Multimedia Tools and Applications (2023) 82:22231–22250
https://doi.org/10.1007/s11042-022-13461-3

1187: RECENT ADVANCES IN MULTIMEDIA INFORMATION SECURITY: CRYPTOGRAPHY AND STEGANOGRAPHY

The unified image encryption algorithm based on composite chaotic system

Jiming Zheng [1,2] · Qingxia Zeng [1]

Received: 21 November 2020 / Revised: 20 September 2021 / Accepted: 7 October 2021 /
Published online: 22 July 2022

Abstract
This paper proposes a fast and unified encryption and decryption algorithm based on a composite chaotic system. By combining Logistic map and Sine map, the New-Logistic-Sine map (NLS map) is obtained. NLS map generated the diffusion key matrix needed in the algorithm process, which can enhance the anti-attack ability of the encryption algorithm. Different from most image cryptography systems, the algorithm adopted in this paper has the same encryption process and decryption process, which can save half of the resources in real applications. Firstly, the Secure Hash Algorithm 256 (SHA256) value of the original image was obtained, and the initial values and control parameters of NLS map and Logistic map were calculated; Secondly, the diffusion key matrix is obtained by iterative the NLS map, and is used to perform the first diffusion of the original image; Thirdly, the permutation key sequence is obtained by iterative the Logistic map, and using the sequence to perform the permutation operation on the image after the first diffusion; Finally, the same diffusion key matrix as the first diffusion operation is used to carry out the second diffusion operation on the displaced image to obtain the final encrypted image. The simulation experiment and security analysis show that the proposed image cryptosystem possessed identical encryption process and decryption process, and the algorithm speed is improved ensure the security of the algorithm.

Keywords Chaos map · Unity encryption · SHA256 · Permutation operation · Diffusion operation

✉ Jiming Zheng
Zhengjm@cqupt.edu.cn

[1] College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

[2] Key Laboratory of Intelligent Analysis and Decision Complex Systems, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

1 Introduction

With the development of science and technology, there are more and more electronic products in our life, and the use of smart-phones and computers is increasing day by day. A large amount of user data information is generated and transmitted on the network. Compared with other data formats, the information carried by digital images is transmitted to users through visualization, which is more intuitive and visual, and has become a widely used data format. At the same time, because digital images contain a large amount of predictable information, the security and privacy of image information has attracted much attention. In order to ensure the safe transmission of image information in the network without leakage, it is necessary to encrypt the image and then transmit it. Common traditional encryption methods include Data Encryption Standard (DES), RSA and Advanced Encryption Standard (AES), which are often used for text data encryption. Compared with text data format, image data has large redundancy and high correlation between pixels. Traditional encryption methods, with too long execution time, are not suitable for image encryption [6, 16, 29]. For data security, a text encryption algorithm is proposed using matrix properties [13]. The algorithm converts 26 letters into 1–26 in sequence, and then converts the converted number sequence into a matrix, and inverts the matrix to obtain the encrypted text. If the algorithm is applied to the image, it is the inverse of a large matrix, with high computational complexity and general confusion effect. At present, some new methods are tried to apply to digital image encryption: cellular automata [17, 21], wavelet transform [20], compressed sensing [4], DNA coding [5, 10, 30], neural network [31] and chaotic map [1, 11, 27], watermarking [25], etc.

Due to the randomness, ergodicity, and sensitivity to initial values and control parameters of the chaotic system, the encryption algorithm of the chaotic system can reduce the number of encryption rounds and the complexity of the algorithm while achieving security standards. Compared with other methods, chaotic-based encryption technology is more superior [9]. Zhou et al. used the Chen map of hyper-chaotic system in [19]. The map has four control parameters and four initial values, which greatly increases the key space and makes the algorithm effective against exhaustive attacks. In [7], Cao et al. used a new two-dimensional chaotic system, which is based on the cascade modulation coupling model to achieve the combination of two one-dimensional chaos. The system has good randomness and ergodicity, and the parameter range is larger than two one-dimensional chaotic maps, which is more suitable for image encryption. Taneja et al. [24] proposed a chaotic cryptosystem based on spatial domain. The purpose of this paper is to replace any chaotic system in the displacement and diffusion stage, so as to achieve the savings of resource calculation. The paper takes Cat map and Henon map as examples, and the experimental results show that this algorithm can achieve its goal well. The high-dimensional chaotic system has a large key space and high security, but the computational complexity is high, which takes longer than a one-dimensional chaotic system.

Zhu et al. [32] used Cat map to calculate the new coordinates of each pixel, and completed the permutation process according to the new coordinates. Here, the two control parameters of Cat map are calculated by Logistic map, in order to increase encryption security. Each pair of coordinates requires multiple calculations to get the final result, and the execution time increases significantly when the performance is slightly improved. Zhang et al. proposed non-adjacent coupled map lattices (NCML) in [26]. Compared with Logistic map and coupled map, NCML has more prominent cryptographic characteristics. The generated sequence is the diffusion key can achieve the security of the algorithm very well. Joshua et al. [15] used the enhanced version of logistic map to generate chaotic key. Considering the reasons of

discontinuous chaotic range, uneven distribution, small key space and chaotic period of logistic map and other one-dimensional maps, the enhanced logistic map can achieve better security of the encryption system. One dimensional chaotic systems have great advantages over high-dimensional chaotic systems in terms of computational complexity. However, the one-dimensional chaotic map has a small key space and low security. Considering these shortcomings, there are currently existing improved methods: cascaded chaotic systems, pseudo-random disturbances, switching systems, and combined systems based on modular arithmetic.

According to Shannon's description, a secure encryption system can be constructed by using the permutation-diffusion structure [18]. Dai et al. only used a round of diffusion operation to complete the encryption process in [8]. The diffusion operation is to XOR the original image pixels with the values generated by the Logistic map and Chebyshev map iteration. After all the pixels of the image are traversed, the encryption is completed. Hua et al. used the two permutation-diffusion process in [12]. This method is efficient and robust to some impulsive noise and data loss. Because this method is a bit-level operation, twice permutation-diffusion process will greatly increase the computational complexity. In [2], Alawida et al. first divide the image up and down, obtain the permutation order according to the chaotic map, and then the upper and lower pixels interact with each other to obtain the diffusion key matrix to realize the diffusion process, complete the first permutation-diffusion process, and then divide the left and right to complete the second permutation-diffusion process. Taneja et al. [22] proposed a different encryption method for different regions and carried out A total of two rounds of displacement - XOR diffusion forward and reverse diffusion. The image matrix is divided into blocks, according to the edge detection results to determine whether each block is an important block, important and unimportant blocks are encrypted in two ways, the algorithm well ensures the security at the same time, improve the encryption efficiency. Kohli et al. [14] proposed a multiple encryption technique that optimizes memory/area chip, operational usage, encryption time, etc.

For all cryptographic systems, chaos system is very important for developing encryption algorithms with good randomness. Meanwhile, considering how to realize the permutation and diffusion process and ensure that the algorithm can resist all kinds of attacks is also the key of encryption algorithm. In this paper, a composite chaotic system based on Logistic map and Sine Map is presented. The good performance of the chaotic system can be known by computing Lyapunov index and National Institute of Standards and Technology (NIST) test. On this basis, a new permutation algorithm is proposed, which can be used in diffusion-permutation-diffusion structure to achieve the same encryption process and decryption process. The remainder of this paper is as follows: The new composite chaotic system is presented in Section 2. Section 3 introduces the algorithm process, Section 4 is the simulation experiment, and Section 5 is the summary.

