IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 2, 2020, accepted October 15, 2020, date of publication October 29, 2020, date of current version November 10, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3034666

An Effective Image Encryption Method Based on Space Filling Curve and Plaintext-Related Josephus Traversal

YING NIU[1] AND XUNCAI ZHANG[2], (Member, IEEE)
[1]College of Architecture Environment Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China
[2]College of Electrical and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China

Corresponding author: Xuncai Zhang (zhangxuncai@pku.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 62072417 and Grant U1804262, and in part by the Henan Provincial Science and Technology Research Project under Grant 202102210177 and Grant 192102210134.

ABSTRACT To improve the effectiveness and security of image encryption, this article proposes an image encryption scheme based on a Y-index Space Filling Curve (SFC) and variable step Josephus traversal. The scheme adopts a confusion-diffusion structure. First, a novel Y-index SFC is designed. SFC is a method of continuously traversing each pixel of an image to encrypt the image, and images are confused by the Y-index SFC. Second, the random sequences generated by the chaotic map are taken as the starting point and step length of the Josephus traversal, and the pixels of the image are scrambled at a bit-level to change the values of the pixels. By changing the step length, the diversity of the Josephus traversal is increased. Finally, ciphertext feedback and chaotic sequence operations are used to further enhance the confusion and diffusion characteristics of the algorithm. By analyzing the experimental results and comparing them with the results of other image encryption algorithms, we show that this algorithm performs better and achieves higher security than other algorithms.

INDEX TERMS Image encryption, space filling curve, Josephus traversal, chaotic map.

I. INTRODUCTION

A digital image has a large amount of data, strong correlation between adjacent pixels, and obvious statistical features. Therefore, the encryption of digital images is much more difficult than the encryption of text messages. Common digital image encryption algorithms include chaotic encryption, graphic encryption, DNA computing encryption, and so on [1]–[3]. The implementations of these encryption methods fall into two categories: scrambling and diffusion. The scrambling process is designed to obfuscate the location of the data in the message and break the correlation of the data. The diffusion process is designed to replace the content of the data, thus masking the real information. Both scrambling and diffusion have certain disadvantages. For example, data scrambling does not change the overall statistical characteristics of the data in the information, and the data diffusion method has poor resistance to data loss in the decryption process. Therefore, scrambling and diffusion processes are often used in a mixture of encryption algorithms.

In an image encryption algorithm, the pseudorandom sequence generated by the chaotic system can be used as both scrambling data and diffusion data. Therefore, as a key generator, the chaotic system is widely used in encryption algorithms. A chaotic system is a very complex, irregular, nonrepeatable, and predictable nonlinear dynamic system with a high initial value of sensitivity. Consequently, an encryption algorithm based on a chaotic system has the advantage of being difficult to crack.

In recent years, many image encryption algorithms based on chaos theory have been proposed. These algorithms either improve existing chaotic systems or use low-dimensional, high-dimensional, or mixed chaotic systems to scramble and displace pixels in images. For example, in 2018, Hua *et al.* [4] proposed a new two-dimensional cosine map that has better ergodicity, more complex behavior, and larger chaotic range than the existing two-dimensional chaotic map. Natiq *et al.* [5] designed a new two-dimensional sine Henon transformation model (2D-sham) by using one-dimensional

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

sine map and two-dimensional Henon map. 2D-sham has high complexity and is highly sensitive to the initial values and control parameters. Vaidyanathan *et al.* [6] designed a new four-dimensional Hyperjerk chaotic system with two exponential nonlinear characteristics; analyzed the dynamic characteristics of the hyperchaotic system through equilibrium analysis, bifurcated graph, dissipation, Lyapunov index and other indicators; and built a simulation circuit to verify the feasibility of the system. Finally, the chaotic system is applied to random number generation, image encryption and sound steganography. In 2019, Farhan *et al.* [7] proposed a novel chaotic system, and the state of the chaotic attractor is used to scramble the rows and columns of the image. The results showed that the proposed encryption method has reliable performance, and the unpredictability of the chaotic attractor makes the encryption method very safe.

