

Received September 13, 2020, accepted September 15, 2020, date of publication September 18, 2020, date of current version October 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3024869

A New Image Encryption Scheme Based on a Novel One-Dimensional Chaotic System

XINGYUAN WANG AND PENGBO LIU 

School of Information Science and Technology, Dalian Maritime University, Dalian 116026, China

Corresponding authors: Xingyuan Wang (xywang@dmlu.edu.cn) and Pengbo Liu (18637190097@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61672124, in part by the Password Theory Project of the 13th Five-Year Plan National Cryptography Development Fund under Grant MMJJ20170203, in part by the Liaoning Province Science and Technology Innovation Leading Talents Program Project under Grant XLYC1802013, in part by the Key Research and Development Projects of Liaoning Province under Grant 2019020105-JH2/103, and in part by the Jinan City 20 Universities Funding Projects Introducing Innovation Team Program under Grant 2019GXRC031.

ABSTRACT This article introduces a novel one-dimensional sine chaotic system (1DSCS) with large parameter interval. The evaluation of 1DSCS indicates that the system has good chaotic characteristics and large parameters space. Based on 1DSCS, a new image encryption scheme is proposed. First, the image is scrambled by the row and column indexes, and then scrambled by the dynamic parameters Arnold map. This scheme avoids the inadequacy of row-column index scrambling and the periodicity of Arnold map. Second, four highly sensitive dynamic diffusion formulas related to plaintext are designed, and the dynamic formulas are selected through the chaotic sequence generated by 1DSCS. Third, the scheme proposed in this article can also be applied to color image encryption. The experimental results demonstrate that the 1DSCS system is suitable for image encryption and the encryption scheme has good security to resist common attacks.

INDEX TERMS Chaotic system, dynamic diffusion, image encryption, scrambling.


I. INTRODUCTION

The era of big data has come, and the transmission rate of information and communication has developed rapidly. Compared with text, image is gradually becoming the most important information carrier of daily information transmission because of its rich information. Image has been involved in the daily chat of personal privacy, military information, medicine, high technology and other key areas. With the further influence of image in various fields of society, it is extremely important for us to ensure the information security of image [1]–[3].

With the development of chaos theory, more and more scholars apply chaos to image encryption [4]–[7]. The chaotic system is sensitive to its parameters and initial values. The chaotic sequence generated by the chaotic system has ergodicity, non-convergence and pseudo-randomness. These characteristics can effectively improve the information security of image encryption system [8], [9]. Therefore, many image encryption algorithms based on chaotic system have been proposed [10], [11]. Because one-dimensional

chaotic system is easy to realize, a variety of image encryption schemes are designed based on one-dimensional chaotic system. Ye and Huang [12] introduced a self-adaptive image encryption algorithm based on an intertwining logistic map. Belazi and El-Latif [13] applied the sine map to enhance the image encryption algorithm. However, one-dimensional chaotic system has the defects of short period, small parameter ranges and unequal distribution of chaotic sequence. To solve those problems, some scholars proposed new one-dimensional chaotic systems. Zhou *et al.* [14] applied a new one-dimensional chaotic system based on seed map, which extends the system parameter range to $[0, 4]$. Patro *et al.* [15] proposed PWLCM system, which avoid the weakness of using a single chaotic map. In this article, a novel one-dimensional sine large parameter interval chaotic system is designed. It is controlled by two parameters. The system analysis shows that the two parameters have a wide range of values. Compared with other 1D chaotic systems, 1DSCS has better chaotic behavior. We apply 1DSCS to generate chaotic sequence and propose a new image encryption scheme based on 1DSCS.

At present, the mainstream image encryption algorithm is composed of scrambling and diffusion [16]–[20]. Common

The associate editor coordinating the review of this manuscript and approving it for publication was Yizhang Jiang .

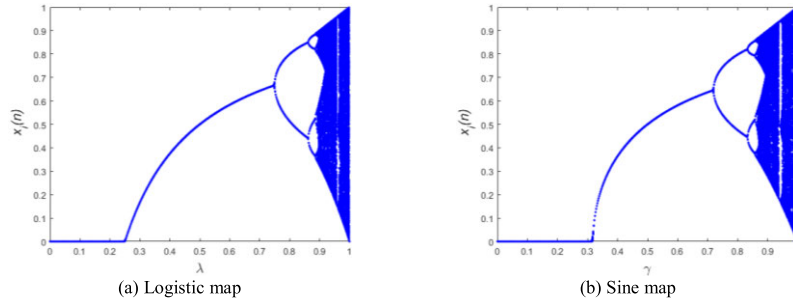


FIGURE 1. The bifurcation diagrams.

scrambling algorithms include magic square transformation, Arnold map, row-column index scrambling, etc. Although these methods have good scrambling effect, they also have their own shortcomings. Arnold map and magic square transformation have periodicity, which means that the image will return to the original image after a certain number of transformations. For row-column index scrambling, row and column indexes are used as scrambling units. Although the method is fast, each column is indexed with repeated row index, which makes the scrambling insufficient. In order to solve these problems, a scrambling algorithm is proposed in this article. This algorithm makes Arnold parameters dynamic and combines Arnold map with row-column index scrambling. In the diffusion stage, we proposed a remainder selection diffusion method, which selects the dynamic diffusion formula by taking the residual of the chaotic sequence generated.

The remaining structure of this article is as follows. The second chapter introduces the 1D chaotic systems, Arnold map and 1DSCS system. The third chapter presents the image encryption steps based on 1DSCS chaotic system. The fourth chapter shows the simulation results and the various tests results of the encryption algorithm. The simulation results and test results of color image are shown in the fifth chapter. Finally, the conclusion is in the sixth chapter.

II. RELATED WORK

This chapter introduces the proposed chaotic system and related analysis.

A. LOGISTIC MAP

Logistic map is a very classical one-dimensional chaotic system, which was used to present the changes of biological population [21]. In recent years, because of its simple structure and unpredictable iterative behavior, it is often used in the field of cryptography to improve the security of encryption [22]. The mathematical definition of Logistic map can be expressed as follow:

$$x_{n+1} = 4\lambda x_n(1 - x_n), \quad (1)$$

where λ is the control parameter, x_n is the output chaotic sequence. The bifurcation of Logistic map is shown in Fig. 1(a). We can see from Fig. 1(a) that when $\lambda \in (0.87, 1]$,

the range of x_n is $(0, 1)$. This means that the chaotic system has a good chaotic behavior.

B. SINE MAP

Sine map is also a 1D chaotic system with simple structure but very complex unpredictability. The definition of sine map can be expressed as follow:

$$x_{n+1} = \gamma \sin(\pi x_n), \quad (2)$$

where $x_n \in (0, 1)$, when the control parameter $\gamma \in (0.87, 1]$. Fig. 1(b) shows the bifurcation of Sine map. As shown in Fig. 1(b), sine map has many similar characteristics with logistic map.

C. 1-D SINE LARGE PARAMETER INTERVAL CHAOTIC SYSTEM

In this section, a novel 1D sine large parameter interval chaotic system is introduced (1DSCS). The system 1DSCS is defined as follow:

$$x_{n+1} = (\mu(3+2\lambda)(1 - \sin(\pi x_n))) \bmod 1. \quad (3)$$

The range of parameter μ that makes 1DSCS in chaotic state will change with the value of parameter λ . After many experiments, we found that 1DSCS has stable and good chaotic behavior when the control parameter $\lambda \in (0, +\infty)$ and $\mu \in (4, +\infty)$. 1DSCS was designed based on Sine map. Moreover, the use of two parameters gives the 1DSCS better chaotic characteristics. When the initial value $x_0 \in (0, 1)$, the x_n generated by 1DSCS iteration is uniformly distributed in $[0, 1]$.

D. 1DSCS BIFURCATION DIAGRAM

Theoretically, the trajectory of a good chaotic system is random and non-adjacent [23]. After the parameters and initial values are determined, the chaotic sequence generated by the system iteration is distributed uniformly in the form of pseudo-random. To observe the 1DSCS dynamic behavior, the bifurcation diagrams of 1DSCS are given in Fig. 2. Fig. 2(a), (b) show the bifurcation of 1DSCS when $\mu = 5$. We notice that when the parameter λ is in all the intervals, the data points are all uniformly distributed in $[0, 1]$ and the system have good chaotic behavior. Fig. 2(c), (d) show the

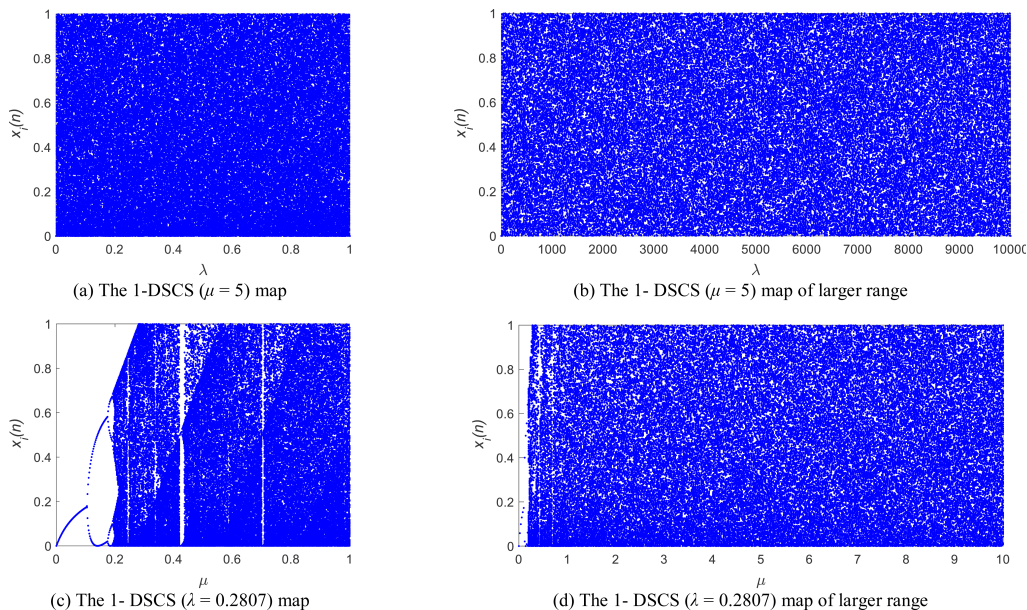


FIGURE 2. The bifurcation diagrams of 1-DSCS.

bifurcation of 1DSCS when $\lambda = 0.2807$. We can find that when the output of the 1DSCS exhibits chaotic behavior with the increase of the parameter μ . As can be seen from Fig. 1, in the chaotic state, the range of both sine map and logistic map are narrow. Compared with sine map and logistic map, the track distribution of 1DSCS covers most of the intervals.

