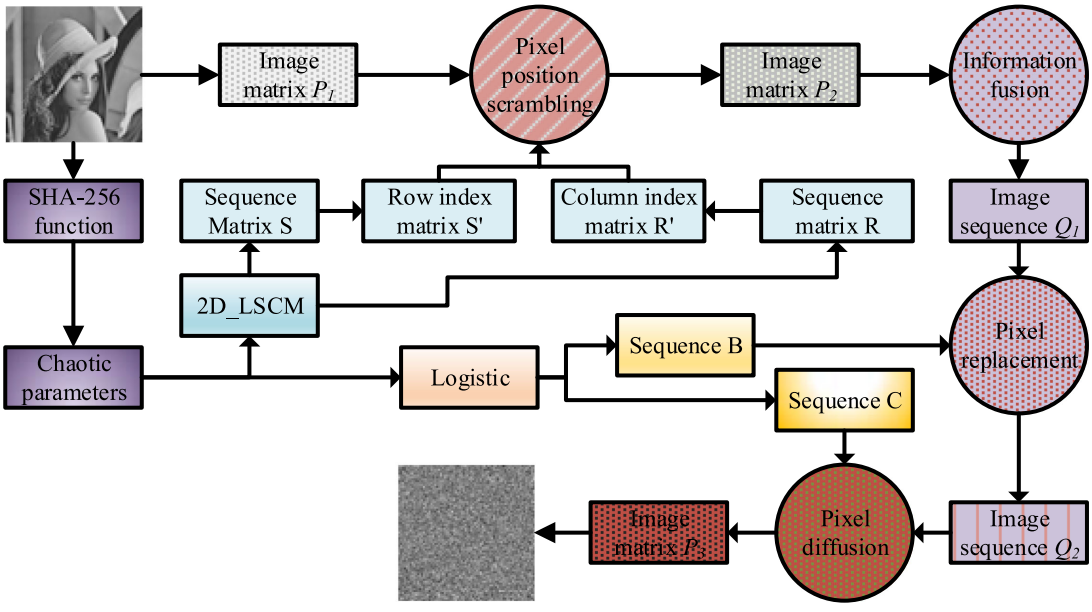
IEEE Photonics Journal
An IEEE Photonics Society Publication

Open Access

A Novel Image Encryption Algorithm Based on Chaotic Sequences and Cross-Diffusion of Bits

Volume 13, Number 1, February 2021

Geng Shengtao
Wu Tao
Wang Shida
Zhang Xuncai, *Member, IEEE*
Niu Ying



DOI: 10.1109/JPHOT.2020.3044222

A Novel Image Encryption Algorithm Based on Chaotic Sequences and Cross-Diffusion of Bits

Geng Shengtao,[1] **Wu Tao**,[1] **Wang Shida,**[1]
Zhang Xuncai,[1] ***Member, IEEE*, and Niu Ying**[2]

[1]School of Electrical and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China
[2]School of Architecture Environment Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China

DOI:10.1109/JPHOT.2020.3044222

Manuscript received October 9, 2020; revised December 3, 2020; accepted December 6, 2020. Date of publication December 11, 2020; date of current version January 6, 2021. This work was supported in part by the National Natural Science Foundation of China under Grants 62072417 and U1804262, and in part by the Key Research and Development Program of Henan Province under Grants 202102210177 and 192102210134. Corresponding authors: Zhang Xuncai; Wu Tao (e-mail: zhangxuncai@pku.edu.cn; 771767203@qq.com).

Abstract: To protect image information security, we propose an encryption algorithm based on chaotic sequences and bits' cross-diffusion. First, two chaotic sequences are generated by two-dimensional logic-sine coupling mapping, and the original image is scrambled with the two sequences. Secondly, the scrambled image is transformed into a one-dimensional sequence, and the low-order bits between every two pixels are fused to change the detailed information of the image, improved the ability to resist differential attacks. Finally, chaotic sequences are generated by iterative logical mapping to perform pixel replacement and ciphertext diffusion. The known plaintext attack can be effectively resisted because of the stochastic combination of diffusion scheme and chaotic sequence. It also has a significant effect on resisting differential attack. Take Lena image as an example, NPCR and UACI can reach 99.6063% and 33.4477%, respectively, which are very close to the ideal value. The encryption scheme has good key sensitivity and strong ability to resist statistical attacks, which shows that the algorithm has adequate security.

Index Terms: Chaotic mapping, cross-diffusion of bits, image encryption.

1. Introduction

It is well known that an effective encryption strategy can better protect the image information. With the continuous development of society, especially in the 5G era, digital images have been widely used in people's social life. Improving the security of digital images in the transmission process has become one of the important topics in computer science. Due to the large amount of information carried by the image and the high correlation between adjacent pixels, the traditional encryption algorithm can no longer meet the rapidly developing image encryption requirements and encryption data with this characteristic will lead to inefficiency.

It is a good choice to encrypt images by a chaotic system because the chaotic system has high randomness, good ergodicity and sensitivity of initial values, which correspond to avalanche, diffusivity and confusability of cryptography [1]. Therefore, the chaotic system has been widely

used in image encryption algorithms [2]–[6]. Previous chaotic encryption technologies were mostly based on low-dimensional discrete chaotic maps [7]–[10]. Due to the limitation of finite calculation precision, low-dimensional chaotic systems have weaknesses of a small period and few periodic orbits, leading to low security of cryptographic techniques. As hyperchaotic systems have more than two Lyapunov indexes, large key space, and more complex and unpredictable nonlinear behavior, they have great application potential in image encryption, which has aroused great research interest scholars. Enayatifar [11] combined two one-dimensional chaotic mappings into a two-dimensional chaotic mapping for the image encryption. Hua *et al.* [12] proposed a two-dimensional chaotic sequence generation model combining sine mapping and logistic mapping in 2015, and the model uses sine mapping and parameters to adjust the output of logistic mapping, thereby enhancing the nonlinearity of the two-dimensional chaotic sequence and randomness. This method improves the encryption system's security, but the encryption algorithm cannot effectively resist the differential attack. The image encryption method only using the chaotic system also has many shortcomings, such as chaotic degradation and low defense ability. Therefore, the combination of chaotic systems and other techniques has become a research hotspot at present. The scrambling operation can not only change the pixel value but also break the correlation with the original pixel value, making the image encryption more secure [13]–[16]. Many bit-based encryption methods have been proposed to reduce the correlation between adjacent pixels [17], [18]. However, these encryption techniques have some limitations. For example, the replacement phase is repetitive [19] and has a high time complexity [20], [21]. Li [22] proposed a robust encryption scheme with an aperiodic chaotic map and random cyclic displacement to solve this problem. First, the original image was scrambled using the aperiodic generalized Arnold transform. Then, for the scrambled image, by changing the pixel value at the specific level and performing a circular bit shift randomly for each pixel, the algorithm's encryption speed is improved, but the algorithm is simple and unsafe. Chai [23] proposed an image encryption algorithm based on Brownian motion, the bit pixels in each bit plane are used as Brownian motion particles to scramble the 8-bit planes of ordinary images, and further improved the security of the algorithm by combining with one-dimensional chaotic system. Still, the encryption process takes a long time.