2 Chaotic system

2.1 New-logistic-sine map

In chaotic maps, Logistic, Tent, Sine, and so on are traditional one-dimensional maps. A chaotic system with good pseudo-random sequence and large parameter in image encryption can be obtained by combining them properly. Alawida et al. proposed a new structure of chaotic map [2], as shown in Fig. 1.

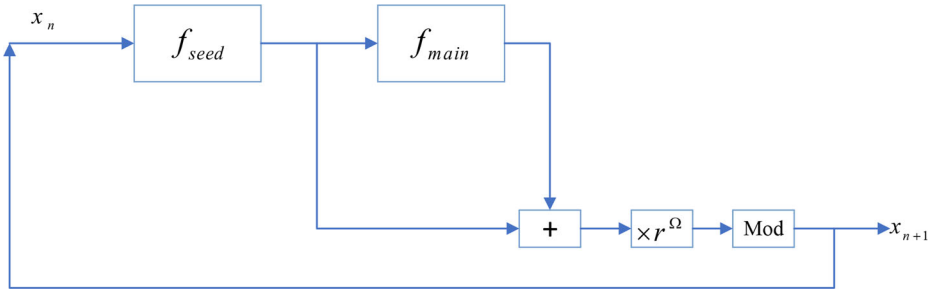


Fig. 1 The structure diagram of chaotic map proposed in [2]

where +, × represents the addition operation and multiplication operation between floating-point numbers, *Mod* is the operation of taking modulus of 1, and the value of Ω determines the calculation time and chaos.

According to Fig. 1, the chaotic map can be represented as,

$$x_{n+1} = \left(\left(f_{main}\left(f_{seed}(x_n)\right) + f_{seed}(x_n)\right)r^{\Omega}\right)\mathrm{mod}1 \tag{1}$$

where f_{main} and f_{seed} are traditional one-dimensional chaotic maps.

Logistic map and Sine map can be represented as Eqs. (2) and (3).

$$\text{Logistic map} : x_{n+1} = r_0 x_n(1-x_n), r_0 \in (3.55, 4] \tag{2}$$

$$\text{Sine map} : x_{n+1} = r_1 \sin(\pi x_n), r_1 \in [0, 1] \tag{3}$$

It can be seen from Fig. 2 that the control parameter range of Logistic map($r_0 \in (3.569945627,\ 4]$) is very small, and when r_0 approaches 4, Logistic map's Lyapunov exponent is positive. The parameter range of Sine map is discontinuous, when r_1 approaches 1, Sine map's Lyapunov exponent is positive [9]. So, in this paper, taking Logistic map as f_{main} and Sine map as f_{seed}, we can overcome the shortcomings of two one-dimensional maps and obtain an improved New-Logistic-Sine map(NLS map).

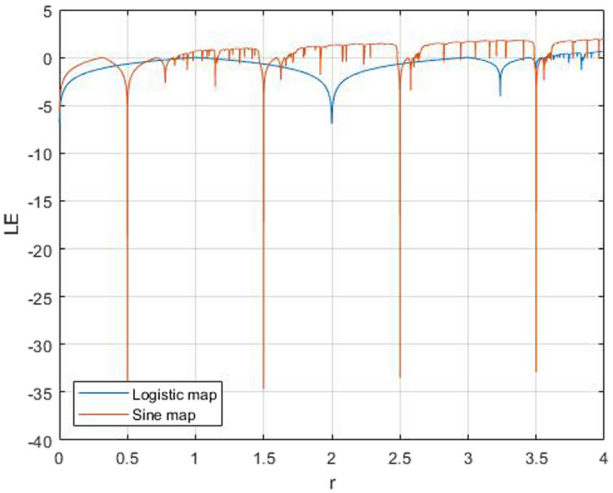


Fig. 2 Lyapunov exponent of Logistic map and Sine map

The NLS map as follows,

$$x_{n+1} = \left(\left(\frac{r^2}{4}\sin(\pi x_n) \times \left(1-\frac{r}{4}\sin(\pi x_n)\right) + \frac{r}{4}\sin(\pi x_n)\right) r^{11}\right) \bmod 1 \tag{4}$$

where $x_{n+1} = ((f_{main}(f_{seed}(x_n)) + f_{seed}(x_n))r^{\Omega}) \bmod 1, f_{\boldsymbol{main}} = rx(1-x), f_{\boldsymbol{seed}} = \frac{r}{4}sin(\pi x), r \in [0, 6]$. Considering the chaotic characteristics and the computation time, set $\Omega = 11$.

Compared with Tent-Logistic-Tent (TLT) map and Tent-Sine-Tent (TST) map [2], the control parameters of the composite chaotic map have a larger chaotic interval. In this interval, NLS map has larger Lyapunov exponent and complex chaotic behavior. It can increase the key space and improve the performance of encryption system when applied to encryption system.

2.2 Performance analysis of NLS map

In order to show the performance of NLS map, it will be compared and analyzed with TLT map and TST map from three aspects, namely bifurcation diagram, chaotic trajectory and Lyapunov index. And the randomness of sequences generated by NLS maps is proved by NIST test.

2.2.1 Bifurcation diagram

Bifurcation refers to the sudden change of topological structure due to the change of control parameters in dynamic system. After continuous bifurcation, the final state is chaos.

Figure 3 shows the bifurcation diagram of TLT map, TST map and NLS map at initial value $x_0 = 0.216$. Abscissa is the control parameter. Fig. 3a shows the bifurcation diagram of TLT map, which stops bifurcation after $r_2 = 1.05$ and reaches chaotic state, and the value range of r_2 is [1.05, 4]; Fig. 3b is the bifurcation diagram of TST map, which stops bifurcation after $r_3 = 1.05$ and reaches chaotic state, and the value range of r_3 is [1.05, 4]; Fig. 3c is the bifurcation diagram of NLS map, which stops bifurcation after $r = 1.509$ and reaches chaotic state, the value range of r is [1.509, 6]. It can be seen from Fig. 3 that the control parameter range of the proposed NLS map is larger, and the key space required for the encryption algorithm is large, which can well resist exhaustive attacks.

2.2.2 Chaotic trajectory

The trajectory of periodic motion is a closed curve, and the trajectory of chaotic motion will never be closed or repeated theoretically. Therefore, chaotic trajectories usually occupy a certain phase space and can reflect the randomness of the output of chaotic systems. If the chaotic trajectory of a chaotic system occupies a large phase space, it has a good random output.

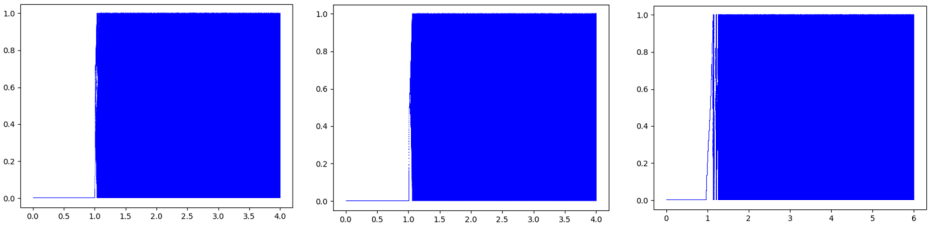


Fig. 3 Bifurcation diagram

The control parameters of TLT map, TST map and NLS map are set as $r_3 = r_4 = r = 2.4111$, and the initial value is $x_0 = 0.216$. In order to show the actual behavior of chaotic system in stable state, we draw the iteration points from 0 to 2000 in each trajectory. It can be seen from Fig. 4 that the NLS map trajectories occupy almost all the phase planes.