E. LYAPUNOV EXPONENTS OF 1DSCS

Chaotic systems have initial value sensitivity, which means that small changes in the input value will cause unpredictable changes in the output value. The Lyapunov exponent (LE) is an important index to evaluate the nonlinear dynamic behavior of chaotic systems [24]. LE can reflect the separation degree of adjacent orbits with time, which can effectively indicate whether there is chaotic behavior in the system. In this section the LE is used to evaluate the sensitivity of chaotic system. The formula for calculating LE is as follow:

$$\omega = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln |g'(x_i)|, \tag{4}$$

where $g(x_i)$ is the chaotic system. A negative LE value indicates that the system is contracting on this timeline. A positive LE value indicates that the system is constantly expanding and folding on this timeline. Therefore, the system has good chaotic behavior when the value of LE is positive. Fig. 3 shows the LE results of 1DSCS and different maps. We noticed that 1DSCS has larger exponents than sine map and logistic map in Fig. 3(a). As shown in Fig. 3(b), all the values of LE are positive and become larger as the value of λ increases when $\mu = 5$. We can infer that when the parameter λ is in the range of $[0, +\infty)$, 1DSCS has chaotic behavior. Fig. 3(c) shows the LE results when $\lambda = 0.2807$, we notice

that LE is positive when $\mu \in [0.79, 1.02] \cup [1.04, 3.50] \cup [3.52, +\infty]$.

F. SHANNON ENTROPY

Shannon entropy (SE) reflects the chaotic degree of the sequence generated by chaotic system [25]. The more ordered the chaotic sequence generated by the system, the lower SE it has. Conversely, the higher the value of SE, the more disorderly the chaotic sequence generated by the system. Fig. 4 depicts the SE value of the proposed 1DSCS and compares it with the classic chaotic system Logistic and Sine’s SE.

It can be seen from Fig. 4 that 1DSCS has larger and more stable SE values than logistic and sine systems. We can conclude that 1DSCS has better ergodicity and randomness.

G. ARNOLD MAP

Arnold map can make the image continuously pull and fold, so the image can hide the image information well [26]. However, the periodicity of Arnold map makes the image return to its original state after a certain number of transformations. Arnold map is defined as follow:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & p \\ q & qp + 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \pmod{M}, \tag{5}$$

where M is the side length of the image. The p and q are the control parameters of Arnold map. The following formula is the inverse transformation of Arnold map:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & p \\ q & qp + 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} \pmod{M}. \tag{6}$$

III. 1-DSCS-BASED ENCRYPTION ALGORITHM

In this chapter, an image encryption algorithm based on 1DSCS is introduced. In the scrambling stage, a method of

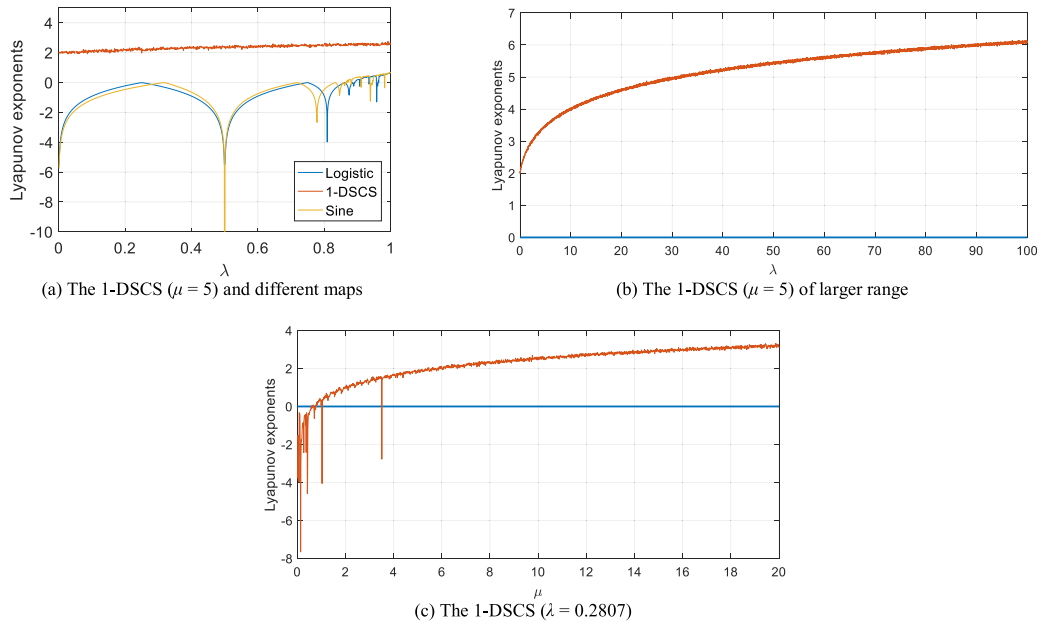


FIGURE 3. The results of LE.

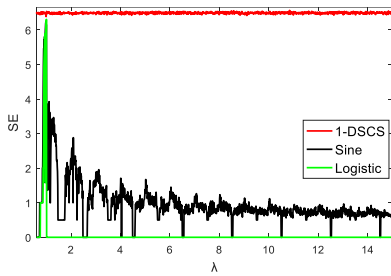


FIGURE 4. The results of SE.

combining row-column sorting index and Arnold map is proposed. The method of selecting dynamic diffusion formula based on remainder is proposed in the diffusion stage.

A. SECRET KEY GENERATION

The secret key is divided into two parts: the first part K_1 is generated by SHA-512, and the second part K_2 is user-defined key. The steps to generate K_1 are as follows:

Step 1: Convert the plain image P to a hexadecimal string by SHA-512.

Step 2: Convert the hexadecimal string to a binary array, then convert the first 504 digits of the array to 12 decimals.

Step 3: Take the average of every 4 decimals as k_1, k_2, k_3 .

K_1 is composed of k_1, k_2 , and k_3 . And K_2 is composed of k_4, k_5 . Refer to the range of 1DSCS system parameters, k_4 can be taken in $[4, +\infty)$, k_5 can be taken in $[0, +\infty)$. The parameters and initial values required for the encryption process are generated according to the following formula:

$$\begin{aligned} \mu &= k_1 + k_4, \\ \lambda &= k_2 + k_5, \\ x_0 &= k_3. \end{aligned} \tag{7}$$

B. SCRAMBLING ALGORITHM

In scrambling stage, there are two scrambling operations. The first scrambling is performed by the row-column index. And the second scrambling is performed by Arnold map. The specific scrambling steps are as follows:

For the plain image P of size $M \times M$ to be encrypted, the specific scrambling algorithm is as follow. In particular, if the image P does not satisfy $M \times M$, it needs to reach this scale by filling 0.

1) FIRST SCRAMBLING OPERATIONS

Step 1: Iterate 1DSCS $2 \times M$ times to generate two sequences r_1 and r_2 with size of $1 \times M$. The parameters and initial values of 1DSCS are given in (7).

Step 2: Build two two-dimensional arrays H_1 and H_2 with 2-row and M -column. The first row of both two arrays is set to 1, 2, 3... M . and put the two sequences r_1 and r_2 into the second row of the two arrays H_1 and H_2 separately. When sorting H_1 and H_2 according to the second row, the positions of the elements of the first row will change. Then, separately take the first row of H_1 and H_2 as the row index S_1 and column index S_2 .

Step 3: Use the following formula to get the first scrambled image P_1 :

$$P_1(i, j) = P(S_1(i), S_2(i)). \tag{8}$$

2) SECOND SCRAMBLING OPERATIONS

Step 1: Continue to iterate 1DSCS 40 times to get the sequence E . In (5), the parameters of Arnold map include p and q , define the parameters p and q as:

$$\begin{aligned} E &= \text{floor}(E \times 10^2) + 3, \\ p &= E(i), \\ q &= E(20 + i). \end{aligned} \tag{9}$$

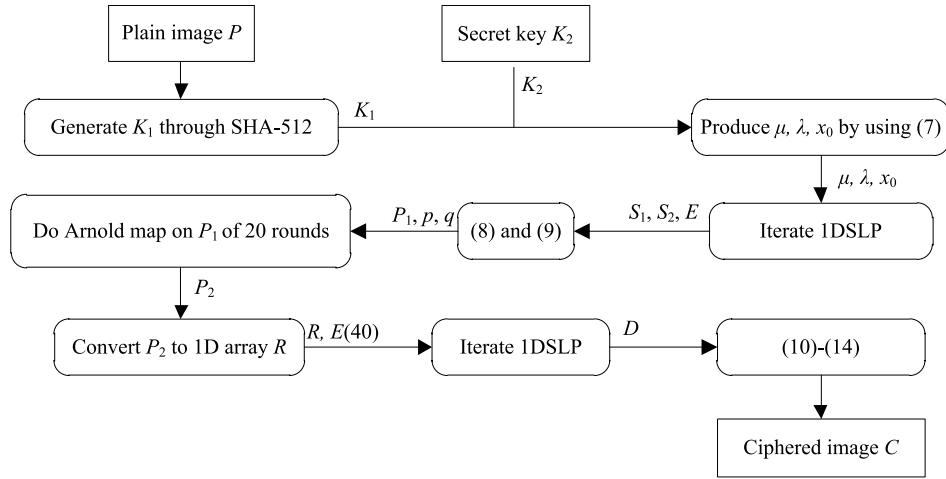


FIGURE 5. Encryption flow chart.

Step 2: Do Arnold map on P_1 of 20 rounds, and then the scrambled image P_2 is generated.