In addition to the above encryption algorithms, some image encryption technologies are based on the combination of DNA coding technology and chaotic system [24], [25]. However, the chaotic system and DNA encryption scheme have some problems, such as small space and cannot resist differential attacks. Yang [26] applied a new Lorenz chaotic system to encryption, this algorithm has a large key space and can effectively resist brute force attack, but it increases encryption time. Tu [27] proposed through DNA coding technology to encrypt the color image. The spatial structure is relatively complex, but the input parameters are few. Liu [28] proposed an encryption scheme combining dynamic S-box and chaotic system. Different from the traditional DNA-based diffusion method, the dynamic S-box generated by the DNA sequence is used to diffuse the pixel value of the image. These research methods have achieved good encryption effect, but there are still some shortcomings. Because of the limited variety of DNA coding rules and DNA coding operations, these schemes fail to make full use of these limited rules, which weakens the significance of using DNA coding to some extent.

To solve these problems, this paper proposes an algorithm to encrypt images by two chaotic systems; thereby, there are more key parameters, increasing the key space and better resist brute force attacks. The pixel scrambling method using two chaotic mapping matrices can effectively break the correlation between the image's adjacent pixels. Fusing the pixel's low-bit information to change the details of the image can better resist differential attacks. Finally, ciphertext diffusion effectively improves the security of the encryption algorithm.

We organize the rest of this paper as follows. Section 2 introduces the Two-Dimensional Logistic-Sine Coupling Mapping. Section 3 introduces the encryption process. Section 4 analyzes the security of the encryption algorithm proposed in this paper and Section 5 concludes this paper.

2. Theoretical Basis

The classical logistic mapping is a one-dimensional discrete time nonlinear mapping, which has the following three characteristics: (1) Extremely dependent on initial conditions, (2) aperiodic, (3) strange attractor. Its expression is:

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

where $x \in [0,\ 1]$, $\mu \in [3.57,\ 4]$, when $\mu > 3.57$, the system is in a chaotic state.

The expression of sine mapping is:

$$x_{n+1} = \beta \sin \pi x_n, \tag{2}$$

where β is a control parameter and $\beta \in [0, 1]$.

Logistic mapping and sine mapping have some disadvantages, such as simple behavior and weak chaotic interval, which may negatively influence some applications based on chaos. When logistic mapping and sine mapping are coupled, a new chaotic map is obtained, which has a rather complex chaos behavior, namely, Two-Dimensional Logistic-Sine Coupling Mapping (2D-LSCM), which is defined as:

$$\begin{cases} x_{n+1} = \sin(\pi\ (4\theta x_n (1 - x_n) + (1 - \theta) \sin(\pi y_n)) \\ y_{n+1} = \sin(\pi\ (4\theta y_n (1 - y_n) + (1 - \theta) \sin(\pi x_{n+1})) \end{cases}, \tag{3}$$

where θ is a control parameter and $\theta \in [0, 1]$. As can be seen from the definition, the logical and sine mappings are coupled, and then perform a sine transformation based on the coupling, expanding the dimension from one dimension to two dimensions. Based on this method, the complexity of logistic mapping and sine mapping can be improved to obtain complex chaotic behavior.

3. Encryption Scheme

Given an $M \times N$ original image matrix $\boldsymbol{P}$, the encryption algorithm is mainly composed of scrambling process, pixel replacement, cross-diffusion of bits and diffusion process.

3.1 Key Generation

To increase the algorithm's security and make the key correlate with the original image, we generate the chaotic system's initial parameters according to the original image's hash value. SHA-256, the name comes from the abbreviation of Secure Hash Algorithm 2, which is a standard for hash function algorithm. Sha-256 generates a 256-bit hash value for any length of a message, and the resulting hash is called a message digest. Original image matrix $\boldsymbol{P}$ input SHA-256 algorithm, get 256-bit hash value H. The hash value H is divided into 32 binary sequences, get $k_1,\ k_2,\ \ldots,\ k_{32}$. The parameter μ of logistic chaotic map was set as 4, 2D-LSCM initial parameters x_0, y_0, θ_0 and the initial parameters z_0 of the logistic chaotic map are calculated by Eq. (4) and Eq. (5)

$$\begin{cases} Q_1 = k_1\ \oplus k_2 \oplus k_3 \oplus k_4 \oplus k_5 \oplus k_6 \oplus k_7 \oplus k_8 \\ Q_2 = k_9\ \oplus k_{10} \oplus k_{11} \oplus k_{12} \oplus k_{13} \oplus k_{14} \oplus k_{15} \oplus k_{16} \\ Q_3 = k_{17}\ \oplus k_{18} \oplus k_{19} \oplus k_{20} \oplus k_{21} \oplus k_{22} \oplus k_{23} \oplus k_{24} \\ Q_4 = k_{25}\ \oplus k_{26} \oplus k_{27} \oplus k_{28} \oplus k_{29} \oplus k_{30} \oplus k_{31} \oplus k_{32} \end{cases}, \tag{4}$$

$$\begin{cases} x_0 = \frac{1}{256}\ mod\ (Q_1 + Q_4, 256) + x_0' \\ y_0 = \frac{1}{256}\ mod\ (Q_2 + Q_4, 256) + y_0' \\ \theta_0 = \frac{1}{256}\ mod\ (Q_3 + Q_4, 256) + \theta_0' \\ z_0 = \frac{1}{3}\ (x_0 + y_0 + \theta_0) + z_0' \end{cases}, \tag{5}$$

3.2 Pixel Scrambling

Adjacent pixels of an image may have a high correlation and data redundancy, and scrambling can remove these high correlations. Two chaotic mapping matrixes are used to generate index

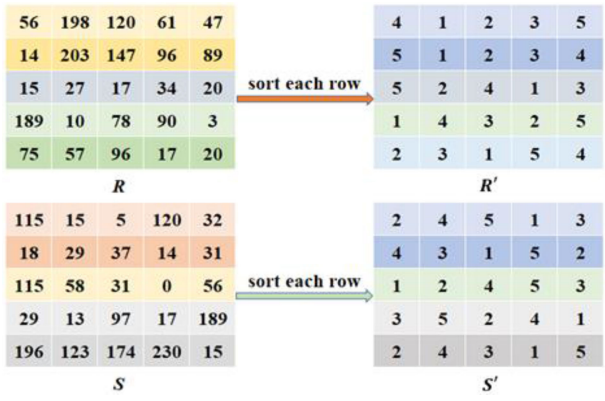


Fig. 1. Builds the index matrix.

matrixes to scramble the image matrix ***P***, and adjacent pixels can be randomly replaced to different positions. The scrambling process is described as follows:

Generate chaotic sequence: Give the initial parameters of the 2D-LSCM mapping, which are iterated 1000 times to eliminate the transient effect. Continue to iterate $M \times N$ times to produce the sequence $\boldsymbol{U} = \{u_1, u_2, \ldots, u_{M\times N}\}$ and $\boldsymbol{V} = \{v_1, v_2, \ldots, v_{M\times N}\}$, Process the data according to the Eq. (6) and Eq. (7) to obtain the sequence ***U'*** and ***V'***, ensure that the obtained data are within a given range.

$$u_i' = floor\left(mod\left(2^{32} \times u_i, 256\right)\right), \tag{6}$$

$$v_i' = floor\left(mod\left(2^{32} \times v_i, 256\right)\right), \tag{7}$$

Build index matrix: Map the sequence ***U'*** and ***V'*** to $M \times N$ matrices ***S*** and ***R***, sort the elements of each row of the matrices ***S*** and ***R*** in descending order. After the sorting is completed, return their position index and generate as shown in Fig. 1 position index matrix ***S'*** and matrix ***R'***. The matrix ***S'*** and matrix ***R'*** are respectively used as the row index matrix and column index matrix;

Pixel position swap: Step 1: Set row index $m = 1$; Step 2: Select the pixels in P with positions $\{[S'_{m,1}, R'_{m,1}], [S'_{m,2}, R'_{m,2}], [S'_{m,3}, R'_{m,3}] \ldots [S'_{m,N}, R'_{m,N}]\}$ and corresponding positions $\{[m, 1], [m, 2][m, 3] \ldots [m, N]\}$ pixels are exchanged in sequence.