2.2.3 Lyapunov exponents

The sensitivity of initial state is the most obvious characteristic of chaotic behavior. Lyapunov exponents(LE) [3] can quantitatively describe the initial state sensitivity. For two trajectories of a chaotic system starting from two close initial states, LE describes their average separation rates. For the differentiable one-dimensional dynamic system $x_{i+1} = f(x_i)$, $i = 0, 1, 2, \ldots$, the LE for 1D maps are given by

$$\lambda = \lim_{N \to +\infty} \frac{1}{N} \sum_{n=1}^{N} \log \left| \frac{dx_{n+1}}{dx_n} \right| \tag{5}$$

A positive LE index indicates that the closed trajectory of the dynamic system diverges in unit time and evolves into a completely different trajectory with the increase of time. Therefore, if the LE index of a dynamic system is positive, then the system is chaotic. And a higher value means better performance.

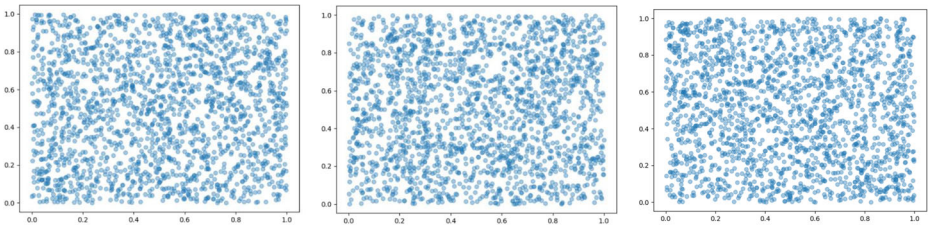
As shown in Fig. 5a, when $r \in [1.509, 6]$ occurs, NLS map is in chaotic state, and Lyapunov exponent is close to 24. The Lyapunov exponents of TLT map and TST map are described in Fig. 5b. When $r \in [1.05, 4]$ occurs, TLT map and TST map are in chaotic state, and the Lyapunov exponent is close to 17. The chaotic interval of NLS map is greater than TLT map and TST map, meanwhile the value of LE is bigger, so NLS maps have better performance.

2.2.4 NIST test

In this paper, the pseudo-randomness of sequences generated by NLS Map is verified by NIST test.

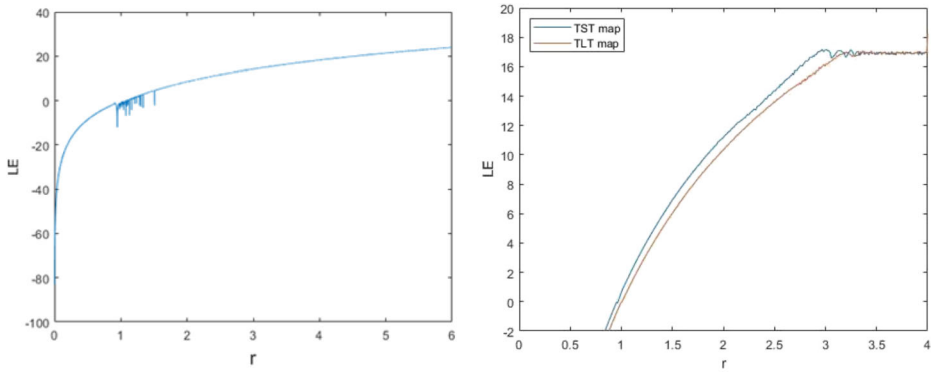
In order to obtain more accurate experimental results, we determined the sequence length of the test as $n = 100000$. In this paper, according to the chaotic trajectory, the initial value and parameter of NLS map are $x_0 = 0.216$, $r = 2.4111$. We iterate Eq. (4) for n times to get a sequence $X = \{x_1, x_2, \cdots, x_n\}$ of length n, and then convert it to the binary sequence $\widehat{X} = \{\widehat{x}_1, \widehat{x}_2, \cdots, \widehat{x}_n\}$, where $\widehat{x}_i$ is defined as follows

$$\widehat{x}_i = \begin{cases} 0, 0 \le x_i < \delta \\ 1, \delta \le x_i < 1 \end{cases} \tag{6}$$



(a) (b) (c)

Fig. 4 Trajectories



(a) (b)

Fig. 5 Lyapunov exponents

and $\delta = \frac{1}{n}\sum_{i=0}^{n} x_i$ is the average value of the test sequence X. We get a n bit long binary sequence finally. It is divided into 10 groups, each of which is 10000bit long. The NIST test consists of 15 test elements, as shown in the first column of Table 1. The third column of Table 1 shows the average of the 10 groups of P values in this test. The test results showed that all P values were greater than 0.01, and all 15 tests passed. Therefore, it shows that the chaotic sequences generated by the composite chaotic system are random.

3 Analysis and design of encryption and decryption algorithm

The encryption and decryption algorithms proposed in this paper are consistent. The algorithm includes two diffusion processes and one permutation process. After inputting the initial key and the original image, the encryption key is calculated, and the encryption image is obtained through the algorithm; then, if the encryption image and encryption key are used as the input, the original image will be obtained by running the same algorithm.

Table 1 NIST test results

Number	Test content	P value	Result
1	monobit_test	0.22246487489566913	PASS
2	frequency_within_block_test	0.9649634373412823	PASS
3	runs_test	0.9972166610004033	PASS
4	longest_run_ones_in_a_block_test	0.34472460277343814	PASS
5	binary_matrix_rank_test	0.8099416821956189	PASS
6	dft_test	0.408862725686047	PASS
7	non_overlapping_template_matching_test	0.9999533067167871	PASS
8	overlapping_template_matching_test	0.998345629033456	PASS
9	maurers_universal_test	0.9996972267981994	PASS
10	linear_complexity_test	0.031198930079309378	PASS
11	serial_test	0.5505265073241046	PASS
12	approximate_entropy_test	0.6421515073154084	PASS
13	cumulative_sums_test	0.26618100544012857	PASS
14	random_excursion_test	0.12514191294948793	PASS
15	random_excursion_variant_test	0.013415871240881371	PASS

Assumes that the size of the original image P is $M \times N$, M is the length of the image and N is the width of the image.

3.1 Initial value and parameter generation of chaotic system

The key sequence and key matrix needed in the process of encryption are obtained by iterative chaotic map. In order to improve the sensitivity of the key and the anti-differential attack ability of the algorithm, the chaotic sequence initial values and control parameters are related to the original image. The method is to use SHA256 function to get the hash value of the original image, and then generate the final initial value (x_0) and control parameter (r) of NLS map and the initial value (y_0) and control parameter (r_0) of Logistic map respectively through the initial keys $\dot{x_0}$, $\dot{r}$, $\dot{y_0}$, $\dot{r_0}$.

Step 1: Through the hash function on the original image, getting a 64 bit hexadecimal string H,

$$H = h_1, h_2, \ldots, h_{64} \tag{7}$$

Step 2: Convert each character of h_j (j = 1, 2, …, 64) into a 4-digit binary system, a 64 binary string T,

$$T = t_1, t_2, \ldots, t_{64} \tag{8}$$

Step 3: According to Eq. (9), divide the string T into 4 parts to calculate $K1$, $K2$, $K3$ and $K4$ respectively,

$$Kl = \sum_{i=n}^{m}(T[i]) \tag{9}$$

where l = 1, 2, 3, 4, for each part, $n = (l-1) * ({}^{64}/_{4})$, $m = n + {}^{64}/_{4}$.