Some scholars combine existing chaotic systems with other methods to design new image encryption algorithms. For example, in 2016, Tong [8] designed a compressible image encryption algorithm by using a hyperchaotic system and combining it with a compression algorithm. This algorithm uses a hyperchaotic system to encrypt the image and image compression technology to compress the ciphertext, which can save storage space. In 2018, Ping *et al.* [9] proposed a two-point diffusion encryption method based on discrete Henon map. If there are multiple processing units, the algorithm can greatly accelerate the diffusion process. In 2019, we proposed an image encryption method combining the Space filling property of a Hilbert curve, the infinity of the H-geometric fractal, and the pseudorandomness of a hyperchaotic system [10]. In addition, through the use of pure mathematics and graphic geometry encryption methods, at present, many encryption algorithms are a combination of the bionic calculation method. For example, in 2019, Wang *et al.* [11] proposed a kind of image encryption algorithm based on chaos and DNA operations. First, the original image was scrambled. Second, the scrambled image was divided into four subimages of the same size, and these subimages were encoded through DNA rules and diffused, respectively. The diffusion rules were realized through DNA XOR operation. The algorithm can resist typical attacks, such as chosen plaintext attack, and has good security. In 2018, Mozaffari [12] proposed an encryption method based on a genetic algorithm and parallel bit plane decomposition. The algorithm converts the original gray image into a set of binary images via local binary mode and bit plane decomposition. The permutation and substitution steps are then performed using crossover and mutation operations through the genetic algorithm (GA). Finally, the scrambled bit planes are combined to obtain an encrypted image.

The work of this article is to combine the chaotic system with the Space Filling Curve (SFC) and propose a new encryption algorithm. Many works have applied SFC curves to image encryption [13]–[16]. For example, in 2012, Gaurav *et al.* [14] proposed a new encryption scheme with three different operation modes based on a dual SFC and fractional wavelet transform (FrWT). In the same year, Bhatnagar proposed a selective image encryption method based on the decomposition of pixels and singular values of interest [17]. In this method, a saw-tooth filling curve was used to scramble the image, and chaotic map and singular value decomposition were used for diffusion to obtain selective encryption of the image. In 2014, Sivakumar and Venkatesan [18] proposed a dynamic SFC scanning method that uses scanning, circular shift and transposing methods to scramble images. In 2019, Murali and Sankaradass [19] proposed a fast encryption algorithm based on a SFC that uses SFC to continuously traverse each pixel of the image and disrupt its position. However, encryption schemes that use only simple scrambling methods are vulnerable to brute force attacks.

Compared with pixel-level scrambling, pixel values and pixel positions are affected by bit-level scrambling. In 2020, Shahna proposed a pixel-level and bit-level image encryption scheme based on chaotic map [20]. In this method, the Hilbert curve and Henon chaotic map are used to obtain double scrambling, which further increases the scrambling effect. However, for the existing scan curve, the image correlation after one scan, or even the three-scan scrambling is still relatively high.

Therefore, this article proposes a novel image encryption method by using Y-index SFC, Josephus traversal, and a chaotic map to obtain double scrambling of the pixel-level and bit-level. First, we present a novel Y-index SFC and design a pixel-level scrambling method based on the Y-index SFC. Second, the Josephus traversal function is improved by adding a variable step length parameter that expands the diversity of the Josephus traversal function. By using the hash value generated by the plain image as the initial parameter of the chaotic map, the key is associated with the plain image, and the sequence generated by the chaotic map is used as the starting point and initial step of Josephus traversal to scramble the pixels at a bit-level. Finally, the bidirectional feedback of ciphertext is realized by combining with chaotic sequences.

The rest of this article is organized as follows: In section 2, we briefly introduce the nonlinear chaotic map and the Josephus traversal. Section 3 describes the proposed Y-index SFC. Section 4 describes the proposed encryption scheme in detail. A comparison between the proposed algorithm and other algorithms proposed in the literature is presented in Section 5, and in the last section, conclusions are given.

II. 2D-LSCM MAP AND JOSEPHUS TRAVERSAL

A. 2D-LSCM MAP

Two-dimensional logic sine coupled map is derived from one-dimensional logic map [21] and sine map [22].

The definitions of logical map and sine map are shown in formulas (1) and (2), respectively.

$$x_{i+1} = 4\gamma x_i (1 - x_i), \tag{1}$$

$$x_{i+1} = \mu \sin(\pi x_i), \tag{2}$$

where the parameter $\gamma, \mu \in (0, 1)$.