Iterate Step 1 to Step 2 for $m = 2 \sim M$, scrambling each row's elements, in turn, obtain the scrambled image matrix ***P'***.

As shown in Fig. 2, we give a 5×5 size image matrix as an example to explain the scrambling process. As shown in Fig. 3, through one scrambling operation, the original image's information cannot be recognized.

3.3 Cross-Diffusion of Bits

A digital image can be broken down into eight bit-planes; different bit planes represent different image information. The high bit plane represents the image's outline information. The low bit plane represents the detailed information of the image, and the middle bit plane represents the image background information. Processing of different bit planes is equivalent to the processing of different information positions of the image. For example, for a gray image with an amplitude gray value between [0, 255], each pixel can be expressed as an eight-bit number, and the bits at the same position of all pixels are combined to form different bit planes. Different bit planes contain different amounts of information [29], as shown in Table 1.

The information contained in the A1∼A3 plane matrix accounts for about 2.64% of the total matrix information, which includes the detailed information of the image. Change the bit value of the image matrix $A_1 \sim A_3$ plane makes the image blurry, which can hide the details of the image

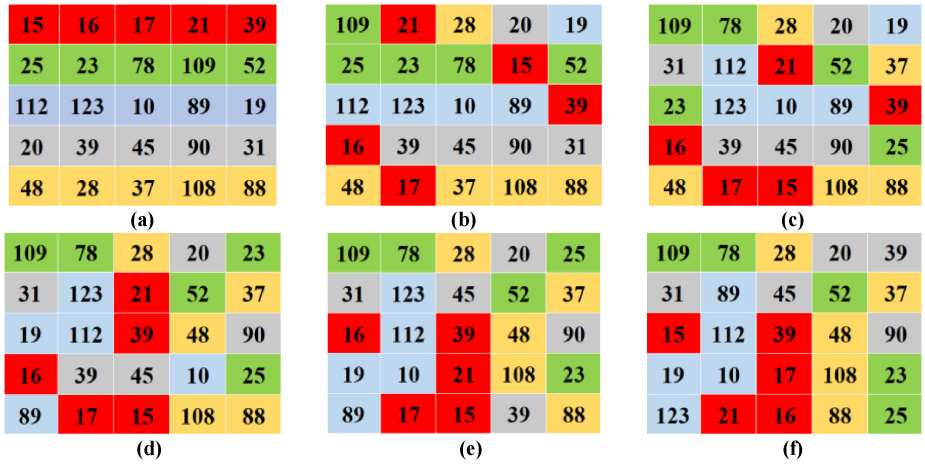


Fig. 2. The scrambled image matrix. (a) The original image matrix P. (b) The scrambled image matrix P_1. (c) The scrambled image matrix P_2. (d). The scrambled image matrix P_3. (e) The scrambling image matrix P_4. (f)The scrambled image matrix P_5.

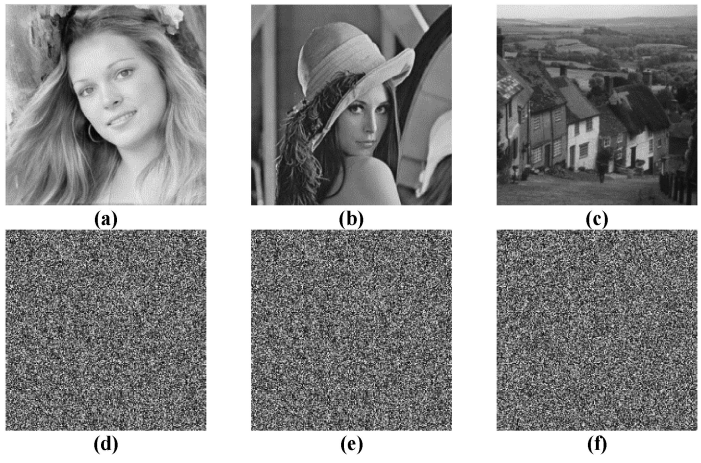


Fig. 3. The original image and scrambled image. (a) The original Elaine image. (b) The original Lena image. (c) The original Hill image. (d) The scrambled Elaine image. (e) The scrambled Lena image. (f) The scrambled Hill image.

TABLE 1
Information on Each Bit Plane

Bit plane	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1
Amount of information (%)	50.20	25.10	12.55	6.28	3.14	1.57	0.78	0.39

more effectively, thereby improving the ability to resist differential attacks. For a pixel, its binary form can be expressed as $a_8a_7a_6a_5a_4a_3a_2a_1$, randomly select two pixels, and exchange the $a_1 \sim a_3$ bits two-pixel values, which can effectively change the details of the two pixels. For two pixels *a*, *b*, the exchange results are *a'* and *b'*, the cross-diffusion of bits rules between the two elements are shown in Fig. 4 below.

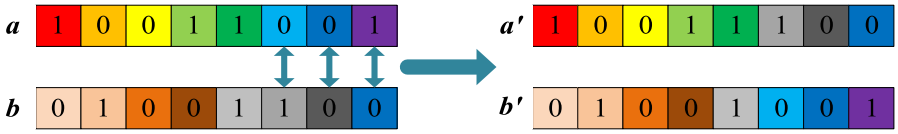


Fig. 4. Pixel fusion rule.

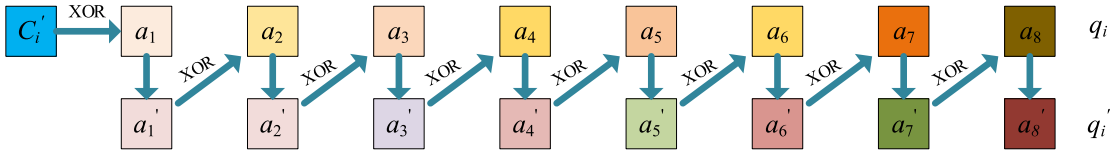


Fig. 5. Bit arithmetic rules.

3.4 Pixel Replacement

The pixel's value can be changed randomly by replacing the pixel so that the pixel value distribution presents a random state. Transform image matrix $\boldsymbol{P}$ into a one-dimensional sequence $\boldsymbol{Q} = \{q_1, q_2, q_3, \ldots, q_{(M \times N)}\}$, chaotic sequence $\boldsymbol{B} = \{b_1, b_2, b_3, \ldots, b_{(M \times N)}\}$. According to formula (8), the sequence Q elements are replaced to obtain the sequence $\boldsymbol{Q'}$.

$$q_i' = floor\left(mod\left(10^{10} \times b_i + q_i, 256\right)\right). \tag{8}$$

3.5 Bit Diffusion

The image encryption algorithm can effectively disrupt the relationship between the original image and the cipher image in the case of diffusion. The chaotic sequence $\boldsymbol{C} = \{c_1, c_2, c_3, \ldots, c_{(M \times N)}\}$, as the key stream for bit diffusion. Sequence C is converted into a binary sequence for the convenience of diffusion to obtain the sequence $\boldsymbol{C'}$. The conversion rules are as follows:

$$\begin{cases} c_j' = 1 \text{ if } c_j > 0.5 \\ c_j' = 0 \text{ else} \end{cases}, \tag{9}$$

The diffusion scheme of the current pixel value is determined based on the sequence C' and the previous diffusion pixel value. Small changes in the original image can spread throughout the encrypted image. The detailed diffusion process is as follows.