C. DIFFUSION ALGORITHM

In diffusion stage, the dynamic diffusion formula is selected by the modulus of chaotic sequence. Compared with the diffusion algorithm with only one formula, the diffusion algorithm proposed in this article is more complicated and difficult to crack. Use the following steps to complete the diffusion of the image:

Step 1: Convert the scrambled image P_2 to a one-dimensional array R . Iterate 1DSCS system to construct a sequence D with size of $M \times M$, the initial value is set to $E(40)$. Perform the following operations:

$$F = \text{floor}(D \times 10^6) \bmod 4. \tag{10}$$

$$T = \text{floor}(R \times 10^{10}) \bmod 256. \tag{11}$$

$$G = 0.99 + D \times 10^{-2}. \tag{12}$$

Step 2: Use the following operation to get the diffusion sequence c .

$$c(1) = R(1) \oplus \text{mod}(\text{floor}(T(1) + 4 \times (G(1) \times k_1 \times (1 - k_1)) \times 10^{10}), 256). \tag{13}$$

$$c(i) = R(i) \oplus T(i) \oplus \text{mod}(\text{floor}(4 \times G(i) \times (c(i - 1)/256) \times (1 - (c(i - 1)/256)) \times 10^{10}), 256), \quad F(i) = 0.$$

$$c(i) = R(i) \oplus T(i) \oplus \text{mod}(\text{floor}(G(i) \times \sin(\pi \times c(i - 1)/256) \times 10^{10}), 256), \quad F(i) = 1.$$

$$c(i) = \text{mod}(R(i) + T(i) + \text{floor}(4 \times G(i) \times (c(i - 1)/256) \times (1 - (c(i - 1)/256)) \times 10^{10}), 256), \quad F(i) = 2.$$

$$c(i) = \text{mod}(R(i) + T(i) + \text{floor}(G(i) \times \sin(\pi \times c(i - 1)/256) \times 10^{10}), 256), \quad F(i) = 3. \tag{14}$$

Step 3: Convert the sequence c into matrix C with size of $M \times M$.

Finally, the encrypted image C is obtained. Fig. 5 is a flow chart that more intuitively describes the steps of image encryption.

D. DECRYPTION ALGORITHM

For a known secret key K , since the decryption process is the inverse of the encryption process, the generation steps of F , T , G , E , S_1 , S_2 are the same as the encryption process. The other specific steps are as follows:

Step 1: Convert the encrypted image C to a one-dimensional array c . Use the following formulas to get the sequence R :

$$R(1) = c(1) \oplus \text{mod}(\text{floor}(T(1) + 4 \times (G(1) \times k_1 \times (1 - k_1)) \times 10^{10}), 256). \tag{15}$$

$$R(i) = c(i) \oplus T(i) \oplus \text{mod}(\text{floor}(4 \times G(i) \times (c(i - 1)/256) \times (1 - (c(i - 1)/256)) \times 10^{10}), 256), \quad F(i) = 0.$$

$$R(i) = c(i) \oplus T(i) \oplus \text{mod}(\text{floor}(G(i) \times \sin(\pi \times c(i - 1)/256) \times 10^{10}), 256), \quad F(i) = 1.$$

$$R(i) = \text{mod}(c(i) - T(i) - \text{floor}(4 \times G(i) \times (c(i - 1)/256) \times (1 - (c(i - 1)/256)) \times 10^{10}), 256), \quad F(i) = 2.$$

$$R(i) = \text{mod}(c(i) - T(i) - \text{floor}(G(i) \times \sin(\pi \times c(i - 1)/256) \times 10^{10}), 256), \quad F(i) = 3. \tag{16}$$

Step 2: Convert R into matrix P_2 with size of $M \times M$. Do the inverse transformation of Arnold map on P_1 of 20 rounds. The parameters of p and q are shown in (9).

Step 3: Use the following formula to get the plain image P :

$$P(S_1(i), S_2(i)) = P_1(i, j). \tag{17}$$

IV. PERFORMANCE ANALYSIS OF GRAY IMAGES

In this chapter, the effect and security of the encryption algorithm are tested in various aspects through gray-scale images

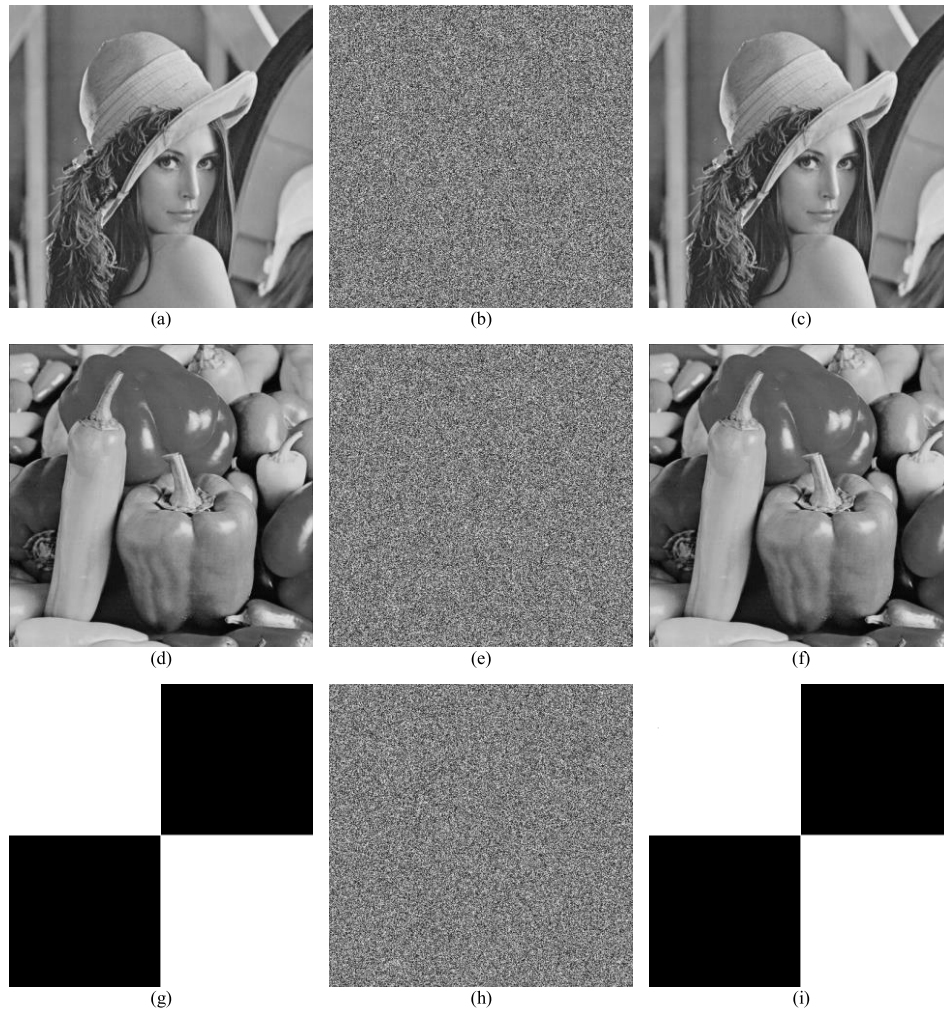


FIGURE 6. Simulation results of gray-scale images. (a) Lena (b) Encrypted Lena (c) Decryption of Lena (d) Pepper (e) Encrypted Pepper (f) Decryption of Pepper (g) Black-white (h) Encrypted Black-white (i) Decryption of Black-white.

A. SIMULATION RESULTS

A good image encryption method can well hide the information contained in the original image [27]. The grayscale images of Lena, Pepper and Black-white with size of 512×512 are used to test the simulation experiments of encryption and decryption algorithms. The simulation results are shown in Fig. 6. It can be seen from Fig. 6 that the encrypted image cannot see any information of the original image.

B. KEY SPACE ANALYSIS

The size of secret key space is one of the important factors that affect the security of encryption. The secret key of the algorithm proposed in this article consists of two parts. K_1 is converted from 504-bit binary, and its space size is 2^{504} . K_2 is a user-defined value, theoretically its spatial range is infinite. The key space of both K_1 and K_2 can resist violent attacks.

C. KEY SENSITIVITY ANALYSIS

In order to resist violent attacks, encryption algorithm should be sensitive to secret key. This means that the decrypted

image cannot obtain any useful information of the original image through the wrong key. In this article, the secret key K is composed of $k_1, k_2, k_3, k_4,$ and k_5 . In this test, secret key K is used to encrypt image Pepper-512, where $k_1 = 0.4774, k_2 = 0.5890, k_3 = 0.2667, k_4 = 14.14447, k_5 = 13.34587$. The correct original image P can be obtained through the correct key K (shown in Fig. 7(a)). Add 10^{-13} to $k_1, k_2, k_3, k_4,$ and k_5 of K respectively to obtain secret keys $K_1, K_2, K_3, K_4,$ and K_5 . Using $K_1, K_2, K_3, K_4,$ and K_5 to decrypt the encrypted image cannot get the correct image (shown in Figs. 7(b)-(f)).

D. HISTOGRAM ANALYSIS

The histogram can reflect the distribution characteristics of an image pixel value. Finding useful information from the distribution of pixel values is often used as a means of known-cipher attacks. Therefore, the histogram of the ciphered image should be evenly distributed so that the information of the original image cannot be displayed. The Lena's, Pepper's and Finger's histograms of plain image and its ciphered images are shown in Fig. 8. As Fig. 8(b), Fig. 8(c) and

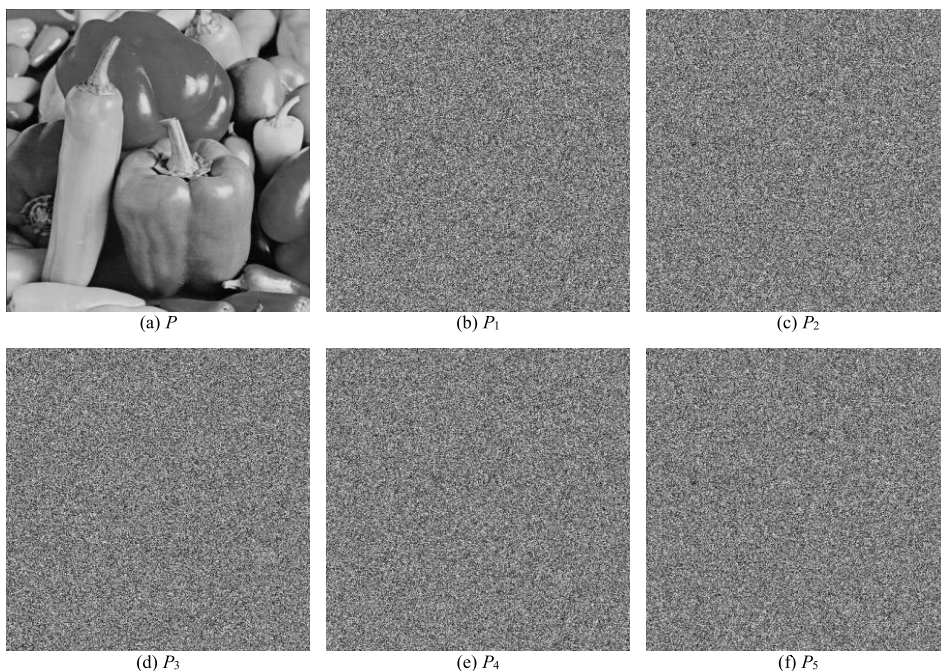


FIGURE 7. Key sensitivity analysis.