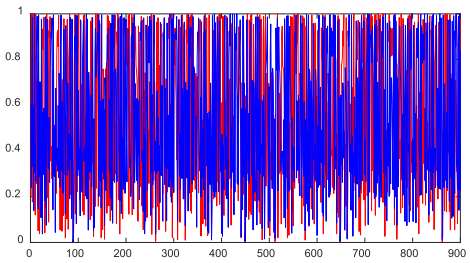


FIGURE 1. 2D-LSCM map simulation figure.

Logic map or sine map alone is used for the encryption operation, and the security of the encryption algorithm is not high due to its lack of ergodicity. To obtain a pseudorandom sequence more in line with the requirements of the encryption algorithm, Hua coupled logical map and sine map to obtain a new chaotic map, which is called 2D-LSCM [4]. The definition of 2D-LSCM is as follows:

$$\begin{cases} x_{i+1} = \sin\left(\pi\left(4\sigma x_i\left(1-x_i\right)+\left(1-\sigma\right)\sin\left(\pi y_i\right)\right)\right) \\ y_{i+1} = \sin\left(\pi\left(4\sigma y_i\left(1-y_i\right)+\left(1-\sigma\right)\sin\left(\pi x_i\right)\right)\right), \end{cases} \tag{3}$$

where the parameter $\sigma \in (0, 1)$. 2D-LSCM first couples logical map and sine map, then sine transformation is performed on the coupling results, and finally, the dimension is extended from one dimension to two dimensions. This process can fully mix the complexity of logical and sine maps to obtain a complex chaotic behavior. When the control parameter is within an appropriate range of values, the two Liapunov indexes in the 2D-LSCM system are positive and large. Therefore, the system has good chaotic properties and complex traversal orbits. The 2D-LSCM significantly improves the characteristics of chaotic systems, and its ergodicity and sensitivity are strong, which can provide a larger key space and higher security for encryption algorithms. Figure 1 is a two-dimensional time series of 2D-LSCM. It can be seen from the figure that the two time series of 2D-LSCM are uniformly distributed in the interval (0, 1).

B. JOSEPHUS TRAVERSAL

The Josephus problem is a cyclic traversal problem that is described as: there are n elements in a circle; traverse the first element in order, delete the t'th element, and continue the operation from the $(t+1)$'th element until the last element is selected from the circle. Finally, according to the order in which the elements are removed from the circle, we can obtain a sequence, namely, the Josephus sequence. The Josephus traversal is expressed as a function, namely, $J = f(n, t)$, where n is the total number of elements, t is the step length, and J is the Josephus sequence obtained. To extend the Josephus traversal, a new rule can be introduced into the Josephus function, that is, the starting point r can be added on the basis of the original rule, and the Josephus function can be extended to $J = f(n, r, t)$. This method greatly increases the diversity of the Josephus traversal. For example, the solution to the function $f(8, 2, 3)$ is to make the elements 1, 2, 3, 4, 5,6, 7, 8 form a circle, and starting with the second element, removing the third element in turn, we can obtain a sequence: 2, 5, 8, 4, 1, 7, 3, and 6. Figure 2 shows the generation of the Josephus sequence and its use in bit scrambling.

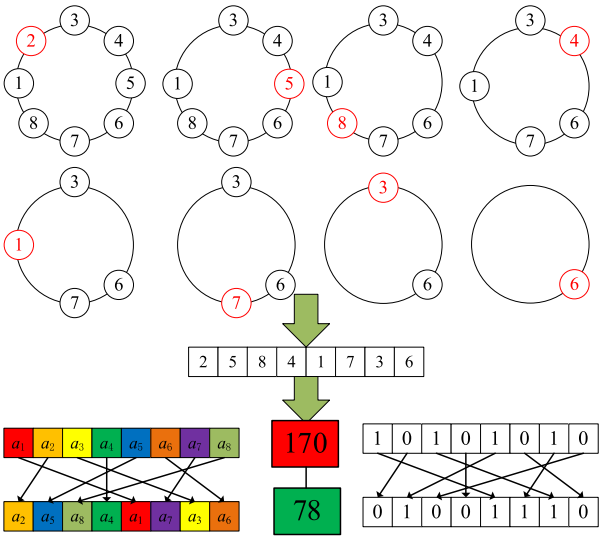


FIGURE 2. The generation of Josephus sequence and its use in bit scrambling.