Step 4: With $K1$, $K2$, $K3$, $K4$ and the initial keys ($\dot{x_0}$, $\dot{r}$, $\dot{y_0}$, $\dot{r_0}$). x_0, r, y_0 and r_0 are generated for the first time.

$$x_0 = \left(\dot{x_0} + K1 + K2\right) mod 1 \tag{10}$$

$$y_0 = \left(\dot{y_0} + K3 + K4\right) mod 1 \tag{11}$$

$$r = \left(\dot{r} + K2 + K4\right) mod 4.491 + 1.509 \tag{12}$$

$$r_0 = \left(\dot{r_0} + K1 + K3\right) mod 0.43 + 3.5699 \tag{13}$$

Step 5: Using the initial keys ($\dot{x_0}$, $\dot{r}$, $\dot{y_0}$, $\dot{r_0}$) and the x_0, r, y_0, r_0 generated in Step 4, the initial values and control parameters required for the final chaotic map are obtained by Eqs. (14–17).

$$x_0 = \left(\dot{x_0} + x_0 + y_0 + r + r_0\right) mod 1 \tag{14}$$

$$y_0 = \left(\dot{y_0} + x_0 + y_0 + r + r_0\right) mod 1 \tag{15}$$

$$r = \left(\dot{r} + x_0 + y_0 + r + r_0\right) mod 4.491 + 1.509 \tag{16}$$

$$r_0 = \left(\dot{r_0} + x_0 + y_0 + r + r_0\right) mod 0.43 + 3.5699 \tag{17}$$

3.2 Encryption and decryption algorithm

The control parameters and initial values ($\dot{x_0}$, $\dot{r}$, $\dot{y_0}$, $\dot{r_0}$) of NLS map and Logistic map are obtained by using the method in Section 3.1. The permutation key sequence and the diffusion key matrix are generated by iterating the two chaotic maps several times. The diffusion key matrix is used to realize the first diffusion process of P to obtain $C1$; Next, the permutation key sequence is used to confuse $C1$ to obtain $C2$; Finally, $C2$ is processed through the second diffusion process to obtain the final image $C3$. The flow chart of the algorithm is shown in Fig. 6.

Using this algorithm, the process of encryption and decryption can be consistent. A detailed diffusion -permutation - diffusion algorithm is given below.

Diffusion algorithm 1:

Step 1: Substitute the x_0, r into the NLS map(Eq. (4)) for 1000 + M*N iterations. In order to obtain the steady-state data, we take the data generated by 1001 iterations to 1000 + M*N iterations as valid data to generate sequences ***List***.Substitute the x_0, r into the NLS map(Eq. (4)) for 1000 + M*N iterations. In order to obtain the steady-state data, we take the data generated by 1001 iterations to 1000 + M*N iterations as valid data to generate sequences ***List***.

Step 2: According to Eq. (18), all values of the sequence ***List*** are converted into integer values between 0 and 255 to obtain the E sequence,

$$E[i] = \left(\boldsymbol{List}[i] * 10^4\right) mod 256 \tag{18}$$

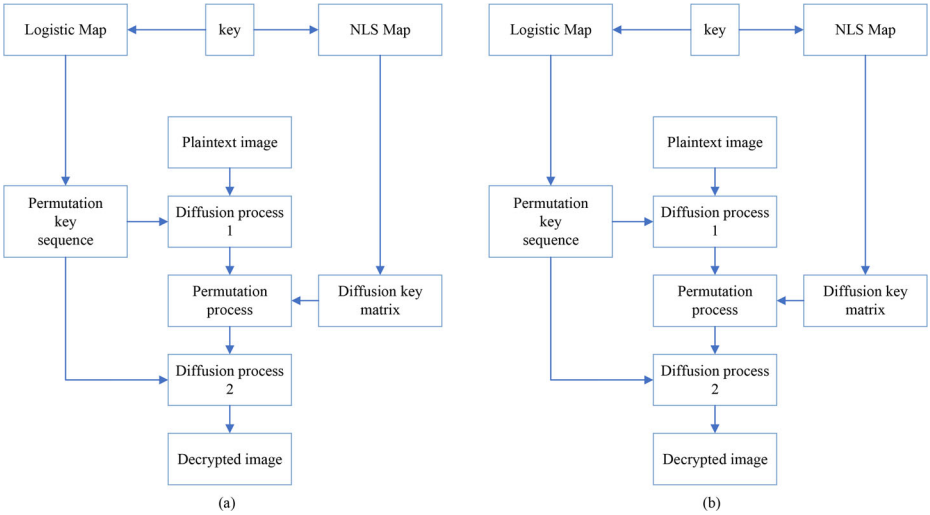


Fig. 6 Proposed image cryptosystem

Step 3: Transform the E sequence into the key diffusion matrix NE of $M \times N$,

$$NE = reshape(E, M, N) \tag{19}$$

Step 4: Add NE into Eq. (20) and get $C1$ from the original image $P(M \times N)$,

$$C1[i,j] = P[i,j] \oplus NE[i,j] \tag{20}$$

where $i = 0, 1, \ldots, M, j = 0, 1, \ldots, N$.

Permutation algorithm:

Step 1: Transform $C1$ into the sequence h_C1 with length of M^*N, and calculate the segmentation position L of the upper and lower parts,

$$L = math.ceil(len(h\_C1)/2) \tag{21}$$

Step 2: Use the initial value y_0 and parameter r_0, iterate Logistic map(Eq. (2)) for $2000 + M^*N - L$ times, also remove the data of the first 2000 times, take the data from 2001 to $2000 + M^*N - L$ times as valid data, and get the sequence X with length of $M^*N - L$.

Step 3: Obtain the index sequence $X1$ of sequence X from small to large, and the range of the index sequence $X \in [0, M^*N - L - 1]$,

$$X1 = np.\mathbf{argsort}(X) \tag{22}$$

Step 4: Transform the size range of $X1$ into $X \in [L, M^*N - 1]$ to obtain the sequence $X2$,

$$X2[i] = M^*N - L + X1[i] \tag{23}$$

where, $i = 0, 1, \ldots, M^*N - L - 1$.

Step 5: Traverse i, swap the front and back of h_C1 according to the sequence $X2$,

$$\begin{aligned} c\_C1 &= h\_C1[i] \\ h\_C1[i] &= h\_C1[X2[i]] \\ h\_C1[X2[i]] &= c\_C1 \end{aligned} \tag{24}$$

where c_C1 is the intermediate value.

Step 6: Reshape the sequence h_C1 into the matrix $C2$ of $M \times N$,

$$C2 = \mathbf{reshape}(h\_C\mathbf{1}, M, N) \tag{25}$$

$C2$ is the permutation result.

Diffusion algorithm 2:

According to Eq. (26), the permutation result $C2$ and the diffusion key matrix are XOR operated to obtain the final encrypted image $C3$

$$C3[i,j] = C2[i,j] \oplus NE[i,j] \tag{26}$$

where $i = 0, 1, \ldots, N j = 0, 1, \ldots, N$.

4 Simulation results and security analysis

4.1 Simulation results

This section gives the simulation results of the proposed scheme. A large number of experiments were carried out based on four grayscale images (from the USC-SIPI image database and computer vision test images [3]), ordinary images of Lena.jpg (256 × 256), Peppers.jpg (256 × 256), Baboon (256 × 256), Boat (256 × 256). Select $\dot{x_0} = 0.4478$, $\dot{y_0} = 0.7643$, $\dot{r} = 2.8113$, $\dot{r_0} = 3.5699$ as the initial keys. The encryption results are shown in the Fig. 7.

In the case of meeting safety requirements, operating speed becomes an important factor in practical applications. In order to evaluate the calculation speed, the Lena image, Peppers image, Baboon image, Boat image are encrypted 10 times, and the average execution time is shown in Table 2. It's obvious from the Table 2 that the encryption speed of the method proposed of our scheme is better than Ref. [5], Ref. [2], and Ref. [28].

4.2 Security analysis

4.2.1 Histogram analysis

Histogram is used to describe the distribution quantity of all pixel values in an image. The more uniform the histogram distribution, the lower the readability and the higher the security of the encrypted image. Figure 8 shows the histogram of original images and the histogram for the corresponding encrypted image.