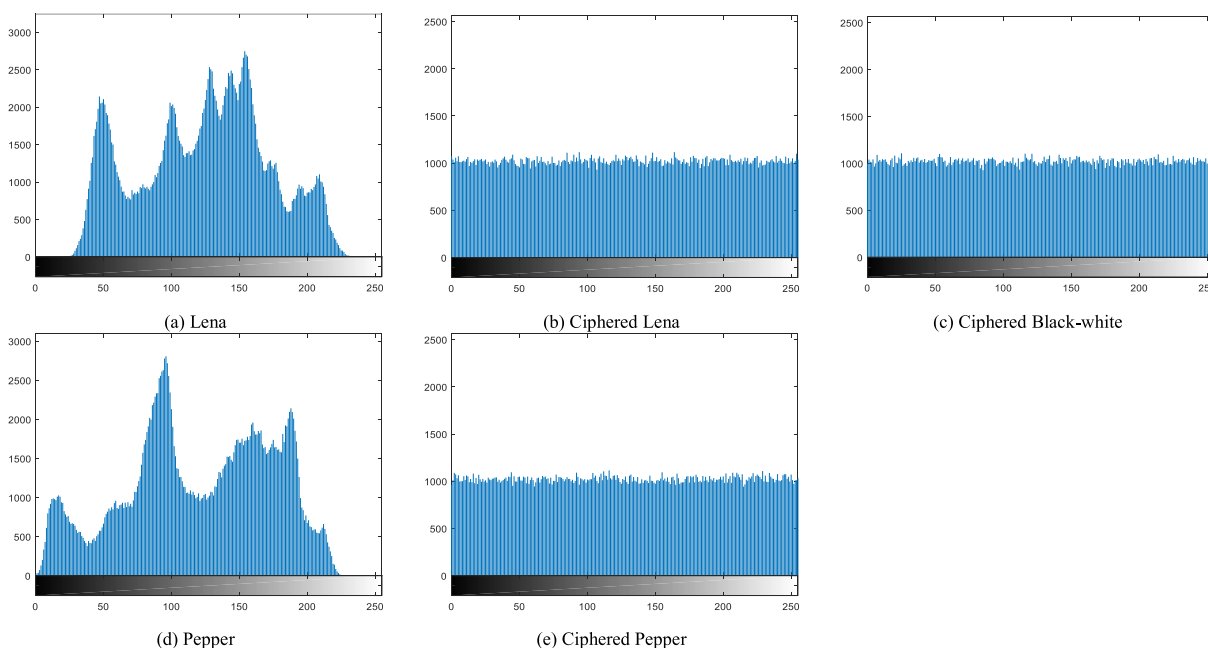


FIGURE 8. Histograms of plain images and ciphered images.

Fig. 8(e) shown, the histograms of the encrypted images are more evenly distributed than the histogram of the original image (shown in Fig. 8(a) and Fig. 8(d)). This indicates that it is difficult for attackers to obtain useful information by statistical analysis.

We further perform histogram analysis of variance and Chi-square test (χ^2 test) on grayscale images of different sizes. Variance is a quantitative analysis of the histogram and

is calculated by the following formula [28]–[30]:

$$\text{var}(Y) = \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n \frac{(y_i - y_j)^2}{2}. \tag{18}$$

where y_i is the number of pixels whose pixel value equals i and y_j is the number of pixels whose pixel value equals j ; Y is a one-dimensional array that records the number of

TABLE 1. Results and comparison of histogram variance.

Algorithms	Images	Original	Encrypted
Our algorithm	Lena (256 × 256)	59494.57	262.46
	Lena (512 × 512)	637992.58	931.49
	Avg. of 6 images (256 × 256)	2728815.13	259.37
	Avg. of 16 images (512 × 512)	18997069.34	1040.02
	Avg. of 3 images (1024 × 1024)	52921631.36	4060.48
Ref. [28]	Lena (256 × 256)	38951	676.8
	Lena (512 × 512)	633400	974.8
Ref. [29]	Lena (512 × 512)	638716.843	1027.593
Ref. [30]	Lena (256 × 256)	30665.703	262.5000

TABLE 2. χ^2 test.

Images	Lena	Boats	Pepper	Baboon	Harbour	Black-white
Plain image	158875	383971	121057	180257	9253412	33292543
Ciphered image	259.2481	249.7813	235.8965	219.2578	260.0254	242.2832
Results	Pass	Pass	Pass	Pass	Pass	Pass

occurrences of each pixel value; $\text{var}(Y)$ is the variance of Y . The low $\text{var}(Y)$ indicates that the pixel value distribution of gray image is highly uniform.

χ^2 test can be used to detect whether the pixel value distribution of the encrypted image is uniform. χ^2 can be calculated by using (19).

$$\chi^2 = \sum_{i=0}^{255} \frac{(f_i - f_0)^2}{f_0} \tag{19}$$

In (19), f_i represents the number of statistics of pixel value i in the image and $f_0 = (M \times N)/256$. Here, $M \times N$ is the size of the image. In theory, if the value of χ^2 is larger, the pixel distribution is more uneven, and the image contains more information. Generally, when the significant level is $\alpha = 0.05$, $\chi_{0.05}^2 = 293.24783$. When the significant level is $\alpha = 0.01$, $\chi_{0.01}^2 = 310.457$ [28].

Table 1 shows the histogram variance of different size images from USC-SIPI database. It also shows the comparison between the proposed scheme and other schemes. In Table 1, we can see that the variance of the encrypted image is much smaller than that of the original image. We can also see that compared with other schemes, our scheme has smaller histogram variance for Lena image.

Table 2 shows the χ^2 test results for six images. As seen in Table 2, compared with the plain image, the χ^2 values of the encrypted images are much smaller and all the results passed.

E. CORRELATION ANALYSIS

The correlation of adjacent pixels indicates the strength of the linear relationship between the two variables [31]. The original image contains useful information and has a high recognition rate, which indicates that the correlation between adjacent pixels of the image is high. The correlation between adjacent pixels of the image should be eliminated for the security of ciphered images. In order to test the correlation

analysis, 6000 pixels are randomly selected from the vertical, diagonal and horizontal directions to test the plain image and the ciphered image. The correlation graphs of image Pepper are shown in Fig. 9. More intuitively, we use (20) and (21) to calculate the correlation value.

$$E(x) = \frac{1}{N} \sum_{i=1}^N x_i, \tag{20}$$

$$D(x) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))^2,$$

$$\text{cov}(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))(y_i - E(y)).$$

$$r_{xy} = \frac{\text{cov}(x, y)}{\sqrt{D(x)}\sqrt{D(y)}}. \tag{21}$$

The test results of the correlation values of Lena, Pepper, Harbour, Boats, Black-white, and other 16 grayscale images are shown in Table 3.

We can see from Fig. 9 that the distribution between adjacent pixels of the ciphered image is more uniformly dispersed than the plain image. As can be seen from Table 3, all the values of the ciphered image are far smaller than the plain image in the three directions of correlation coefficient test. Therefore, the algorithm has good security to resist statistical attacks.

F. INFORMATION ENTROPY

Information entropy is used to describe the confusion degree of image information, which is calculated by using (22) [31]:

$$H(m) = \sum_{j=0}^{2^N-1} p(m_j) \log_2 \frac{1}{p(m_j)}. \tag{22}$$

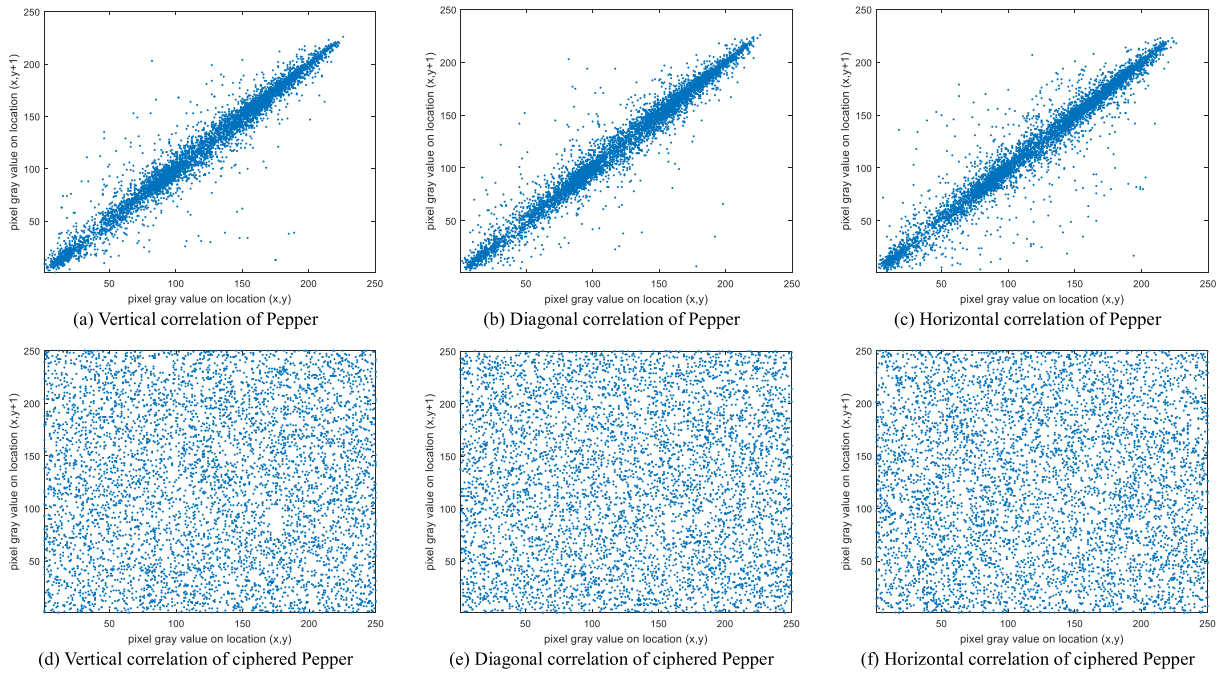


FIGURE 9. Correlation graphs of pepper.