To further increase the diversity of the Josephus traversal, we let the step length of Josephus traversal dynamically updated; when an element is removed, the step length is updated. Let $t=t+k$, where k is the increment of step length. Then, the Josephus function is further extended to $J=f(n, r, t, k)$. For example, the function $J=f(8, 2, 3, 2)$ produces Josephus sequences of 2, 5, 3, 6, 7, 1, 4, and 8.

III. SPACE FILLING CURVE

A SFC refers to a kind of function curve that includes the whole two-dimensional or even multidimensional space within a one-dimensional curve. According to different arrangement rules, different SFCs can be obtained. A SFC can continuously access all the pixels in the image at one time and carry out linear sorting or coding for each pixel, shuffling the pixel position of the original image, and reorganizing the pixels to produce a new pixel image. Because of its fast speed, SFC has been widely used in multidimensional indexing, geographical research, and other fields. The image can be quickly scrambled by scanning with a SFC. Several common SCAN modes are shown in Figure 3. It is very convenient to use the SCAN mode to disrupt the position of pixel, but some SCAN modes also have certain defects, as shown in Figure 3(a), 3(b), and 3(c). Once scanned, multiple adjacent elements will still be adjacent. If the number of scans is too small, the ability to break the correlation between adjacent pixels will be poor.

Therefore, according to the characteristics of the image, we propose a novel space filling curve—-Y-index. The Y-index is a very simple way to scan an image because it visits every point in a two-dimensional space and never maintains the same direction for more than three consecutive pixels. If the

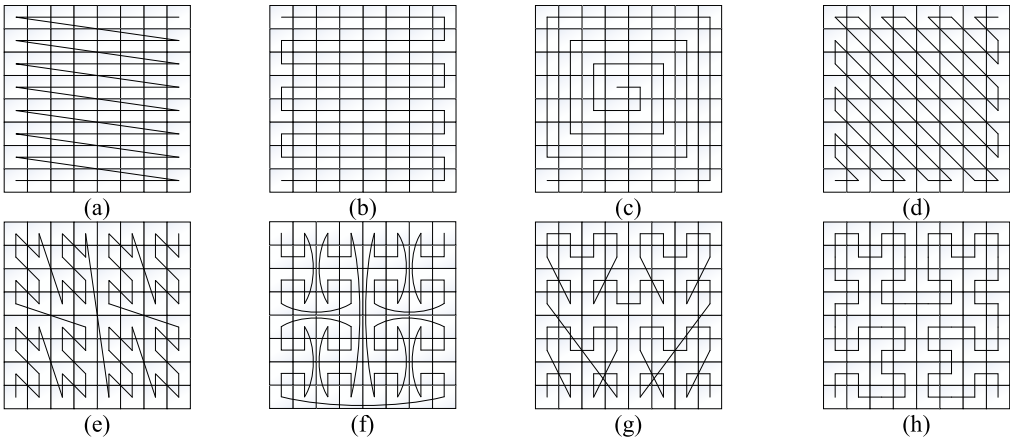


FIGURE 3. Common SCAN modes. (a) Row SCAN, (b) Row-Prime SCAN, (c) Spiral SCAN, (d) Cantor SCAN, (e) Peano SCAN, (f) Gray SCAN, (g) U-Index SCAN, (h) Hilbert SCAN.

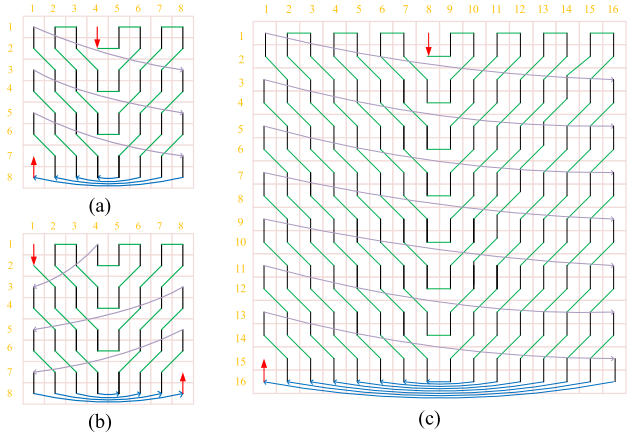


FIGURE 4. Schematic diagram of Y-index SFC scanning.