It can be seen from Fig. 8 that the histogram distribution after the original image density is uniform. In addition, three criteria are used to reflect the uniform consistency of the histogram, namely, maximum deviation(M_D), irregular deviation(I_D) and deviation from uniform histogram(DUH) [2]. They are defined respectively as follows,

$$M_D = \frac{D_0 + D_{N+1}}{2} + \sum_{i=1}^{N-1} D_i \tag{27}$$

$$I_D = \sum_{i=0}^{N-1} |h_i - A_H| \tag{28}$$

$$DUH = \frac{\sum_{C_i=0}^{255} |H_{C_i} - H_C|}{M \times N} \tag{29}$$

where $D_i = |D_{C_i} - D_{P_i}|$ is the absolute value of the difference between the two pixels corresponding to the same position of the encrypted image and the original image; N is the number of grey values with the range of [0,256]; h_i is the difference between the encrypted image and the original image in the histogram under the same index i; A_H is the average of the histogram; H_{C_i} is the value corresponding to histogram index, $H_C = \frac{M \times N}{256}$represents the average grayscale value of each pixel.

A large M_D indicates that the pixel value of the encrypted image has been greatly changed, which means that the encryption scheme is secure; A smaller I_D indicates that the pixel values of encrypted images are distributed evenly, which is a very important feature of encrypted

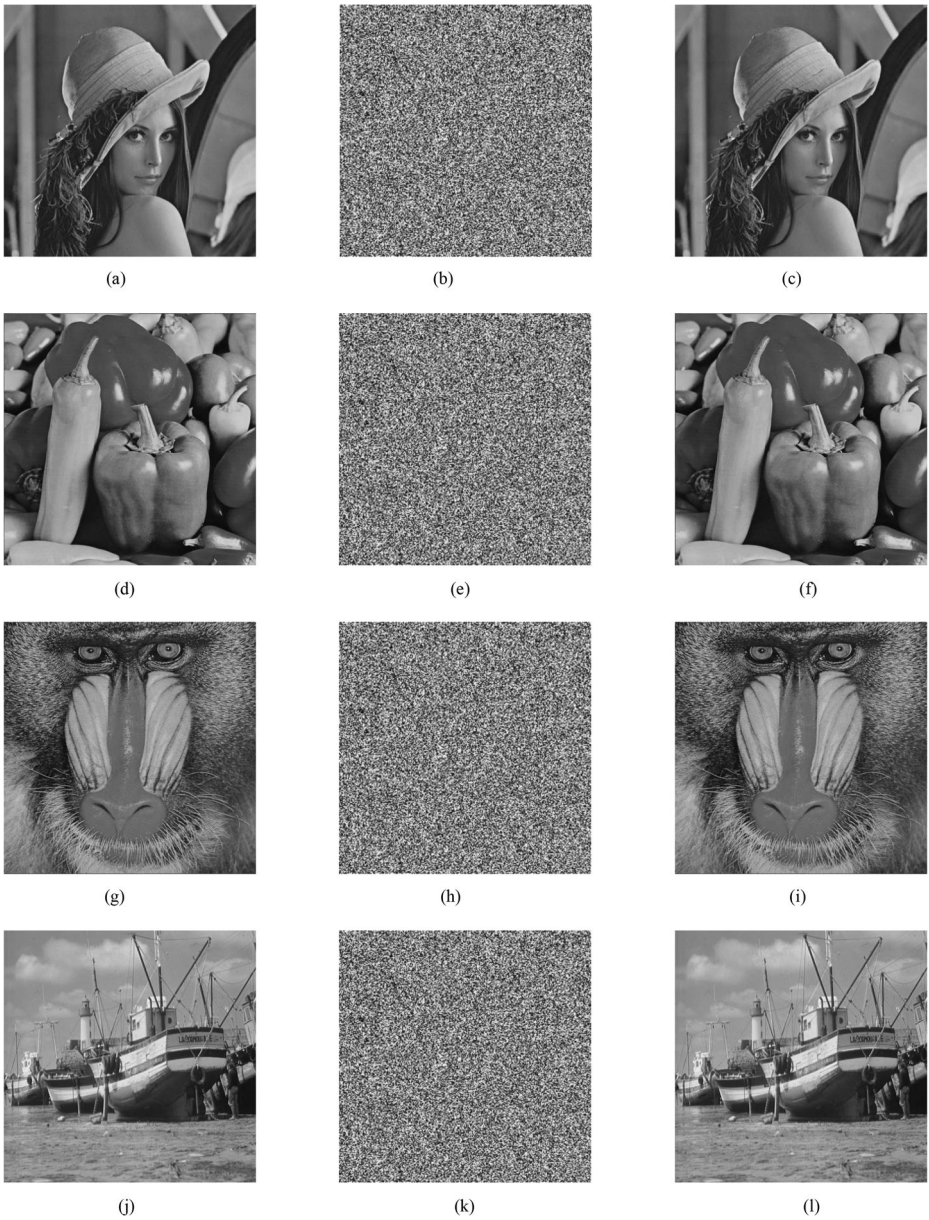


Fig. 7 Simulation results

images; The lower the DUH is, the closer the histogram is to the ideal state, and the distribution is uniform.

From the Table 3, according to the average data of the four graphs, the effect of the text in histogram data is comparable to Ref. [28] and better than Ref. [2], Ref. [5].

Table 2 Algorithm execution time

Image/Time(s)	Lena	Peppers	Baboon	Boat
Our scheme	2.8206	2.7976	1.8579	2.2517
Ref. [28]	11.0436	11.3264	9.1277	9.6584
Ref. [2]	3.5249	3.6609	2.9547	4.5114
Ref. [5]	9.5682	8.4198	9.0797	9.0608

4.2.2 Correlation coefficient analysis

For any given image, each pixel has a high correlation with its adjacent pixels in the horizontal, vertical, and diagonal directions. A good encryption algorithm can break the correlation of adjacent pixels in an image. In order to measure the correlation between adjacent pixels(x, y), it is necessary to select the adjacent pixels from the image and then calculate their correlation coefficient r_{xy}. The pixel distribution of plaintext image and encrypted image is shown in Fig. 9.

$$r_{xy} = \frac{\mathrm{cov}(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{30}$$

$$cov(x,y) = E\{(y-E(y))(x-E(y))\} \tag{31}$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i, D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i-E(x))^2 \tag{32}$$

The closer r_{xy} is to 0, the better. The average correlation of 1000 times was calculated by comparing three references.

As can be seen from the data in Table 4, the correlation between the encryption results obtained by the algorithm in this paper is all less than 0.03, effectively reducing the high