Transform the image matrix $\boldsymbol{P}$ into a one-dimensional sequence $\boldsymbol{Q} = \{q_1, q_2, q_3, \ldots, q_{(M \times N)}\}$, and let the sequence after diffusion as $\boldsymbol{E} = \{e_1, e_2, e_3, \ldots, e_{(M \times N)}\}$, in order to make the calculation reversible, let $e_1 = q_1$. Set $L_j = c_j' + t$, where $j = 2, 3, \ldots, M \times N$, t can take any bit of the pixel value $e_{(j-1)}$, where the highest bit of the pixel $e_{(j-1)}$ is selected. For any pixel q_i, its bits are expressed as $a_8 a_7 a_6 a_5 a_4 a_3 a_2 a_1$;

1) If $L_j = 0$, the current pixel value q_i remains unchanged;
2) If $L_j = 1$, the operation rules are shown in Eq. (10), and the corresponding $a_1' \sim a_8'$ are replaced by the corresponding $a_1 \sim a_8$ of q_i, respectively;

$$\begin{cases} a_j' = a_i \ \ if \ \ c_j' = 0 \\ a_j' = \overline{a_j} \ \ else \end{cases}, \tag{10}$$

3) If $L_j = 2$, c_j' is used to perform XOR operation with a_1 in q_j. Each operation result replaces the original bit and applies to the next bit XOR calculation, thus resulting $a_1' \sim a_8'$ replaces the corresponding $a_1 \sim a_8$ in q_j, and the detailed operation process is shown in Fig. 5.

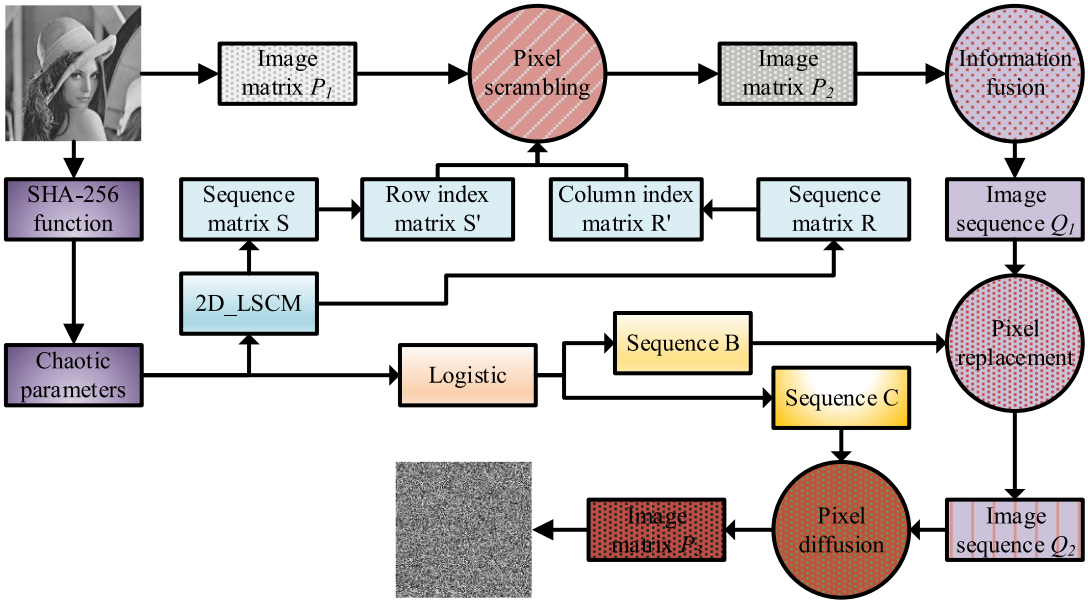


Fig. 6. The encryption flowchart.

3.6 Encryption Step

The encryption process of the image encryption algorithm is divided into two parts. The first part is the pixel scrambling process: the chaotic sequence generated by the 2D-LSCM chaotic system is used as the index matrix to scramble the pixel position; second, pixel transformation and diffusion. After cross-diffusion of bits is carried out for each pixel of the original image, the remainder is taken for pixel replacement. Finally, the logistic system will generate the key stream to encrypt image diffusion. An encryption flowchart as shown in Fig. 6 is presented to explain the encryption process.

Input: Original image $\boldsymbol{P}$, initial parameters $x_0^{'}$, $y_0^{'}$, $\theta_0^{'}$, $z_0^{'}$.
Output: Cipher image $\boldsymbol{P_3}$.

Transform the original image $\boldsymbol{P}$ into image matrix $\boldsymbol{P_1}$ with the size of $M \times N$:

1) **Initial value generation:** Input the image matrix $\boldsymbol{P_1}$ into SHA-256 algorithm and output 256 bits hash value H, the chaotic initialization parameters x_0, y_0, θ_0 and z_0 were calculated according to Eq. (4) and Eq. (5);
2) **Pixel scrambling:** Iterative 2D-LSCM chaotic mapping produces two chaotic sequences $\boldsymbol{U}$ and $\boldsymbol{V}$, and processes the data according to Eq. (6) and Eq. (7) to obtain the sequence $\boldsymbol{U'}$ and $\boldsymbol{V'}$. Map the sequence $\boldsymbol{U'}$ and $\boldsymbol{V'}$ into the matrices $\boldsymbol{R}$ and $\boldsymbol{S}$ of size $M \times N$. Sort each row of two matrices in descending order, and return its position index, and then generate the row index matrix $\boldsymbol{S'}$ and column index matrix $\boldsymbol{R'}$. Scrambling the image matrix $\boldsymbol{P_1}$ according to the method described in Section 3.2; obtaining the image matrix $\boldsymbol{P_2}$;
3) **Cross-diffusion of bits:** According to the cross-diffusion of bits method described in Section 3.3, the image matrix $\boldsymbol{P_2}$ is transformed into a one-dimensional sequence $\boldsymbol{Q} = \{q_1, q_2, q_3, \ldots, q_{(M \times N)}\}$, and the $a_1 \sim a_3$ of every two-pixel bits are exchanged, the image sequence $\boldsymbol{Q_1}$ is obtained. If the length of the one-dimensional sequence is odd, the last element is not exchanged;
4) **Pixel replacement:** Through logistic system to generate a chaotic sequence of length $2 \times M \times N$, and intercept the first $M \times N$ elements of the sequence as chaotic sequence $\boldsymbol{B} = \{b_1, b_2, b_3, \ldots, b_{(M \times N)}\}$. Cut off the last $M \times N$ element are used as chaotic sequence $\boldsymbol{C} = \{c_1,$