TABLE 3. Correlation coefficients of images.

Images	Plain image			Cipher image		
	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Lena	0.97218	0.98566	0.95933	0.00096	0.00271	-0.00054
Pepper	0.97602	0.97943	0.96320	0.00151	0.00079	0.00024
Harbour	0.90415	0.81897	0.77272	-0.00018	-0.00043	0.00438
Boats	0.93783	0.97141	0.92232	0.00074	-0.00341	0.00087
Black-white	0.99613	0.99612	0.99226	-0.00201	-0.00101	0.00024
5.2.08	0.89651	0.93963	0.86030	-0.00087	0.00047	-0.00041
5.2.09	0.86183	0.90178	0.80477	0.00256	-0.00390	0.00674
5.2.10	0.92769	0.94005	0.89711	-0.00024	0.00142	-0.00648
7.1.01	0.92026	0.96203	0.90704	-0.00008	-0.00346	0.00133
7.1.02	0.94501	0.94585	0.89572	0.00053	0.00220	-0.00111
7.1.03	0.93208	0.94569	0.90165	-0.00484	-0.00052	-0.00176
7.1.04	0.96667	0.97642	0.95530	0.00163	-0.00265	0.00328
7.1.05	0.91150	0.94159	0.89268	-0.00246	-0.00228	-0.00313
7.1.06	0.90670	0.94060	0.88682	-0.00047	0.00051	0.00031
7.1.07	0.87669	0.88582	0.83779	0.00030	-0.00221	0.00421
7.1.08	0.92989	0.95772	0.92239	0.00028	-0.00083	-0.00025
7.1.09	0.93034	0.96560	0.91632	-0.00571	0.00257	-0.00488
7.1.10	0.94733	0.96455	0.93109	-0.00096	0.00258	0.00594
Boat512	0.97157	0.93755	0.92210	0.00757	0.00086	0.00075
Gray21.512	0.99984	0.99681	0.99665	0.00096	-0.00210	-0.00462
Ruler.512	0.46413	0.45569	-0.03059	0.00500	-0.00007	0.00110
Mean	0.91307	0.92424	0.86224	0.00020	-0.00046	0.00034
Ref. [32]	-	-	-	-0.00328	-0.00078	-0.00018
Ref. [33]	-	-	-	-0.0052	0.022	-0.0103

TABLE 4. Information entropy of images.

Images	Lena	Boats	Pepper	Harbour	Baboon
Plain image	7.4461	7.1914	7.5932	6.7834	7.3814
Our algorithm	7.9992	7.9993	7.9993	7.9993	7.9992

TABLE 5. Information entropy of different algorithms.

Algorithm	Cipher image
Proposed	7.99926
Ref. [34]	7.9972
Ref. [35]	7.9989
Ref. [36]	7.9993

TABLE 6. Local information entropy.

Images	Lena	Boats	Pepper	Baboon	Harbour	Black-white
Local information entropy	7.90230	7.90275	7.90272	7.90256	7.90271	7.90242
Pass or Fail	Pass	Pass	Pass	Pass	Pass	Pass

TABLE 7. Encryption quality test results (energy).

Images	Encrypted image	Ref. [38]	Ref. [39]
Cameraman (256 × 256)	0.015639	0.015644	-
All white	0.015687	0.015634	0.0151
All black	0.015639	0.015637	0.0144

In (22), $p(m_i)$ denotes the occurrence probability of pixel m_i . In theory, encrypted image has higher information confusion and security when the value of information entropy is closer to 8. The test results of information entropy of five images are shown in Table 4. As seen in Table 4, the information entropy of the images encrypted by our algorithm has been significantly improved, and all are close to the ideal value 8. And compared with Ref. [34]–[36] in Table 5, our algorithm has better information entropy, which means that the encrypted image is very messy and it is difficult for an attacker to obtain useful information from it. Therefore, encrypted image has good security.

G. LOCAL INFORMATION ENTROPY

Local information entropy is an index that reflects the degree of information confusion of the local encrypted image [31]. The local information entropy is calculated by using (23).

$$\overline{H}_{k,T_B}(M) = \sum_{i=1}^k \frac{H(M_i)}{k}. \tag{23}$$

In (23), M is the image, k and T_B indicate that k groups containing T_B pixels are randomly selected from the image M . $H(M_i)$ represents the information entropy of M_i and M_i consists of T_B pixels. In the test, at $k = 30$, confidence level $\alpha = 0.001$ and $T_B = 1936$, the encrypted image will meet the security requirements if the value of local information entropy should be between 7.90190131 and 7.90303733 [37].

TABLE 8. Encryption quality test results (contrast).

Images	Encrypted image	Ref. [38]	Ref. [39]
Cameraman (256 × 256)	10.6380	10.647	-
All white (256 × 256)	10.9829	10.4830	10.4780
All black (256 × 256)	10.4925	10.614	10.4398

TABLE 9. Encryption quality test results (homogeneity).

Images	Encrypted image	Ref. [38]	Ref. [39]
Cameraman (256 × 256)	0.38503	0.38827	-
All white (256 × 256)	0.37645	0.38996	0.3798
All black (256 × 256)	0.38921	0.38842	0.3894

TABLE 10. Encryption quality test results (correlation).

Images	Cameraman (256 × 256)	Black (256 × 256)	Lena (256 × 256)	Lena (512 × 512)
Plain image	0.92273	-	0.93582	0.95237
Encrypted image	0.00126	-0.00154	0.00232	0.00270

Calculation results of local information entropy are shown in Table 6. As seen in Table 6, the local information entropy of the six images all passed the test.

H. ENCRYPTION QUALITY MEASUREMENT

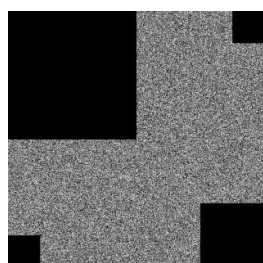
1) ENERGY

Energy is the square sum of the elements of the Gray Level Co-occurrence Matrix (GLCM), which reflects the uniformity of the gray image. GLCM is a statistical combination of pixel gray levels [38]. Energy is calculated by the following formula:

$$Energy = \sum_{x,y} p(x,y)^2. \tag{24}$$

TABLE 11. Evaluation of NPCR.

File name	NPCR (%)					
	Ref. [41]	Ref. [42]	Ref. [43]	Ref. [44]	Ref. [45]	proposed
256×256	$N_{\sigma}^* \geq 99.5693$					
5.1.09	99.6063	99.5956	99.6505	99.6064	99.6140	99.5804
5.1.10	99.6059	99.6018	99.6063	99.6154	99.5880	99.5972
5.1.11	99.6093	99.6079	99.6490	99.6244	99.6033	99.5987
5.1.12	99.6057	99.6063	99.6170	99.5703	99.5631	99.6002
5.1.13	99.6110	99.6155	99.5605	99.6109	99.5789	99.6201
5.1.14	99.6114	99.6124	99.6216	99.6364	99.6765	99.5895
512×512	$N_{\sigma}^* \geq 99.5893$					
5.2.08	-	99.6021	99.5987	99.5870	99.6037	99.6136
5.2.09	-	99.6082	99.6220	99.6260	99.6029	99.6048
5.2.10	-	99.6269	99.6162	99.6124	99.6124	99.6029
7.1.01	-	99.6017	99.6166	99.5992	99.6082	99.6029
7.1.02	-	99.6128	99.6109	99.6075	99.6174	99.6159
7.1.03	-	99.5968	99.6216	99.6079	99.6120	99.6243
7.1.04	-	99.6098	99.6090	99.5988	99.5911	99.6067
7.1.05	-	99.6021	99.6063	99.6170	99.6178	99.6098
7.1.06	-	99.6014	99.6101	99.6272	99.6174	99.6105
7.1.07	-	99.6136	99.6220	99.5931	99.5922	99.6025
7.1.08	-	99.6089	99.6101	99.6094	99.6056	99.5995
7.1.09	-	99.6079	99.5861	99.6162	99.6086	99.5945
7.1.10	-	99.6037	99.6120	99.6045	99.5941	99.6113
Boat512	-	99.5991	99.6086	99.6154	99.6101	99.5903
Gray21.512	-	99.6124	99.6040	99.6022	99.6159	99.6178
Ruler.512	-	99.6185	99.6227	99.6120	99.6212	99.6204
1024×1024	$N_{\sigma}^* \geq 99.5994$					
5.3.01	-	99.6094	99.6099	99.5931	99.6072	99.6063
5.3.02	-	99.6064	99.6099	99.6128	99.6116	99.6096
7.2.01	-	99.6080	99.6130	99.6156	99.6204	99.6093
Pass/All	6/6	25/25	23/25	23/25	24/25	25/25
Mean	99.6096	99.6076	99.6126	99.6088	99.6078	99.6055
Std	0.002623	0.007047	0.01717	0.01392	0.01406	0.01053



(a) Random data loss



(b) Decryption image of (a)

FIGURE 10. Clipping attack.

where $p(x, y)$ is the number of GLMC matrices. The range of energy value is [0, 1]. Low energy values confirm high degree of image disorder. Table 7 shows the energy values of encrypted images.

2) CONTRAST

Contrast analysis calculates the brightness contrast of adjacent pixels, which reflects the depth and clarity of texture grooves [38]. A good image encryption scheme requires higher contrast to verify that the texture is non-homogeneous. Contrast can be expressed by the following formula:

$$Contrast = \sum_{x,y}^M |x - y|^2 P(x, y). \tag{25}$$

where $p(x, y)$ is the number of GLMC matrices and M is the sum of rows and columns. The contrast test results are shown in Table 8.

3) HOMOGENEITY

Homogeneity reflects the closeness of the distribution of elements in GLCM relative to the diagonal of GLCM [38].

TABLE 12. Evaluation of UACI.