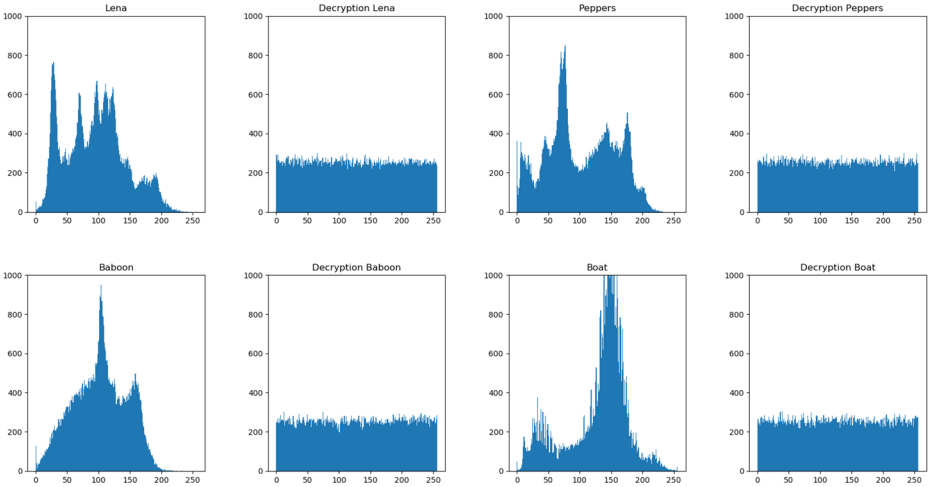


Fig. 8 Histograms of the original and encrypted images

Table 3 Histogram analysis and Comparison

Image		M_D	I_D	DUH
Lena	Our scheme	44,623.0	28,866.0	0.0491
	Ref. [28]	45,091.0	28,956.0	0.0469
	Ref. [2]	44,739.0	29,022.0	0.0494
	Ref. [5]	45,504.0	28,692.0	0.0574
Peppers	Our scheme	36,135.0	36,560.0	0.05070
	Ref. [28]	36,291.0	36,574.0	0.05075
	Ref. [2]	35,745.5	36,928.0	0.04779
	Ref. [5]	36,579.0	36,526.0	0.0629
Baboon	Our scheme	48,756	25,618	0.0509
	Ref. [28]	48,136	25,670	0.0477
	Ref. [2]	47,965	25,464	0.0513
	Ref. [5]	48,010	25,680	0.0715
Boat	Our scheme	55,224	40,556	0.0502
	Ref. [28]	55,335.5	40,912	0.0515
	Ref. [2]	55,255.5	40,668	0.0517
	Ref. [5]	55,088	40,922	0.0795

correlation between pixels of the original image [5]. According Table 4, the algorithm in this paper is better than Ref. [2] and Ref. [5], and is equivalent to Ref. [28].

4.2.3 Information entropy

Information entropy is another standard to evaluate the distribution of image gray value, which reflects the uncertainty of image information. The information entropy of sequence s can be expressed as,

$$H(s) = \sum_{i=0}^{L-1} P(s_i) \log_2 \frac{1}{P(s_i)} \tag{33}$$

where $L = 2^M$, and M is the total number of samples (for images with gray level of 8, M is 8). s_i is the gray value of the image, $P(s_i)$ stands for the probability of the element s_i in the

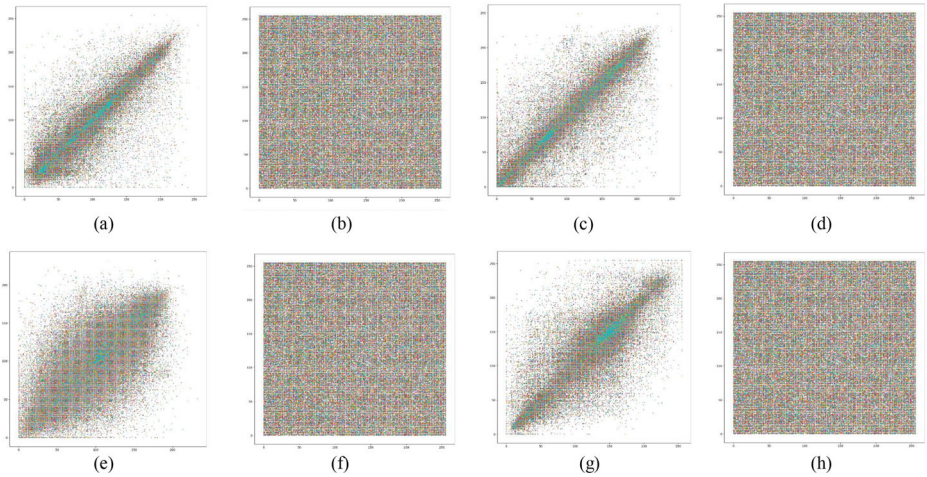


Fig. 9 The distribution of two adjacent pixels in the original image and the cipher image

Table 4 Correlation coefficients

Image		Horizontal	Vertical	Diagonal
Lena	Our scheme	−0.0045	0.0024	0.0019
	Ref. [28]	−0.0014	0.0034	−0.0066
	Ref. [2]	−0.0042	−0.0074	0.0043
	Ref. [5]	−0.0028	0.0066	−0.0023
Peppers	Our scheme	−0.0004	−0.0031	−0.0031
	Ref. [28]	−0.0024	−0.0034	0.0046
	Ref. [2]	−0.0022	−0.0049	0.0049
	Ref. [5]	−0.0097	−0.0035	0.0002
Baboon	Our scheme	−0.0095	0.0028	0.0064
	Ref. [28]	−0.0048	0.0006	0.0028
	Ref. [2]	0.0054	−0.0042	−0.0011
	Ref. [5]	−0.0056	0.0026	0.0112
Boat	Our scheme	−0.0065	0.0028	0.0075
	Ref. [28]	−0.0015	0.0012	0.0014
	Ref. [2]	0.0013	0.0092	0.0458
	Ref. [5]	−0.0012	0.0059	0.0003

sequence s, and $\sum_{i=0}^{L-1} P(s_i) = 1$. The value of information entropy is close to 8, which indicates that the encryption algorithm makes the image more random and can resist certain statistical analysis.

From the Table 5, it can be seen that for the results of four pictures, the uncertainty of image information in this paper is on average better than Ref. [5], and the effect is similar to Ref. [28].

4.2.4 Analysis of defense against differential attack

Cryptography differential attack is a very common attack. Differential attack as a selective plaintext attack, the attacker tends to change one or more values of pixels of original image, and then use the attacked encryption algorithm encrypt the image. By analyzing the cipher image, attacker can find out some specific relations, thus obtain the correct encryption key or expose some clear information. The diffusion operation of the image encryption algorithm can make the small change of plaintext affect the cipher-text pixel as much as possible. When the two images were completely different, the expected value of the number of pixels change rate(NPCR) was 99.6094%. The expected value of the unified average changing intensity(UACI) is 33.4635%.

From the Table 6, it can be seen that, compared with the four pictures, the algorithm proposed in this paper is better than Ref. [2] and Ref. [28] in differential attack, and the effect is similar to Ref. [5].

Table 5 Comparison of information entropy analysis

Image/H(s)	Lena	Peppers	Baboon	Boat
Our scheme	7.9974	7.9971	7.9975	7.9967
Ref. [28]	7.9975	7.9970	7.9973	7.9969
Ref. [2]	7.9972	7.9972	7.9969	7.9971
Ref. [5]	7.9963	7.9954	7.9942	7.9929

4.2.5 Ability to resist noise and cropping attacks

In the actual communication channel, cipher-text image will be disturbed by noise or cropping attacks, so the image encryption algorithm should be able to resist noise interference and data loss to a certain extent. Gaussian noise and pepper and salt noise are two common kinds of noise in image transmission. The mean square error (MSE) and the peak signal-to-noise ratio (NSPR) are used to measure. The smaller the MSE is, the stronger the anti-noise and data loss capability is. The equation is,

$$MSE = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}\left[f(x,y)-f^{'}(x,y)\right]^2 \tag{34}$$

$$PSNR = 10\log_{10}\left(255^2/MSE\right)dB \tag{35}$$

where $f(x, y)$ represents the pixel value at position (x, y), and $f'(x, y)$ represents the pixel value after data loss or noise attack at this position.

It can be seen from Figs. 10, 11 and 12 and Table 7 that the method proposed in this paper can well resist noise and data loss attacks.