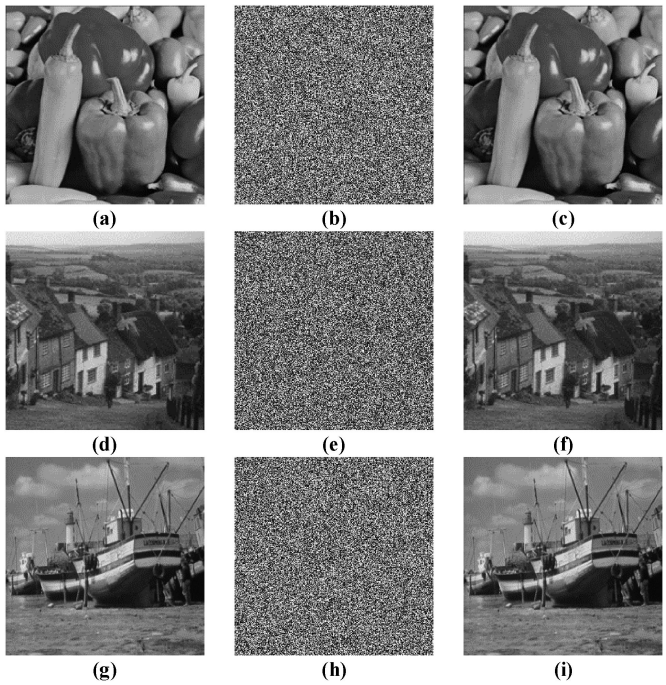


Fig. 7. The original images, cipher images and decrypted images. (a) The original Peppers image. (b) The encrypted Peppers image. (c) The decrypted Peppers image. (d) The original Hill image. (e) The encrypted Hill image. (f) The decrypted Hill image. (g) The original Boat image. (h) The encrypted Boat image. (i) The decrypted Boat image.

$c_2, c_3, \ldots, c_{(M \times N)}\}$, according to formula (8), the chaotic sequence $\boldsymbol{B}$ is used to replace the elements in the sequence $\boldsymbol{Q_1}$ to obtain the sequence $\boldsymbol{Q_2}$;

5) **Bit diffusion:** According to the encryption diffusion technology described in Section 3.5, the $\boldsymbol{Q_2}$ sequence is encrypted and diffused, and the diffused sequence is converted into a matrix of size $M \times N$ to obtain an image matrix $\boldsymbol{P}_3$, that is cipher images.

4. Experimental Results and Safety Analysis

To verify the algorithm's security, a computer with Windows 10, 8.00 GB RAM configuration, Intel(R) Core(TM) i7-4510 CPU @ 2.00GHz was used for experimental simulation of the python3.7 version. In this paper, experiments were performed on pepper, Hill, Boat, and Face images with the size of 256 $\times$ 256. The system parameters x_0', y_0', θ_0', and z_0' of Eq. (5) were all set to 0. The original image, cipher image and decrypted image are shown in Fig. 7. Through intuitive observation, the cipher image can no longer see the characteristics of the original image.

The following is the security analysis of key space, histogram analysis, correlation analysis, histogram distribution, noise attack and clipping attack analysis.

4.1 Key Space

Brute force cracking attacks are the most common and simple method to crack cipher images; the attacker can break the cipher image by trying each key. The larger the key space of the algorithm, the stronger the ability to resist brute force attacks. This algorithm's key includes x_0, y_0, θ_0, z_0 and a 256-bit hash value generated by SHA-256, and the calculation accuracy of x_0, y_0, θ_0, and z_0 is 10^{-10}, SHA-256. The key space is 2^{128}, so the total key space is 3.4028×10^{82}. Therefore, the key space of this algorithm is very large. With the existing computer technology, it is difficult to directly

TABLE 2
Encryption Key Sensitivity

Metrics	NPCR (%)	UACI (%)
$x_0 + 10^{-10}$	99.6155	33.4800
$y_0 + 10^{-10}$	99.6460	33.3433
$\theta_0 + 10^{-10}$	99.6216	33.4156
$z_0 + 10^{-10}$	99.6124	33.4191

TABLE 3
Decrypted Key Sensitivity Analysis

Metrics	NPCR (%)	UACI (%)
$x_0 + 10^{-10}$	99.5819	33.4097
$y_0 + 10^{-10}$	99.6124	33.3021
$\theta_0 + 10^{-10}$	99.6033	33.4552
$z_0 + 10^{-10}$	99.6307	33.4273

find the key used by the encryption algorithm by brute force attack; thus, the algorithm in this paper can resist exhaustive attacks.

4.2 Key Sensitivity Analysis

Key sensitivity means that the encrypted image generated by the encryption algorithm on the same image with two slightly different encryption keys is also completely different. The key with a slight difference from the decrypted key can also decrypt the same encrypted image will cause failure. The image encryption algorithm's key sensitivity is directly proportional to its ability to resist brute force attacks. By fine-tuning the key and analyzing the key's sensitivity, the image encryption algorithm's diffusion effect can be detected. Since an attacker may encrypt the original image with a guessed part of the correct key, it is possible to reconstruct a part of the original image based on careful analysis of the encrypted image. Usually, NPCR (pixel change rate) and UACI (pixel average change intensity) are used to measure the key's sensitivity. NPCR and UACI are shown in Eq. (11).

$$\begin{cases} NPCR = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} D_{(i,j)}}{M \times N} \times 100\% \\ UACI = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} |P_1(i,j) - P_2(i,j)|}{255 \times M \times N} \times 100\% \end{cases} \tag{11}$$

If $\boldsymbol{P}_{1(i,j)} \neq \boldsymbol{P}_{2(i,j)}$, $\boldsymbol{D}_{(i,j)} = 1$; otherwise, $\boldsymbol{D}_{(i,j)} = 1$. The theoretically expected values of NPCR and UACI are respectively 99.6094% and 33.4635%. Taking Lena as an example, as shown in Table 2, when the parameters of the original key are increased by 10^{-10} respectively, the values of NPCR and UACI between the cipher image $\boldsymbol{P}_2$ encrypt by key change and the cipher image $\boldsymbol{P}_1$ encrypt by the original key.

The sensitivity of the key is more obvious in the decrypted process. When the decryption key is slightly changed, the decrypted image will be very different from the original image, either through the comparison of corresponding pixel values or through the analysis of visual effects. Take the Lena image as an example; each time only one of the parameter variables is increased by 10^{-10}, respectively, the resulting decrypted image is shown in Fig. 8. Table 3 shows the various indicators of the difference between the decrypted image and the original image when the decrypted key changes slightly. It shows that the key sensitivity of the algorithm is sensitive and can effectively protect the image information.

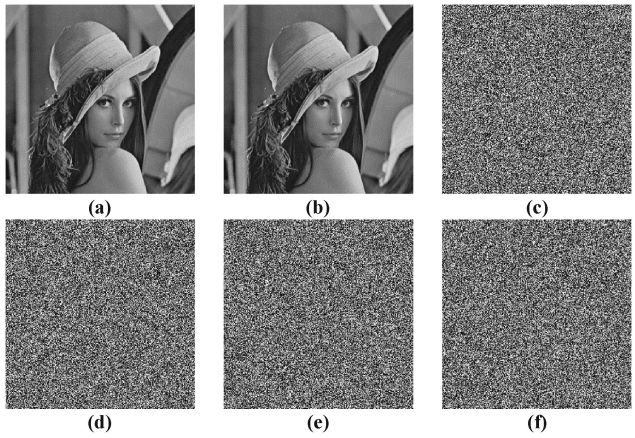


Fig. 8. The analysis of key sensitivity. (a) The original Lena image. (b) The decrypted Lena image. (c) The decrypted Lena image with $x_0 + 10^{-10}$. (d) The decrypted Lena image with the $y_0 + 10^{-10}$. (e) The decrypted Lena image with the $\theta_0 + 10^{-10}$. (f) The decrypted Lena image with the $z_0 + 10^{-10}$.