File name	UACI (%)					
	Ref. [41]	Ref. [42]	Ref. [43]	Ref. [44]	Ref. [45]	proposed
256×256	$(U_{\sigma}^{+-}, U_{\sigma}^{++}) = (33.2824, 33.6447)$					
5.1.09	33.4585	33.45034	33.4387	33.4456	33.4032	33.4602
5.1.10	33.4161	33.43234	33.4701	33.4946	33.3557	33.4087
5.1.11	33.4791	33.412039	33.4150	33.5541	33.4696	33.4700
5.1.12	33.4550	33.462423	33.5082	33.4302	33.4634	33.4890
5.1.13	33.4143	33.497386	33.4939	33.4438	33.3046	33.4552
5.1.14	33.4896	33.465875	33.7240	33.4655	33.4796	33.3661
512×512	$(U_{\sigma}^{+-}, U_{\sigma}^{++}) = (33.3730, 33.5541)$					
5.2.08	-	33.464625	33.4694	33.0080	33.4493	33.5378
5.2.09	-	33.48115	33.4704	33.4804	33.5077	33.4761
5.2.10	-	33.454698	33.5688	33.4563	33.4457	33.4522
7.1.01	-	33.476568	33.4531	33.5037	33.4890	33.4518
7.1.02	-	33.451951	33.3931	33.4237	33.4190	33.4638
7.1.03	-	33.412212	33.4599	33.4291	33.4689	33.4376
7.1.04	-	33.49961	33.4471	33.4739	33.4997	33.4808
7.1.05	-	33.405391	33.3758	33.4362	33.4313	33.5293
7.1.06	-	33.514572	33.4942	33.3954	33.4760	33.4154
7.1.07	-	33.52977	33.4876	33.4073	33.4470	33.4411
7.1.08	-	33.51067	33.5078	33.4332	33.5203	33.5279
7.1.09	-	33.437745	33.4584	33.4177	33.4704	33.4677
7.1.10	-	33.493778	33.4332	33.4344	33.4892	33.4366
Boat512	-	33.486903	33.4197	33.4654	33.5414	33.4451
Gray21.512	-	33.512627	33.4906	33.4608	33.4331	33.4406
Ruler.512	-	33.454171	33.5193	33.4262	33.4363	33.4837
1024×1024	$(U_{\sigma}^{+-}, U_{\sigma}^{++}) = (33.4183, 33.5088)$					
5.3.01	-	33.424132	33.4413	33.4585	33.4886	33.4719
5.3.02	-	33.498697	33.5189	33.4605	33.4384	33.4542
7.2.01	-	33.437057	33.4428	33.4556	33.4192	33.4848
Pass/All	6/6	25/25	22/25	25/25	25/25	25/25
Mean	33.4635	33.46667	33.4761	33.4344	33.4539	33.4619
Std	0.031335	0.035381	0.06716	0.09488	0.05053	0.03765

TABLE 13. Time test.

Images size	Encryption time	Decryption time
256×256	0.506	0.465
512×512	1.718	1.703
1024×1024	6.639	7.138

Homogeneity is defined as the following formula:

$$Homogeneity = \sum_{x,y} \frac{p(x,y)}{1 + |x - y|} \tag{26}$$

where $p(x,y)$ is the number of GLMC matrices. The range of homogeneity value is [0, 1]. Low homogeneity values confirm high security of encrypted image. Table 9 shows the homogeneity values of encrypted images.

4) CORRELATION

Correlation is to calculate the similarity degree of GLMC in row or column direction, which reflects the local gray correlation. Correlation is calculated by the following formula:

$$Corr = \frac{\sum_x \sum_y (xy)p(x,y) - \mu_x \mu_y}{\sigma_x \sigma_y} \tag{27}$$

where $p(x,y)$ is the number of GLMC matrices. The range of correlation value is [0, 1]. A low correlation value indicates that the local gray scale correlation is small, and the encryption algorithm has high security. Table 10 shows the correlation values of encrypted images. It can be seen from Table 10 that the correlation value of the encrypted image is far less than that of the original image.

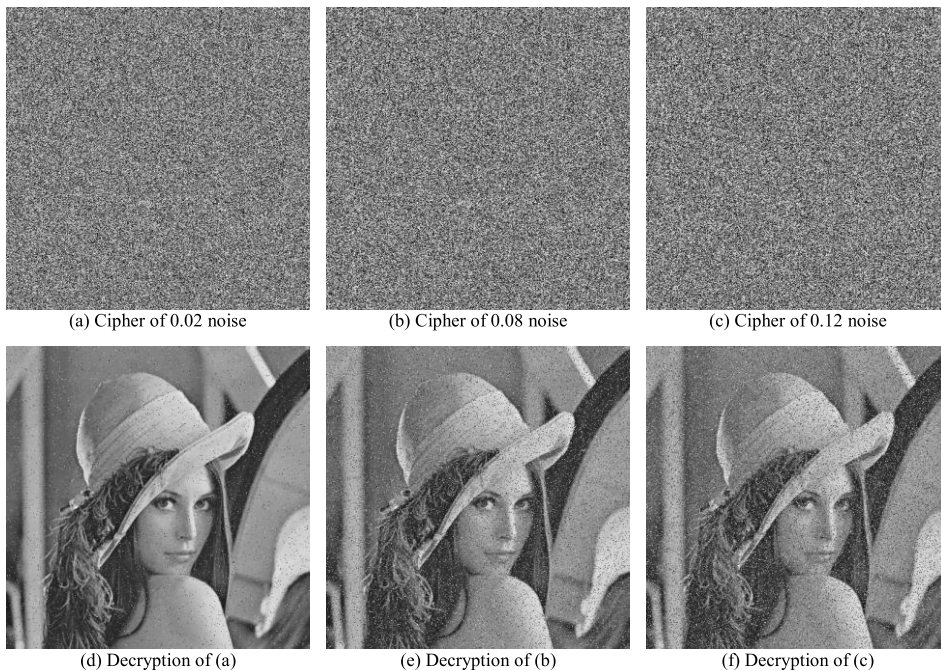


FIGURE 11. Test for noise attack.

TABLE 14. Evaluation of UACI.

Tests	Entropy of information	χ^2	Chang value $P(2, 5)$		Chang value $P(418, 314)$	
			NPCR (%)	UACI (%)	NPCR (%)	UACI (%)
R	7.99926	267.6055	99.6307	33.4882	99.6215	33.4922
G	7.99923	278.8125	99.6151	33.4306	99.6246	33.4499
B	7.99939	240.7480	99.6197	33.5330	99.6101	33.4936
Average	7.99929	262.3887	99.6218	33.4839	99.6187	33.4785

I. DIFFERENTIAL ATTACK ANALYSIS

Sometimes the attacker will get the rules of the encryption algorithm by analyzing the encrypted images with slight changes in the two plain images, which is called differential attack. The number of pixel changes rate (NPCR) and the unified average changing intensity (UACI) can be used to evaluate the difference between the two images. For the images with size of $M \times N$, NPCR and UACI are calculated by using (28) [31].

$$\begin{aligned}
 D(i, j) &= \begin{cases} 1, & C_1(i, j) \neq C_2(i, j) \\ 0, & C_1(i, j) = C_2(i, j), \end{cases} \\
 NPCR &= \frac{\sum_{i,j} D(i, j)}{M \times N} \times 100\%, \\
 UACI &= \frac{1}{M \times N} \left[\sum_{i,j} \frac{|C_1(i, j) - C_2(i, j)|}{255} \right] \times 100\%. \tag{28}
 \end{aligned}$$

In (28), $C(i, j)$ represents the value of position (i, j) in the image C , D is the grayscale level. In this test, C_2 is an image generated by randomly changing single pixel value of C_1 .

The critical value (N_σ^*) of NPCR and critical interval ($U_\sigma^{*-}, U_\sigma^{*+}$) of UACI are used to evaluate whether the algorithm passes the test. The algorithm can pass the test when the value of NPCR is higher than N_σ^* and the value of UACI is in the interval ($U_\sigma^{*-}, U_\sigma^{*+}$). According to [40], N_σ^* and ($U_\sigma^{*-}, U_\sigma^{*+}$) are defined as follows:

$$N_\sigma^* = \frac{D - \Phi^{-1}(\sigma)\sqrt{D/(M \times N)}}{D + 1}. \tag{29}$$

and

$$\begin{cases} U_\sigma^{*-} = \mu_u - \Phi^{-1}(\sigma/2)\sigma_u \\ U_\sigma^{*+} = \mu_u + \Phi^{-1}(\sigma/2)\sigma_u, \end{cases} \\
 \mu_u = \frac{D + 2}{3D + 3}, \\
 \sigma_u^2 = \frac{(D + 2)(D^2 + 2D + 3)}{18(D + 1)^2 D(M \times N)}. \tag{30}$$

For more accurate experimental results, we tested 25 gray images of different sizes from USC-SIPI database. Following the discussion in [40], σ was set to 0.5. The results of NPCR and UACI are shown in Table 11 and Table 12. We can see that the average values of NPCR and UACA are 99.6091% and 33.4418%, which are close to the optimal values 99.609% and