4.2.6 Edge distortion

After edge detection, the result will expose the boundary and curve mark of the object. A good encryption algorithm needs to disrupt its boundary and can't get effective information by edge detection of encrypted images. Thus, two new edge-dependent evaluation parameters viz. Edge Ratio(ER) and Edge Differential Ratio(EDR) are introduced in [23]. These are mathematically represented as follows:

$$ER = \frac{\sum_{i,j=0}^{N,M}\widehat{B}(i,j)}{\sum_{i,j=0}^{N,M}B(i,j)} \tag{36}$$

Table 6 NPCR and UACI analysis and comparison

Image		NPCR	UACI
Lena	Our scheme	99.6201%	33.4490%
	Ref. [28]	99.6063%	33.3792%
	Ref. [2]	99.6521%	33.3324%
	Ref. [5]	99.6093%	33.5315%
Peppers	Our scheme	99.6002%	33.4312%
	Ref. [28]	99.6490%	33.5088%
	Ref. [2]	99.6170%	33.4048%
	Ref. [5]	99.6322%	33.6981%
Baboon	Our scheme	0.995742%	33.5371%
	Ref. [28]	99.6231%	33.5946%
	Ref. [2]	99.6246%	33.3721%
	Ref. [5]	99.617%	33.4085%
Boat	Our scheme	99.6068%	33.4217%
	Ref. [28]	99.5819%	33.4781%
	Ref. [2]	99.6047%	33.5374%
	Ref. [5]	99.583%	33.4699%

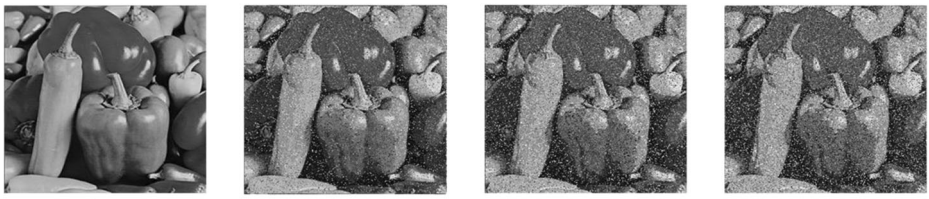


Fig. 10 The decryption effects under different levels of Gaussian noise

$$EDR = \frac{\sum_{i,j=0}^{N,M} |B(i,j) - \widehat{B}(i,j)|}{\sum_{i,j=0}^{N,M} \left(B(i,j) + \widehat{B}(i,j) \right)} \tag{37}$$

where $B(i, j)$ and $\widehat{B}(i,j)$ denote the bit value in the detected binary matrix for the original and the encrypted image, respectively.

An encrypted image should have fewer number of edges as compared to its original counterpart. In other words, lower ER value reflects better cryptosystem and vice versa. Meanwhile, high EDR values reflect that the left-out edges in the encrypted image have large deviation from the original image edges.

The values obtained for ER and EDR evaluation of the proposed technique is indicated in Table 8. The mean value of ER is less than 0.6, and the mean value of EDR is greater than 0.8, which means that more than 80% of the edge data have been transformed. The algorithm proposed in this paper can scramble the edge pixels well, and the data is better than other references.

4.2.7 Key space and key sensitivity

Key space is too small, easy to be attacked by the hacker exhaustive. In order to avoid the key being broken, the key space needs to be greater than 2^{100} [9]. The algorithm proposed in this paper requires four key values, and the accuracy of each key value is in 10^{-14}, so the total key space of this algorithm has 10^{60}, greater than 2^{100}, and it can resist exhaustive attack very well.

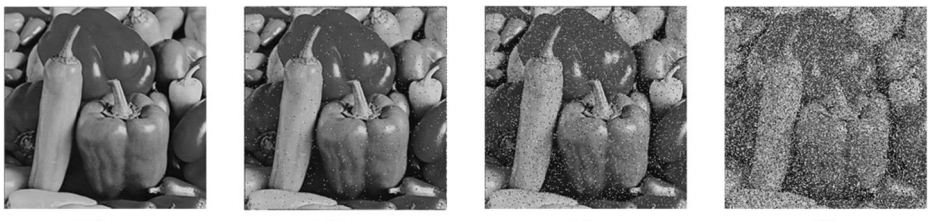


Fig. 11 The decryption effects under different levels of pepper and salt noise

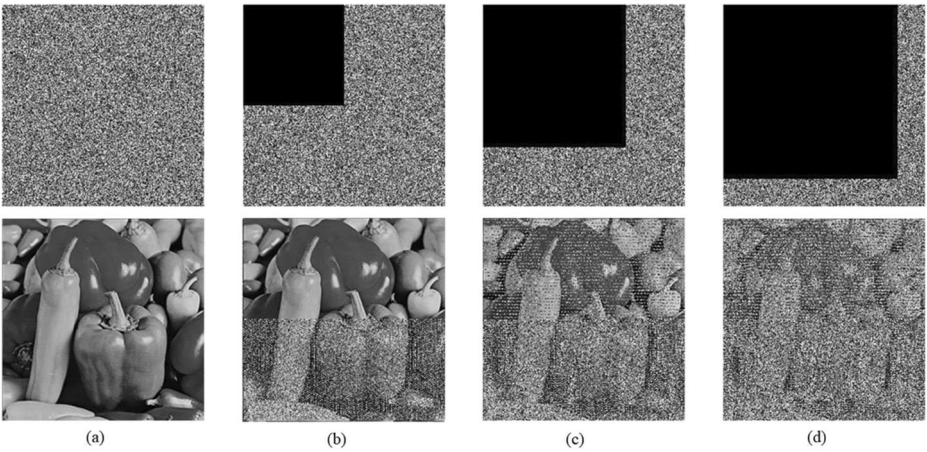


Fig. 12 The decryption effect under different degrees of data loss

Key sensitivity can be divided into encryption key sensitivity and decryption sensitivity. In encryption, two slightly different keys are used to encrypt the same picture. If the two encrypted images obtained are very different, it means that the encryption algorithm is very sensitive; if the decryption key is slightly changed in the decryption stage, If the decrypted image is a disordered image, the sensitivity of decryption is very strong.

To satisfy Kerchoff's rule, the encryption system's key needs to be sensitive enough that, knowing the encryption algorithm and not knowing the key, it is still difficult for an attacker to obtain meaningful information. The sensitivity of the key mainly includes the sensitivity of the encryption key and the sensitivity of the decryption key. Encryption image A is obtained by encrypting the original image with the keys presented in this paper, and encryption graph B is obtained after changing the keys. Figure A is decrypted using the changed keys to obtain Figure C. To measure the difference between Figure A and Figure B, and between the original and Figure C, the NPCR and UACI parameters mentioned in Sec.4.2.4 are used here.

The keys change is mainly in that $x_0 = 0.4478$ changes to $x_0 = 0.44780000000001$. From Table 9, we can see that a slight change in the keys will result in a completely different situation of encryption and decryption. It also proves that the algorithm satisfies the security of the key.