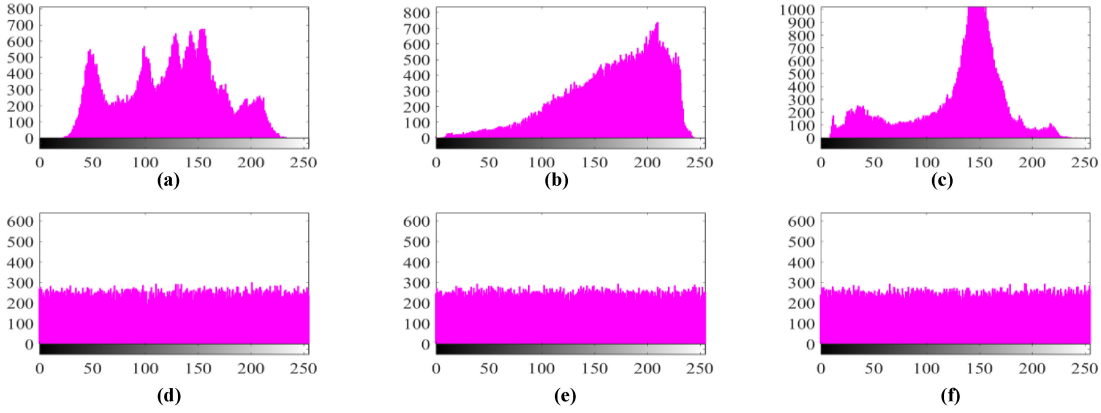


Fig. 9. The histogram of the original image and encrypted image. (a) The histogram of original Lena image. (b) The histogram of original Boat image. (c) The histogram of original Face image. (d) The histogram of encrypted Lena image. (e) The histogram of encrypted Boat image. (f) The histogram of encrypted Face image.

4.3 Histogram Analysis

The histogram represents the distribution of image features and shows the general regularity of images. Fig. 9 shows the histograms of the original image and the cipher image of Lena, Boat, and Face. Fig. 9(a), Fig. 9(c) and Fig. 9(e) show that in the original image, the pixel value distribution is uneven. But Fig. 9(b), Fig. 9(d) and Fig. 9(f) show that the pixel value of the cipher image presents a flat and uniform distribution characteristic.

In addition to cipher image visual analysis, the chi-square (2) test [30] was used to analyze the difference between the original and cipher images. For grayscale images of size $M \times N$, chi-square (2) is defined as follows:

$$\begin{cases} x^2 = \sum_{L=0}^{255} \frac{(observed\ value - expected\ value)^2}{Expected\ value} \\ expected\ value = \frac{M \times N}{256} \end{cases} \tag{12}$$

If the significance level is 0.05 and the x^2_{test} result of the test on the encrypted image is lower than $x^2_{0.05}(255) = 293.25$, then the histogram can be considered to be evenly distributed. As shown in

TABLE 4
Chi-Square Test of the Histogram

Image	Lena	Boat	Face
$x^2_{0.05}$	293.25	293.25	293.25
x^2_{test}	229.5313	273.4063	213
Decision	Pass	Pass	Pass

TABLE 5
Correlation Analysis With The Existing Methods for LENA Image in Size 256 × 256

	Original	Ours	Ref.[31]	Ref[32]	Ref[33]	Ref[34]
Horizontal	0.9621	-0.0020	-0.0072	0.0617	-0.0061	0.0012
Vertical	0.9247	-0.0065	0.0062	-0.0033	0.0094	0.0021
Diagonal	0.9190	0.0087	0.0120	-0.0022	0.0081	0.0115

Table 4, the algorithm in this paper tested the generated encrypted images, and the results were all lower than the theoretical value of 293.25. Therefore, the encryption algorithm proposed in this paper has passed the x^2_{test} test.

4.4 Correlation Analysis

The correlation coefficient of adjacent pixels can reflect the degree of diffusion of image pixels. The correlation between adjacent pixels of the original image is very high. To resist statistical attacks, the correlation between adjacent pixels of the cipher image must be reduced. The correlation calculation between pixels is shown in Eq. (13):

$$\begin{cases} E\ (x) = \frac{1}{N}\ \sum_{i=1}^{N} x_i \\ D\ (x) = \frac{1}{N}\ \sum_{i=1}^{N} (x_i - E\ (x))^2 \\ cov\ (x, y) = \frac{1}{N}\ \sum\limits_{i=1}^{N} ((x_i - E\ (x))\ (y_i - E\ (y))) \\ \rho_{xy} = \frac{cov(x,y)}{\sqrt{D_x} \times \sqrt{D_y}} \end{cases} \tag{13}$$

where x and y represent pixel values, $cov(x, y)$ represent the covariance, D(x) is the variance, E(x) is the mean, and ρ_{xy} is the correlation coefficient. 10000 pairs of pixels were randomly selected from the original image and the encrypted image, and the correlation between the original image and the encrypted image in the horizontal, vertical and diagonal directions was calculated, respectively. The correlation coefficients in all directions are given in Table 5. This encryption scheme breaks the correlation between adjacent pixels very well, and the encryption effect is some current encryption algorithms.

4.5 Local Shannon Entropy

The local Shannon entropy (LSE) of the image can better represent the image's randomness. Some known weaknesses of the global Shannon entropy can be overcome [35]. For an image **P**, randomly select k non-overlapping image blocks S_1, S_2, …, S_k with T_B pixels, the LSE can be defined as:

$$\overline{H_{k,\ T_B}}(\mathbf{P}) = \sum_{i=1}^{k} \frac{H(S_i)}{k} \tag{14}$$

TABLE 6
The LSE Scores of Different Cipher-Images

Test images	Original image	Local information entropy
Lena (256×256)	7.4153	7.9023
Boat (256×256)	7.1572	7.9027
Hill (256×256)	7.4460	7.9019
Face (256×256)	7.4248	7.9035

TABLE 7
NPCR and UACI Analysis With the Existing Methods for LENA Image in Size 256 × 256

Arithmetic	Ours	Ref.[31]	Ref.[32]	Ref.[33]	Ref.[34]
NPCR (%)	99.6063	99.5039	99.5063	99.4600	99.3011
UACI (%)	33.4477	31.6549	32.2367	37.6390	34.5755

TABLE 8
NPCR Randomness Test

Image	NPCR (%)	Theoretical NPCR critical value		
		$N^*_{0.001} = 99.5341\%$	$N^*_{0.01} = 99.5527$	$N^*_{0.05} = 99.5693\%$
		0.001-level	0.01-level	0.05-level
Lena	99.6063	pass	pass	pass
Boat	99.6096	pass	pass	pass
Face	99.6307	pass	pass	pass

where $H(S_i)$ is the Shannon entropy of image block S_i and can be defined as:

$$H(S_i) = -\sum_{i=1}^{I} P(I)\, log_2(P(I)) \tag{15}$$

where I is the total number of pixel values and $P(I)$ is the probability of Ith values.

We tested the local information entropy of the encrypted Lena (256 × 256), Face (256 × 256), Hill (256 × 256) and Boat (256 × 256) images. We set the parameters (k, T_B) = (30, 1936) and significance $\alpha = 0.05$, then the ideal LSE is 7.902469317 and an image is considered to pass the test if the obtained LSE falls into the interval (7.901901305, 7.903037329). The test results are shown in Table 6, the local information entropy of encrypted images almost all in the interval, this means that the proposed algorithm can encrypt images into cipher-images with high randomness.

