorithm based on hyper-chaos, Physics Letters A, 372:38(2008),5973-5978.
[15] R. Rhouma, E. Solak, S. Belghith, Cryptanalysis of a new substitution-diffusion based image cipher, Commun. Nonlinear Sci. Numer. Simulat., 15(2010), 1887-1892.
[16] X. Wang, G. He, Cryptanalysis on a novel image encryption method based on total shuffling scheme, Optics Commun., 284 (2011), 5804-5807.
[17] R. Ye, A novel chaos-based image encryption scheme with an efficient permutation-diffusion mechanism, Optics Communications, 284(2011), 5290-5298.
[18] Y. Zhou, L. Bao, C. L. Philip Chen, Image encryption using a new parametric switching chaotic system, Signal Processing, 93(2013), 3039-3052.
[19] Y. Zhou, L. Bao, C. L. Philip Chen, A new 1D chaotic system for image encryption, Signal Processing, 97(2014), 172-182.
[20] X. Wang, D. Luan, A novel image encryption algorithm using chaos and reversible cellular automata, Commun. Nonlinear Sci. Numer. Simulat., 18(2013), 3075-3085.
[21] Z.-L. Zhu, W. Zhang, K.-W. Wong, H. Yu, A chaos-based symmetric image encryption scheme using a bit-level permutation, Information Sciences, 181(2011), 1171-1186.
[22] L. Teng, X. Wang, A bit-level image encryption algorithm based on spatiotemporal chaotic system and self-adaptive, Optics Communications, 285(2012), 4048-4054.
[23] W. Zhang, K.-W. Wong, H. Yu, Z.-L. Zhu, An image encryption scheme using lightweight bit-level confusion and cascade cross circular diffusion. Optics Communications, 285 (2012), 2343- 2354.
[24] W. Zhang, K.-W. Wong, H. Yu, Z.-L. Zhu, Asymmetric color image encryption algorithm using the intrinsic features of bit distributions. Common. Nonlinear Sci. Numer. Simulat., 18 (2013), 584-600.
[25] W. Zhang, H. Yu, Z. Zhu, Color image encryption based on paired interpermuting planes, Optics Communications, 338(2015), 199-208.
[26] X. Wang, H. Zhang, A color image encryption with heterogeneous bit-permutation and correlated chaos, Optics Communications, 342(2015), 51-60.
[27] X. Y. Wang, J. F. Zhao, H. J. Liu, A new image encryption algorithm based on chaos, Optics Communications, 285(2012), 562-566.
[28] M. Haeri, M. S. Tavazoei, Comparison of different one-dimensional maps as chaotic search pattern in chaos optimization algorithms, Appl. Math. Comput., 187(2007), 1076-1085.
[29] M. Hasler and Y. L. Maistrenko, An introduction to the synchronization of chaotic systems: coupled skew tent map, IEEE Transactions on Circuits and Systems, 44(1997), 856-866.
[30] R. Ye, W. Guo, A Chaos-based Image Encryption Scheme Using Multimodal Skew Tent Maps, Journal of Emerging Trends in Computing and Information Sciences, 4:10(2013), 800-810.
[31] B. Schneier, Cryptography: Theory and Practice, CRC Press, Boca Raton, 1995.
[32] Y. Wang, K. W. Wong, X. F. Liao, G. R. Chen, A new chaos-based fast image encryption algorithm, Applied Soft Computing, 11(2011), 514-522.
[33] M. Wu, An improved discrete Arnold transform and its application in image scrambling and encryption, Acta Phys. Sin., 63:9(2014), 090504.
[34] C. E. Shannon, Communication theory of secrecy systems, Bell Syst. Tech. J, 28(1949), 656-715.
[35] G. Alvarez, S. Li, Some basic cryptographic requirements for chaos-based cryptosystem, International Journal of Bifurcation and Chaos, 16(2006), 2129-2151.
[36] IEEE Computer Society, IEEE standard for binary floating-point arithmetic, ANSI/IEEE std. 1985:754-1985.
[37] J. Chen, Z. Zhu, C. Fu, H. Yu, L. Zhang, An efficient image encryption scheme using gray code based permutation approach, Optics and Lasers in Engineering, 67(2015), 191-204.
[38] K. Wong, B. Kwok, W. Law, A fast image encryption scheme based on a chaotic standard map, Physics Letter A, 372:15(2008), 2645-2652.
[39] K. Wong, B. Kwok, C. Yuen, An efficient diffusion approach for chaos-based image encryption, Chaos, Solitons and Fractals, 41:5(2009), 2652-2663.