Table 7 Test results of resistance to noise and cropping attacks

Attack		MSE				PSNR			
		Our scheme	Ref. [28]	Ref. [2]	Ref. [5]	Our scheme	Ref. [28]	Ref. [2]	Ref. [5]
Gaussian noise	Variance =0.0001	28,762	21,712	21,797	22,787	3.54	4.76	4.74	4.55
	Variance =0.0003	28,672	21,775	21,742	22,261.	3.55	4.751	4.754	4.65
	Variance =0.0005	28,337	21,705	21,815	22,322	3.60	4.76	4.74	4.64
salt and pepper noise	Density=0.01	416	21,695	21,659	1921	21.93	4.76	4.7742	15.29
	Density=0.05	2158	21,781	21,663	8596	14.78	4.75	4.7735	8.78
	Density=0.25	10,813	21,812	21,681	22,643	7.79	4.76	4.7701	4.58
Cropping	1/4th	5412	21,814	21,677	10,560	10.74	4.74	4.7706	7.89
	1/2th	10,856	21,811	21,781	16,272	7.75	4.74	4.7498	6.01
	3/4th	16,199	21,662	21,762	19,972	6.0	4.77	4.7537	5.12

Table 8 ER and EDR obtained for various test images and references

Image		Lena	Peppers	Baboon	Boat
Our scheme	ER	0.7997	0.7893	0.3355	0.5492
	EDR	0.8361	0.8339	0.7741	0.8107
Ref. [28]	ER	0.8089	0.8042	0.3263	0.5434
	EDR	0.8349	0.8218	0.7728	0.8129
Ref. [2]	ER	1.7436	0.7677	0.3292	0.5622
	EDR	0.8995	0.8334	0.7725	0.8012
Ref. [5]	ER	0.8128	0.7928	0.3261	0.54
	EDR	0.8333	0.8332	0.7736	0.805

Table 9 NPCR and UACI values of images after modification of encryption and decryption keys

Image		NPCR	UACI
Lena	encryption	99.6017%	33.4031%
	decryption	99.6017%	30.2626%
Peppers	encryption	99.6322%	33.5391%
	decryption	99.6322%	30.6939%
Baboon	encryption	99.6429%	33.5429%
	decryption	99.6429%	28.7915%
Boat	encryption	99.6139%	33.4049%
	decryption	99.6139%	28.6115%

5 Conclusion

This paper proposes a unified encryption algorithm. The cryptosystem used New-Logistic-Sine map to generate the key stream for encryption/decryption process and employed the NIST test to show the key stream has good statistical characteristics. In the cryptosystem, the encryption algorithm and the decryption algorithm are the same, which can save half of the hardware and software resources while ensuring secure communication. In the simulation results and security analysis, we can see that the algorithm has a great improvement in the execution time and meets all kinds of security standards. In general, this paper does not have a great advantage in pixel distribution, which will be further studied in the future.

Acknowledgements This work is supported by the National Natural Science Foundation of China (Grant No. 61901074).

References

1. Alawida M, Teh JS, Samsudin A, Alshoura W'H (2019) An image encryption scheme based on hybridizing digital chaos and finite state machine. Signal Process 164:249–266
2. Alawida M, Samsudin A, Teh JS, Alkhawaldeh RS (2019) A new hybrid digital chaotic system with applications in image encryption. Signal Process 160:45–58
3. Belazi A, Abd el-Latif AA, Belghith S (2016) A novel image encryption scheme based on substitution-permutation network and chaos. Signal Process 138:155–170
4. Chai X, Zheng X, Gan Z, Han D, Chen Y (2018) An image encryption algorithm based on chaotic system and compressive sensing. Signal Process 148:124–144
5. Chai X, Gan Z, Yuan K, Chen Y, Liu X (2019) A novel image encryption scheme based on DNA sequence operations and chaotic systems. Neural Computing & Application 31(1):219–237
6. Chen J, Zhu Z-l, Fu C, Zhang L-b, Zhang Y (2015) An image encryption scheme using nonlinear inter-pixel computing and swapping based permutation approach. Commun Nonlinear Sci Numer Simulat 23(1–3): 294–310

7. Chun C, Sun K, Liu W (2018) A novel bit-level image encryption algorithm based on 2D-LICM hyperchaotic map. Signal Process 143:122–133
8. Dai Y, Wang X (2012) Medical image encryption based on a composition of Logistic Maps and Chebyshev Maps. 2012 IEEE International Conference on Information and Automation, Shenyang, 210–214. https://doi.org/10.1109/ICInfA.2012.6246810
9. Farah MAB, Farah A, Farah T (2019) An image encryption scheme based on a new hybrid chaotic map and optimized substitution box. Nonlinear Dynamics 99:3401–3064
10. Hongjun L, Wang X, Kadir A (2012) Image encryption using DNA complementary rule and chaotic maps. Appl Soft Comput 12:1457–1466
11. Hua Z, Jin F, Xu B, Huang H (2018) 2D logistic-sine-coupling map for image encryption. Signal Process 149:148–161
12. Hua Z, Yi S, Zhou Y (2018) Medical image encryption using high-speed scrambling and pixel adaptive diffusion. Signal Process 144:134–144
13. Jha DP, Kohli R, Gupta A (2016) Proposed encryption algorithm for data security using matrix properties. International Conference on Innovation and Challenges in Cyber Security (ICICCS-INBUSH) 2016:86–90. https://doi.org/10.1109/ICICCS.2016.7542316
14. Kohli R, Kumar M (2013) FPGA implementation of cryptographic algorithms using multi-encryption technique. International Journal of Advanced Research in Computer Science and Software Engineering 3(5):112–120
15. Joshua C. D, Jian-Ping Li, Prince C. Addo (2019) An image cryptosystem based on pseudorandomly enhanced chaotic DNA and random permutation. Multimedia Tools & Applications 78:24979–25000
16. Lin T, Wang X, Meng J (2018) A chaotic color image encryption using integrated bit-level permutation. Multimedia Tools & Applications 77:883–6896
17. Mohamed KF (2014) A parallel block-based encryption schema for digital images using reversible cellular automata. Engineering Science & Technology, an International Journal 27(2):85–94
18. Murillo-Escobar MA et al (2017) A novel pseudorandom number generator based on pseudorandomly enhanced logistic map. Nonlinear Dynamics 87(1):407–425
19. Nanrun Z et al (2015) Image compression–encryption scheme based on hyper-chaotic system and 2D compressive sensing. Opt Laser Technol 82:121–133
20. Sara T, Nijad AN (2014) Image cryptographic algorithm based on the Haar wavelet transform. Inf Sci 269: 21–34
21. Souyah A, Faraoun KM (2016) Fast and efficient randomized encryption scheme for digital images based on Quadtree decomposition and reversible memory cellular automata. Nonlinear Dynamics 84:715–732
22. Taneja N, Raman B (2012) Combinational domain encryption for still visual data. Multimedia Tools & Applications 59:775–793
23. Taneja N, Raman B, Gupta I (2011) Selective image encryption in fractional wavelet domain. AEUE - International Journal of Electronics and Communications 65(4):338–344
24. Taneja N, Raman B, Gupta I (2012) Chaos based cryptosystem for still visual data. Multimedia Tools & Applications 61(2):281–298
25. Taneja N, Bhatnagar RB et al (2013) Joint watermarking and encryption for still visual data. Multimedia Tools & Applications 67(3):593–606
26. Ying-Qian Z, Wang X-Y (2015) A new image encryption algorithm based on non-adjacent coupled map lattices. Appl Soft Comput 26:10–20
27. Yong Z (2018) The image encryption algorithm based on chaos and DNA computing. Multimedia tools & applications 77(16):21589–21615
28. Yong Z (2018) The unified image encryption algorithm based on chaos and cubic S-box. Inf Sci 450:361–377
29. Zhang Y, Tang Y (2018) A plaintext-related image encryption algorithm based on chaos. Multimedia Tools & Applications 77:6647–6669. https://doi.org/10.1007/s11042-017-4577-1
30. Zheng J, Zheng L, Tang Z (2020) An Image Encryption Algorithm Based on Multichaotic System and DNA Coding. Discret Dyn Nat Soc. Article ID 5982743, pages 16. https://doi.org/10.1155/2020/5982743
31. Zhou S (2015) Image Encryption Technology Research Based on Neural Network. 2015 International Conference on Intelligent Transportation, Big Data and Smart City, Halong Bay: 462–465. https://doi.org/10.1109/ICITBS.2015.119
32. Zhu Z, Zhang W, Wong K-w, Yu H (2011) A chaos-based symmetric image encryption scheme using a bit-level permutation. Inf Sci 181:1171–1186