4.6. Differential Attack Analysis

The differential attack involves changing any bit of the original image's pixel value and then comparing the differences between the encrypted images produced by the slightly altered images. NPCR and UACI are usually used to test the performance of image encryption algorithms against differential attack. Φ^{-1} is the inverse cumulative representative under the standard normal distribution density function. When the NPCR $>$ $NPCR^*_\alpha$, stands for NPCR pass the test. When the value of UACI is in the interval [$UACI^{*-}_\alpha$, $UACI^{*+}_\alpha$], it means UACI passes the randomness test. The ideal values of NPCR and UACI were 99.609375% and 33.463541%, respectively. Table 7 represents the NPCR and UACI of lean images under different algorithms. NPCR and UACI randomness tests are shown in Table 8 and Table 9. As can be seen from these tables, the NPCR and UACI of the encrypted images generated by our proposed encryption algorithm are very close to the ideal values. All the encrypted images have passed the critical tests of NPCR and UACI. So the algorithm can resist differential attack effectively.

TABLE 9
UACI Randomness Test

Image	UACI (%)	Theoretical UACI critical value		
		$UACI^{*-}_{0.001} = 33.1594\%$	$UACI^{*-}_{0.01} = 33.2255\%$	$UACI^{*-}_{0.05} = 33.2824\%$
		$UACI^{*+}_{0.001} = 33.7677\%$	$UACI^{*+}_{0.01} = 33.7016\%$	$UACI^{*+}_{0.05}33.6447\%$
		0.001-level	0.01-level	0.05-level
Lena	33.4477	pass	pass	pass
Boat	33.5173	pass	pass	pass
Face	33.5562	pass	pass	pass

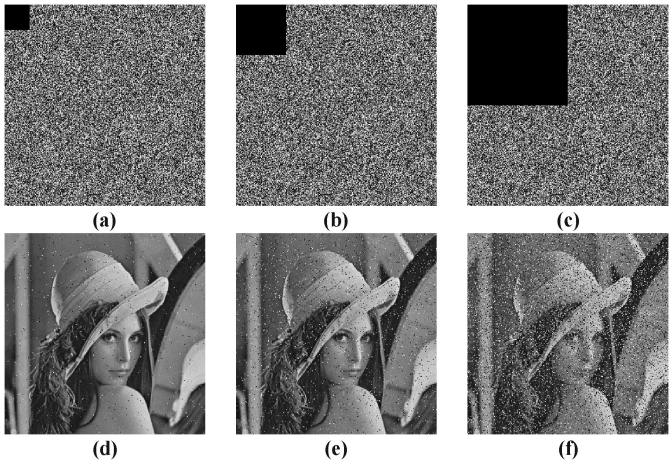


Fig. 10. Robustness against occlusion attack under low computing precision. (a) Encrypted image with 1/64 data loss. (b) Encrypted image with 1/16 data loss. (c) Encrypted image with 1/4 data loss. (d) Decrypted image with 1/64 data loss. (e) Decrypted image with 1/16 data loss. (f) Decrypted image with 1/4 data loss.

TABLE 10
The Performance Assessment Results of Robustness Against Occlusion Attack

Occlusion	Horizontal	Vertical	Diagonal	NPCR (%)	UACI (%)
0	0.9639	0.9292	0.9116	0	0
1/64	0.9019	0.8972	0.8654	1.5900	0.4490
1/16	0.7449	0.7579	0.7159	6.3187	1.7756
1/4	0.4106	0.4053	0.3754	25.0992	7.1369

4.7 Robustness Against Occlusion Attack

Occlusion attack analysis is to delete a part of the pixels in the encrypted image, after the original decrypted algorithm, compare and analyze the obtained decrypted image and the original image to see if the result can restore the image to the greatest extent. Because the encrypted image may also lose part of its data due to various reasons during the transmission process. Suppose a certain amount of data is lost in the cipher image, and the recovery ability of the decrypted algorithm is limited. In that case, the decrypted image of the cipher image after the loss of information cannot provide enough valid information to cause decrypted failure. As shown in Fig. 10(a), Fig. 10(b) and Fig. 10(c), Lena encryption images data are loss 1/64, 1/16 and 1/4, respectively and then decrypted. The resulting decrypted image is as shown in Fig. 10(d), Fig. 10(e) and Fig. 10(f). The performance assessment results of robustness against occlusion attack are shown in Table 10.

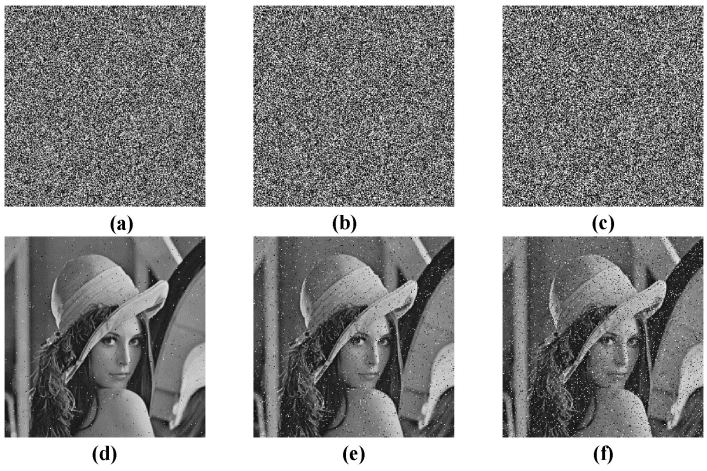


Fig. 11. Robustness against noise attack under low computing precision. (a) Encrypted image with 0.01 salt and pepper noise. (b) Encrypted image with 0.05 salt and pepper noise. (c) Encrypted image with 0.1 salt and pepper noise. (d) Decrypted image with 0.01 salt and pepper noise. (e) Decrypted image with 0.05 salt and pepper noise. (f) Decrypted image with 0.1 salt and pepper noise.

4.8 Robustness Analysis Against Noise

In the process of image transmission, some data will inevitably be affected by Gaussian noise and salt noise, and there is the possibility of loss. The influence of image encryption algorithm on data loss should have the immune ability, which is the image has anti-noise and anti-data loss ability in the transmission process. The anti-noise ability is also one of the standards to measure the performance of the encryption algorithm. We used pepper and salt noise with noise intensity of 0.01, 0.05 and 0.1 respectively to disturb the cipher, and then decrypted the interfered cipher. The decrypted image interfered with by noise is shown in Fig. 11. It can be seen from the figure that, despite noise interference, the decrypted image can still be recognized even if the noise intensity reaches 0.1, indicating that the algorithm has a certain noise resistance capability.

5. Conclusion

The algorithm proposed in this paper encrypts the image based on chaotic sequences and cross-diffusion of bits. The pseudo-random sequences generated by the 2D-LSCM system scrambles the position pixels of image, and the pseudo-random sequence generated by the logistic system diffuses the pixels of image. In the encryption scheme, the proposed scrambling method has an excellent effect. The encryption method that introduces information fusion technology and cross-diffusion has achieved good results in enhancing resistance to differential attack. The experimental results show that this algorithm performs well in various tests and is superior to the encryption algorithm compared. It can effectively resist exhaustive attacks, statistical analysis, and it can be used to protect image information.