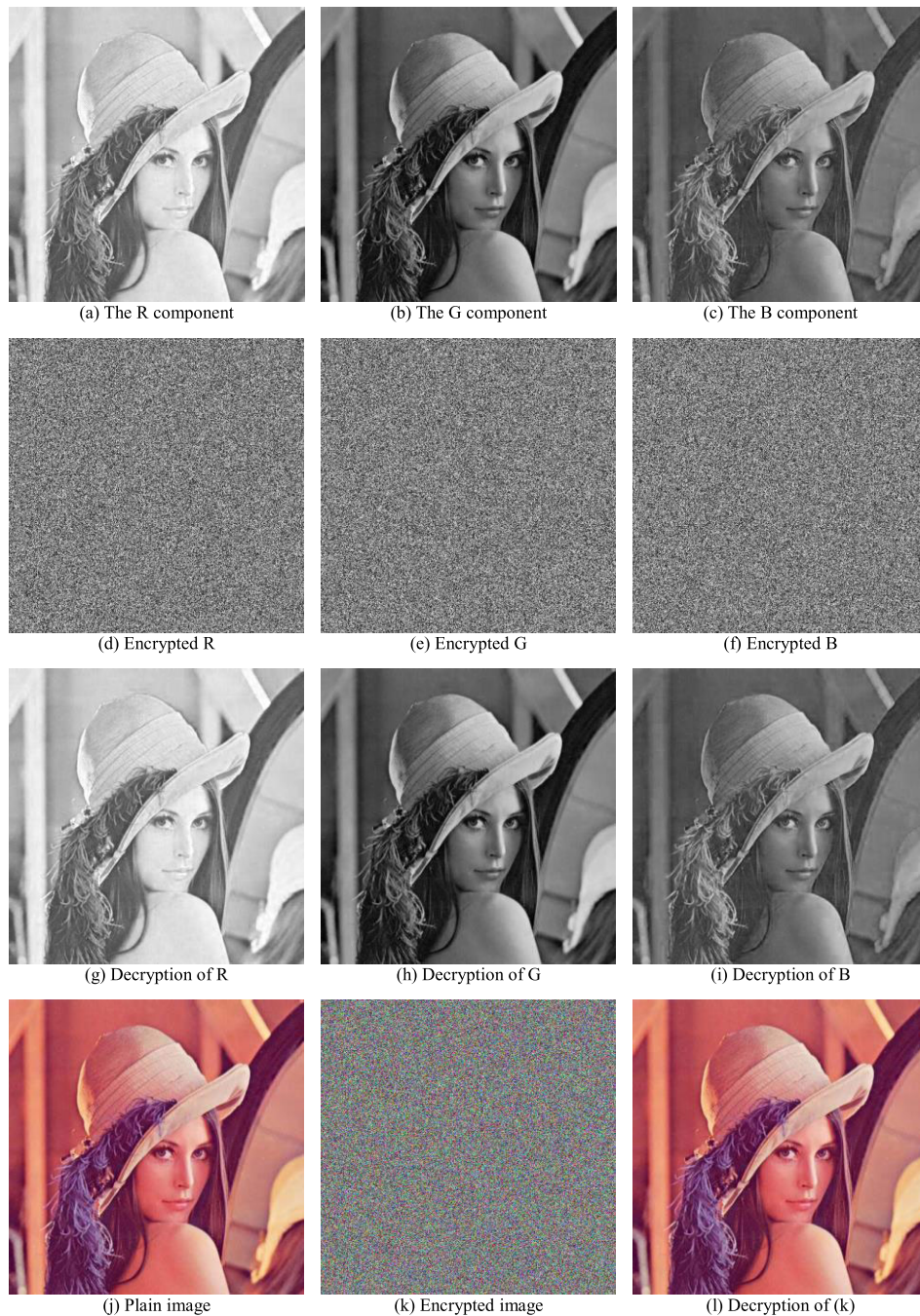


FIGURE 12. Simulation results of *Lena_color*.

33.464%. This shows that the proposed algorithm is highly sensitive to small changes in plaintext. Compared with other algorithms, our algorithm can pass the tests for all the pictures whereas other algorithms failed to pass the tests for individual images. This indicates that the algorithm has enough ability to resist differential attacks.

J. ROBUSTNESS ANALYSIS

In the process of image transmission or malicious interference, data loss or pixel change will occur. Robustness refers to the fact that the encrypted image can

still get effective information when the data is missing or changed. In this section, clipping attack and noise attack are used to test the anti-interference ability of the algorithm.

In order to simulate clipping attack, some pixels are randomly selected in the encrypted image and their pixel values are changed to 0 (shown in Fig. 10(a)). And the decryption image of Fig. 10(a) is as shown in Fig. 10(b). It can be seen from Fig. 10 that even if data is lost from different locations, the decrypted image can still obtain the information of the plain image.

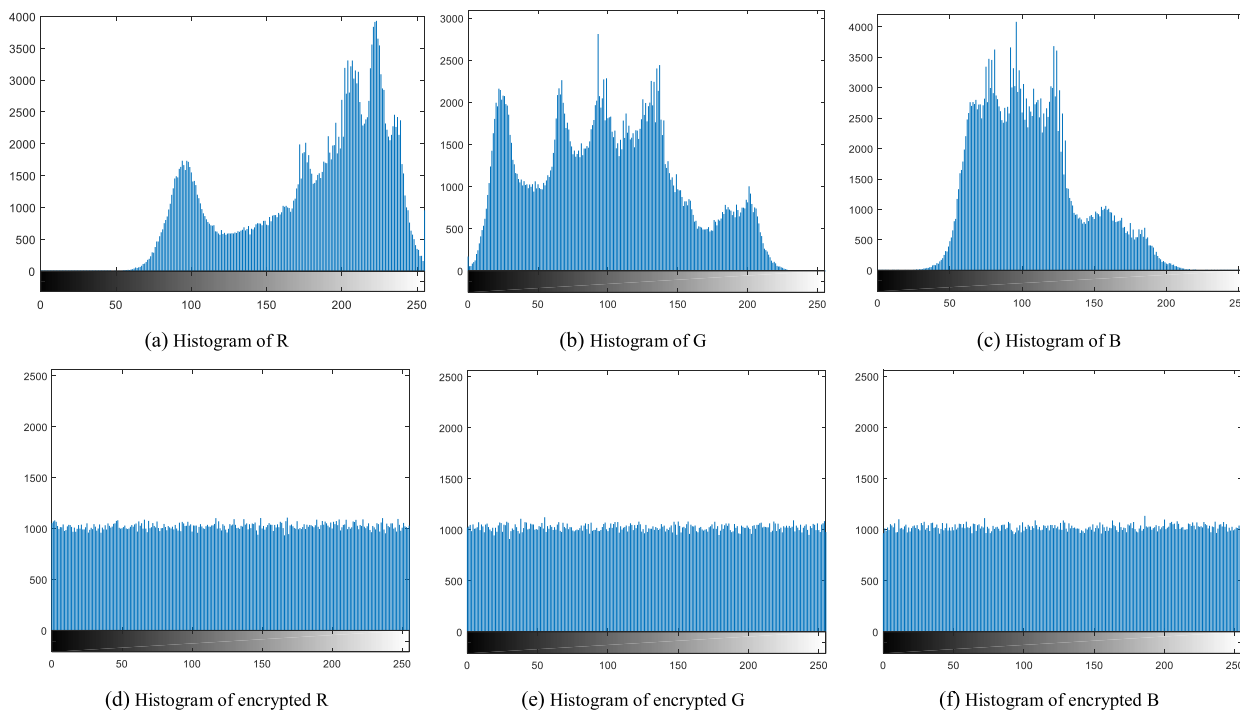


FIGURE 13. Histogram of R, G and B components.



FIGURE 14. Robustness analysis of color image.

In the noise test, 0.02, 0.08 and 0.12 Salt-and-Peppers noise are used to interfere with the encrypted image (shown in Figs. 11(a)-(c)). Figs. 11(d)-(f) shows that the decrypted images can display the information of the plain image under different levels of noise attack. As can be seen from Fig. 11 that the algorithm can effectively resist the noise attacks.

K. COMPUTATIONAL TIME ANALYSIS

The time consumption of the algorithm proposed in this article is mainly generated by Arnold map, and as the image size becomes larger, the time for Arnold map is also longer. The time tests are performed on Matlab 2016a, Intel Core i5-7500 CPU with 8 GB RAM and Window 10 operating system. The results of the time test are shown in Table 13.

V. PERFORMANCE ANALYSIS OF COLOR IMAGES

In this chapter, we applied the proposed encryption scheme to the encryption of color images and tested the security of the encrypted images.

A. SIMULATION RESULTS

The color image contains more information and is composed of three grayscale images R, G, and B. For the encryption of color images, first use the algorithm proposed in this article to encrypt R, G, and B, and then synthesize the encrypted R, G, and B into an encrypted image. The results of color image encryption and decryption are shown in Fig. 12.

B. HISTOGRAM RESULTS

As shown in Fig. 13, the Figs. 13(a)-(c) are the histograms of Figs. 12(a)-(c) and Figs. 13(d)-(f) are the histograms of Figs. 12(d)-(f). It can be seen from Fig. 13 that the histograms of the three encrypted images of R, G, and B are all uniform. It shows that the algorithm in this article also has a good encryption effect on color images.

C. ROBUSTNESS ANALYSIS

The robustness of the algorithm on color images is an important test indicator. Fig. 14 shows the clipping attack and

TABLE 15. Comparison analyses for grayscale images and color images.

Tests	Entropy of information	χ^2	NPCR (%)	UACI (%)
Avg. of grayscale images	7.99926	244.4154	99.6055	33.4619
Avg. of color images	7.99929	262.3887	99.6203	33.4812
Avg. of color images [46]	7.9993	-	99.60	33.32

noise attack on the color image. As seen from Fig. 14(b) and Fig. 14(d), the algorithm is very robust to color image, and the images decrypted by the attacked image can display the information of the plain image.

D. OTHER ANALYSIS

In this section, the information entropy, χ^2 , NPCR and UACI of encrypted images R, G, and B are tested respectively. Table 14 shows the test results. Furthermore, comparison for grayscale images and color images is shown in Table 15. From Table 14 and Table 15 we can see that the test result is close to the ideal value, which indicates that the encrypted color image has high security and the algorithm has good encryption effect for both grayscale image and color image.

VI. CONCLUSION

In this article, a new one-dimensional chaotic system 1DSCS is proposed. The bifurcation graph test and Lyapunov exponent test were carried out on 1DSCS. The experimental results show that 1DSCS has a good chaotic effect. Based on 1DSCS, a hybrid scrambling method and a dynamic diffusion method based on remainder are proposed. Then, the proposed encryption algorithm was simulated and tested in grayscale image and color image respectively. The tests include key space, correlation analysis, information entropy, χ^2 , NPCR, UACI, and robustness analysis. According to the simulation results and security analysis, it can be seen that the algorithm has a good encryption effect and can resist common attacks.

Although the experimental results show that our algorithm has high security, it has not been extended to practical applications at present. In future work, we will extend the algorithm from the theoretical stage to practical applications. In addition, we want to extend this algorithm to other fields, such as audio encryption and video encryption.

REFERENCES

- [1] W. Liu, K. Sun, and C. Zhu, "A fast image encryption algorithm based on chaotic map," *Opt. Lasers Eng.*, vol. 84, pp. 26–36, Sep. 2016.
- [2] N. Chidambaram, P. Raj, T. Karruppuswamy, and R. Amirtharajan, "An advanced framework for highly secure and cloud-based storage of colour images," *IET Image Process.*, to be published, doi: [10.1049/iet-ipr.2018.5654](https://doi.org/10.1049/iet-ipr.2018.5654).
- [3] S. Rethinam, R. Sundararaman, J. B. Rayappan, and R. Amirtharajan, "Ring oscillator as confusion—Diffusion agent: A complete TRNG drove image security," *IET Image Process.*, to be published, doi: [10.1049/iet-ipr.2019.0168](https://doi.org/10.1049/iet-ipr.2019.0168).
- [4] T. S. Ali and R. Ali, "A new chaos based color image encryption algorithm using permutation substitution and Boolean operation," *Multimedia Tools Appl.*, vol. 79, nos. 27–28, pp. 19853–19873, Jul. 2020.
- [5] I. Yasser, F. Khalifa, M. A. Mohamed, and A. S. Samrah, "A new image encryption scheme based on hybrid chaotic maps," *Complexity*, vol. 2020, pp. 1–23, Jul. 2020.
- [6] C. Zhu and K. Sun, "Cryptanalyzing and improving a novel color image encryption algorithm using RT-enhanced chaotic tent maps," *IEEE Access*, vol. 6, pp. 18759–18770, 2018.
- [7] Z.-H. Gan, X.-L. Chai, D.-J. Han, and Y.-R. Chen, "A chaotic image encryption algorithm based on 3-D bit-plane permutation," *Neural Comput. Appl.*, vol. 31, no. 11, pp. 7111–7130, Nov. 2019.
- [8] K. A. K. Patro and B. Acharya, "A novel multi-dimensional multiple image encryption technique," *Multimedia Tools Appl.*, vol. 79, nos. 19–20, pp. 12959–12994, May 2020.
- [9] C. Chen, K. Sun, and Q. Xu, "A color image encryption algorithm based on 2D-CIMM chaotic map," *China Commun.*, vol. 17, no. 5, pp. 12–20, May 2020.
- [10] X. Chai, X. Fu, Z. Gan, Y. Zhang, Y. Lu, and Y. Chen, "An efficient chaos-based image compression and encryption scheme using block compressive sensing and elementary cellular automata," *Neural Comput. Appl.*, vol. 32, no. 9, pp. 4961–4988, May 2020.
- [11] R. I. Abdelfatah, "A new fast double-chaotic based image encryption scheme," *Multimedia Tools Appl.*, vol. 79, nos. 1–2, pp. 1241–1259, Jan. 2020.
- [12] G. Ye and X. Huang, "An efficient symmetric image encryption algorithm based on an intertwining logistic map," *Neurocomputing*, vol. 251, pp. 45–53, Aug. 2017.
- [13] A. Belazi and A. A. A. El-Latif, "A simple yet efficient S-box method based on chaotic sine map," *Optik*, vol. 130, pp. 1438–1444, Feb. 2017.
- [14] Y. Zhou, L. Bao, and C. L. P. Chen, "A new 1D chaotic system for image encryption," *Signal Process.*, vol. 97, pp. 172–182, Apr. 2014.
- [15] K. A. K. Patro, A. Soni, P. K. Netam, and B. Acharya, "Multiple grayscale image encryption using cross-coupled chaotic maps," *J. Inf. Secur. Appl.*, vol. 52, Jun. 2020, Art. no. 102470.
- [16] X. Wang, L. Feng, and H. Zhao, "Fast image encryption algorithm based on parallel computing system," *Inf. Sci.*, vol. 486, pp. 340–358, Jun. 2019.
- [17] X. Wang and S. Gao, "Image encryption algorithm based on the matrix semi-tensor product with a compound secret key produced by a Boolean network," *Inf. Sci.*, vol. 539, pp. 195–214, Oct. 2020.
- [18] C. Chen, K. Sun, and S. He, "An improved image encryption algorithm with finite computing precision," *Signal Process.*, vol. 168, Mar. 2020, Art. no. 107340.
- [19] C. Zhu, Z. Gan, Y. Lu, and X. Chai, "An image encryption algorithm based on 3-D DNA level permutation and substitution scheme," *Multimedia Tools Appl.*, vol. 79, no. 11, pp. 7227–7258, 2020.
- [20] Q. Xu, K. Sun, C. Cao, and C. Zhu, "A fast image encryption algorithm based on compressive sensing and hyperchaotic map," *Opt. Lasers Eng.*, vol. 121, pp. 203–214, Oct. 2019.
- [21] R. M. May, "Simple mathematical models with very complicated dynamics," *Nature*, vol. 261, no. 5560, pp. 459–467, Jun. 1976.
- [22] N. Singh and A. Sinha, "Optical image encryption using hartley transform and logistic map," *Opt. Commun.*, vol. 282, no. 6, pp. 1104–1109, Mar. 2009.
- [23] X. Wang and S. Gao, "Image encryption algorithm for synchronously updating Boolean networks based on matrix semi-tensor product theory," *Inf. Sci.*, vol. 507, pp. 16–36, Jan. 2020.
- [24] M. A. B. Farah, A. Farah, and T. Farah, "An image encryption scheme based on a new hybrid chaotic map and optimized substitution box," *Nonlinear Dyn.*, vol. 99, no. 4, pp. 3041–3064, Mar. 2020.
- [25] M. Alawida, A. Samsudin, J. S. Teh, and R. S. Alkhalaf, "A new hybrid digital chaotic system with applications in image encryption," *Signal Process.*, vol. 160, pp. 45–58, Jul. 2019.
- [26] G. Ye and K.-W. Wong, "An efficient chaotic image encryption algorithm based on a generalized arnold map," *Nonlinear Dyn.*, vol. 69, no. 4, pp. 2079–2087, Sep. 2012.
- [27] Z. Hua, B. Xu, F. Jin, and H. Huang, "Image encryption using Josephus problem and filtering diffusion," *IEEE Access*, vol. 7, pp. 8660–8674, 2019.
- [28] K. A. K. Patro, B. Acharya, and V. Nath, "Secure, lossless, and noise-resistive image encryption using chaos, hyper-chaos, and DNA sequence operation," *IETE Tech. Rev.*, vol. 37, no. 3, pp. 223–245, May 2020.
- [29] J. Wu, X. Liao, and B. Yang, "Image encryption using 2D Hénon-sine map and DNA approach," *Signal Process.*, vol. 153, pp. 11–23, Dec. 2018.
- [30] Q. Lu, C. Zhu, and X. Deng, "An efficient image encryption scheme based on the LSS chaotic map and single S-Box," *IEEE Access*, vol. 8, pp. 25664–25678, 2020.
- [31] X. Chai, X. Zheng, Z. Gan, and Y. Chen, "Exploiting plaintext-related mechanism for secure color image encryption," *Neural Comput. Appl.*, vol. 32, no. 12, pp. 8065–8088, Jun. 2020.

- [32] J. Chen, F. Han, W. Qian, Y.-D. Yao, and Z.-L. Zhu, "Cryptanalysis and improvement in an image encryption scheme using combination of the 1D chaotic map," *Nonlinear Dyn.*, vol. 93, no. 4, pp. 2399–2413, Sep. 2018.
- [33] Y. Zhang, "The image encryption algorithm based on chaos and DNA computing," *Multimedia Tools Appl.*, vol. 77, no. 16, pp. 21589–21615, Aug. 2018.
- [34] C. Lakshmi, K. Thenmozhi, J. B. B. Rayappan, and R. Amirtharajan, "Hopfield attractor-trusted neural network: An attack-resistant image encryption," *Neural Comput. Appl.*, vol. 32, no. 15, pp. 11477–11489, Aug. 2020.
- [35] X. Chai, "An image encryption algorithm based on bit level brownian motion and new chaotic systems," *Multimedia Tools Appl.*, vol. 76, no. 1, pp. 1159–1175, Jan. 2017.
- [36] A. Alghafis, F. Firdousi, M. Khan, S. I. Batool, and M. Amin, "An efficient image encryption scheme based on chaotic and deoxyribonucleic acid sequencing," *Math. Comput. Simul.*, vol. 177, pp. 441–466, Nov. 2020.
- [37] X. Wang, H. Zhao, and M. Wang, "A new image encryption algorithm with nonlinear-diffusion based on multiple coupled map lattices," *Opt. Laser Technol.*, vol. 115, pp. 42–57, Jul. 2019.
- [38] S. Zhu and C. Zhu, "Plaintext-related image encryption algorithm based on block structure and five-dimensional chaotic map," *IEEE Access*, vol. 7, pp. 147106–147118, 2019.
- [39] J. Khan, J. Ahmad, S. Ahmed, H. Siddiq, S. Abbasi, and S. Kayhan, "DNA key based visual chaotic image encryption," *J. Intell. Fuzzy Syst.*, vol. 37, no. 2, pp. 2549–2561, 2019.
- [40] Y. Wu, J. P. Noonan, and S. Aгаian, "NPCR and UACI randomness tests for image encryption," *Cyber. J. Multidisciplinary, J. Sci. Technol., J. Sel. Areas Telecommun.*, vol. 1, no. 2, pp. 31–38, 2011.
- [41] C. Cao, K. Sun, and W. Liu, "A novel bit-level image encryption algorithm based on 2D-LICM hyperchaotic map," *Signal Process.*, vol. 143, pp. 122–133, Feb. 2018.
- [42] X. Wang and J. Yang, "A novel image encryption scheme of dynamic S-boxes and random blocks based on spatiotemporal chaotic system," *Optik*, vol. 217, Sep. 2020, Art. no. 164884.
- [43] Y. Wu, Y. Zhou, J. P. Noonan, and S. Aгаian, "Design of image cipher using Latin squares," *Inf. Sci.*, vol. 264, pp. 317–339, Apr. 2014.
- [44] Z. Hua and Y. Zhou, "Image encryption using 2D logistic-adjusted-sine map," *Inf. Sci.*, vol. 339, pp. 237–253, Apr. 2016.
- [45] X. Li, Z. Xie, J. Wu, and T. Li, "Image encryption based on dynamic filtering and bit cuboid operations," *Complexity*, vol. 2019, pp. 1–16, Feb. 2019.
- [46] X.-L. Chai, Z.-H. Gan, Y. Lu, M.-H. Zhang, and Y.-R. Chen, "A novel color image encryption algorithm based on genetic recombination and the four-dimensional memristive hyperchaotic system," *Chin. Phys. B*, vol. 25, no. 10, Oct. 2016, Art. no. 100503.



control, image processing, chaos cryptography, systems biology, and complex networks.



XINGYUAN WANG received the Ph.D. degree in computer software and theory from Northeastern University, China, in 1999. From 1999 to 2001, he was a Postdoctoral Researcher with Northeastern University. He is currently a Professor of information science and technology with Dalian Maritime University, China. He has published three books and more than 400 scientific articles in refereed journals and proceedings. His research interests include nonlinear dynamics and

PENGBO LIU received the bachelor's degree from the College of Computer Science and Technology, Henan University of Technology, China. He is currently pursuing the master's degree in information science and technology with Dalian Maritime University, China. His main research interests include chaotic encryption and image processing.

...