References

[1] J. S. Teh, M. Alawida, and Y. C. Sii, “Implementation and practical problems of chaos-based cryptography revisited,” *J. Inf. Secur. Appl.*, vol. 50, Feb. 2020, Art. no. 102421.
[2] W. C. Qiu and S. J. Yan, “An image encryption algorithm based on the combination of low - dimensional chaos and high-dimensional chaos,” in *Proc. 3rd Int. Conf. Electron. Inf. Technol. Comput. Eng.*, Oct. 2019, pp. 684–687.
[3] W. J. Cao, Y. J. Mao, and Y. C. Zhou, “Designing a 2D infinite collapse map for image encryption,” *Signal Process.*, vol. 171, Jun. 2020, Art. no. 107457.
[4] H. R. Shakir, “An image encryption method based on selective AES coding of wavelet transform and chaotic pixel shuffling,” *Multimedia Tools Appl.*, vol. 78, no. 18, pp. 26073–26087, Jun. 2019.

[5] S. J. Xu, L. H. Wang, and J. Z. Wang, "A fast image encryption algorithm based on high-dimension chaotic system," in *Proc. IEEE 14th Int. Conf. Commun. Technol.*, Nov. 2012, pp. 829–835.
[6] X. Fu, B. Liu, Y. Xie, W. Li, and Y. Liu, "Image encryption-then-transmission using DNA encryption algorithm and the double chaos," *IEEE Photon. J.*, vol. 10, no. 3, Jun. 2018, Art. no. 3900515.
[7] H. Liu and X. Wang, "Color image encryption using spatial bit-level permutation and high-dimension chaotic system," *Opt. Commun.*, vol. 284, no. 16, pp. 3895–3903, Aug. 2011.
[8] J. Fridrich, "Image encryption based on chaotic maps," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. Comput. Cybern. Simul.*, vol. 2, Oct. 1997, pp. 1105–1110.
[9] Q. Zhang, L. Guo, and X. P. Wei, "Image encryption using DNA addition combining with chaotic maps," *Math. Comput. Modelling*, vol. 52, no. 11, pp. 2028–2035, Dec. 2010.
[10] Z. Y. Hua, B. X. Xu, and F. Jin, "Image encryption using josephus problem and filtering diffusion," *IEEE Access*, vol. 7, pp. 8660–8674, Jan. 2019.
[11] R. Enayatifar, "Chaos based image encryption using a hybrid genetic algorithm and a DNA sequence," *Opt. Lasers Eng.*, vol. 56, pp. 83–93, May 2014.
[12] Z. Y. Hua, B. X. Xu, and F. Jin, "2D logistic-sine-coupling map for image encryption," *Signal Process.*, vol. 149, pp. 148–161, Aug. 2018.
[13] Y. Sun, P. Yan, H. Zhang, and Y. Zhang, "A novel bit-level image encryption algorithm based on coarse-grained chaotic signals," in *Proc. IEEE 4th Int. Conf. Image, Vis. Comput.*, Xiamen, China, Jul. 2019, pp. 187–192.
[14] X. L. Chai, H. Wu, Z. Gan, and Y. Zhang, "An efficient approach for encrypting double color images into a visually meaningful cipher image using 2D compressive sensing," *Inf. Sci.*, ISSN 0020-0255, https://doi.org/10.1016/j.ins.2020.10.007. 2020.
[15] R. Vidhya and M. Brindha, "A novel conditional butterfly network topology based chaotic image encryption," *J. Inf. Secur. Appl.*, vol. 52, Jun. 2020, Art. no. 102484.
[16] X. L. Chai, H. Wu, Z. Gan, and Y. Zhang, "Hiding cipher-images generated by 2-D compressive sensing with a multi-embedding strategy." *Signal Process.*, vol. 171, Jun. 2020, Art. no. 1850124, https://doi:10.1016/j.sigpro.2020.107525.
[17] Y. Guo, S. Jing, Y. Zhou, X. Xu, and L. Wei, "An image encryption algorithm based on logistic-fibonacci cascade chaos and 3D bit scrambling," *IEEE Access*, vol. 8, pp. 9896–9912, Jan. 2020.
[18] S. M. Wadi and N. Zainal, "Decomposition by binary codes-based speedy image encryption algorithm for multiple applications," *IET Image Process.*, vol. 9, no. 5, pp. 413–423, May 2015.
[19] Y. Q. Zhang and X. Y. Wang, "A symmetric image encryption algorithm based on mixed linear–nonlinear coupled map lattice," *Inf. Sci.*, vol. 273, no. 273, pp. 329–351, Jul. 2014.
[20] H. Liu and X. Wang, "Color image encryption using spatial bit-level permutation and high-dimension chaotic system," *Opt. Commun.*, vol. 284, no. 16/17, pp. 3895–3903, Aug. 2011.
[21] R. Gopinath and M. Sowjanya, "Image encryption for color images using bit plane and edge map cryptography algorithm," *Proc. Int. J. Eng. Res. Technol.*, vol. 1, no. 8, pp. 1–4, Oct. 2012.
[22] F. Y. Li, H. B. Wu, and G. Zhou, "Robust real-time image encryption with aperiodic chaotic map and random-cycling bit shift," *J. Real-Time Image Process.*, vol. 16, no. 3, pp. 775–790, Jun. 2019.
[23] X. L. 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, Dec. 2015.
[24] A. Belazi, M. Talha, S. Kharbech, and W. Xiang, "Novel medical image encryption scheme based on chaos and DNA encoding," *IEEE Access*, vol. 7, pp. 36667–36681, Mar. 2019.
[25] X. L. Chai, H. Wu, Z. Gan, and Y. Zhang, "A color image cryptosystem based on dynamic DNA encryption and chaos," *Signal Process.*, vol. 155, pp. 44–62, Feb. 2019.
[26] X. Yang, "Application of a switching lorenz chaotic system in image encryption," *Packag. Eng.*, vol. 5, pp. 179–184, May 2018.
[27] Z. W. Tu *et al.*, "Color image encryption algorithm based on DNA sequences," *Comput. Eng.*, vol. 037, no. 010, pp. 1933–1939, Dec. 2015.
[28] Y. Liu, J. Wang, J. Fan, and L. Gong, "Image encryption algorithm based on chaotic system and dynamic S-boxes composed of DNA sequences," *Multimedia Tools Appl.*, vol. 75, no. 8, pp. 1–20, Jul. 2014.
[29] S. Ma, Y. Zhang, Z. Yang, J. Hu, and X. Lei, "A new plaintext-related image encryption scheme based on chaotic sequence," *IEEE Access*, vol. 7, pp. 30344–30360, Feb. 2019.
[30] G. D. Ye and X. L. Huang, "A feedback chaotic image encryption scheme based on both bit-level and pixel-level," *J. Vib. Control*, vol. 22, no. 5, pp. 1171–1180, Aug. 2015.
[31] S. Mozaffari, "Parallel image encryption with bit-plane decomposition and genetic algorithm," *Multimedia Tools Appl.*, vol. 77, no. 19, pp. 25799–25819, Oct. 2018.
[32] X. Lv, X. F. Liao, and B. Yang, "Bit-level plane image encryption based on coupled map lattice with time-varying delay," *Modern Phys. Lett. B*, vol. 32, no. 10, Apr. 2018, Art. no. 1850124.
[33] L. F. Liu, "A new simple one-dimensional chaotic map and its application for image encryption," *Multimedia Tools Appl.*, vol. 77, no. 16, pp. 21445–21462, Aug. 2018.
[34] F. Peng, S. S. Qiu, and M. Long, "An image encryption algorithm with parameters controlled by external keys," *J. South China Univ. Technol. (Natural Sci. Ed.)*, vol. 33, no. 7, pp. 20–23, Jul. 2015.
[35] Y. Wu and Y. Zhou, "Local shannon entropy measure with statistical tests for image randomness," *Inf. Sci.*, vol. 222, pp. 323–342, Oct. 2013.