Y-index is used to scan continuously more than two times, the probability of the original adjacent pixels being adjacent again is close to 0. Figure 4 shows a schematic of the Y-index SFC. Figure 4(a) is an 8*8 schematic diagram, in which scanning is carried out along the Y-index from the middle; Figure 4(b) is a schematic diagram of curve scanning starting from the upper left corner, and Figure 4(c) is a schematic diagram of curve scanning of 16*16. From the figure can we see that the scrambling degree of the Y-index is significantly better than the scanning mode shown in Figure 3. In the next section, we verify the advantages of the Y-index in image scrambling through correlation analysis.

IV. ENCRYPTION SCHEME DESIGN

We assume that the size of the plain image P is $M \times N$ and that $P_{i,j}$ represents the pixel value in the ith row and jth column, which satisfies $1 \leq i \leq M$, $1 \leq j \leq N$.

A. CONTROL PARAMETERS AND INITIAL VALUES OF CHAOTIC SYSTEMS

The third-generation secure hashing algorithm (SHA-3) can convert arbitrary length character information into hashes of the same length. The key generated by the hash value, even if the original image has a 1-bit difference, and the hash value generated by the SHA-3 will be completely different, corresponding to different encryption keys. Associating the original image information with the hash function can not only generate a large key space and enhance its ability to resist exhaustive attack, but the small change of plaintext can also spread to the whole cipher image, which can be widely used for image encryption. After the SHA-3 (256) operation, the original image generates a 256-bit hash value. The hash value is converted into binary and used as the key K to generate the initial value of the 2D-LSCM system. The resulting initial value has the advantage of randomness.

K can be divided into 32 bytes, represented as: k_1, k_2..., k_{32}. The initial values of the two sets of chaotic maps are calculated by formulas (4) and (5).

$$\begin{cases} x_0^1 = \dfrac{1}{256} mod((k_1 \oplus k_2 \oplus k_3 \oplus k_4) \\ \qquad + k_5 + k_6 + k_7 + k_8, 256) + x_0^{'} \\ y_0^1 = \dfrac{1}{256} mod((k_9 \oplus k_{10} \oplus k_{11} \oplus k_{12}) \\ \qquad + k_{13} + k_{14} + k_{15} + k_{16}, 256) + y_0^{'} \\ \sigma^1 = mod\left(\sum_{i=205}^{256} K_i * 2^{-i} + \sigma^{'}, 1\right), \end{cases} \tag{4}$$

$$\begin{cases} x_0^2 = \dfrac{1}{256} mod((k_{17} \oplus k_{18} \oplus k_{19} \oplus k_{20}) \\ \qquad + k_{21} + k_{22} + k_{23} + k_{24}, 256) + x_0^{'} \\ y_0^2 = \dfrac{1}{256} mod((k_{25} \oplus k_{26} \oplus k_{27} \oplus k_{28}) \\ \qquad + k_{29} + k_{30} + k_{31} + k_{32}, 256) + y_0^{'} \\ \sigma^2 = mod\left(\sum_{i=1}^{52} K_i * 2^{-i} + \sigma^{'}, 1\right), \end{cases} \tag{5}$$

where x_0', y_0', and σ' are the given initial values.

B. PIXEL SCRAMBLING

Different from the traditional scrambling operation, the scrambling proposed in this article includes pixel scrambling and bit scrambling. Double scrambling can not only break the correlation of adjacent pixels but also change

TABLE 1. Correlation analysis of Scrambled images of the Y-Index, Hilbert, and Square-Wave curves.

SFC		Level	Vertical	Positive diagonal	Antidiagonal
1 round	Hilbert	0.9481157	0.3272188	0.3319006	0.3401213
	Square-wave	0.5737576	0.9097830	0.5502397	0.55267017
	Y-index	**0.8992039**	**0.1566772**	**0.1607419**	**0.1532096**
2 rounds	Hilbert	0.6587059	0.2133864	0.1757162	0.1749927
	Square-wave	0.3806944	0.1862852	0.6193793	0.8327900
	Y-index	**0.1190782**	**0.0054943**	**0.0262753**	**0.0017950**

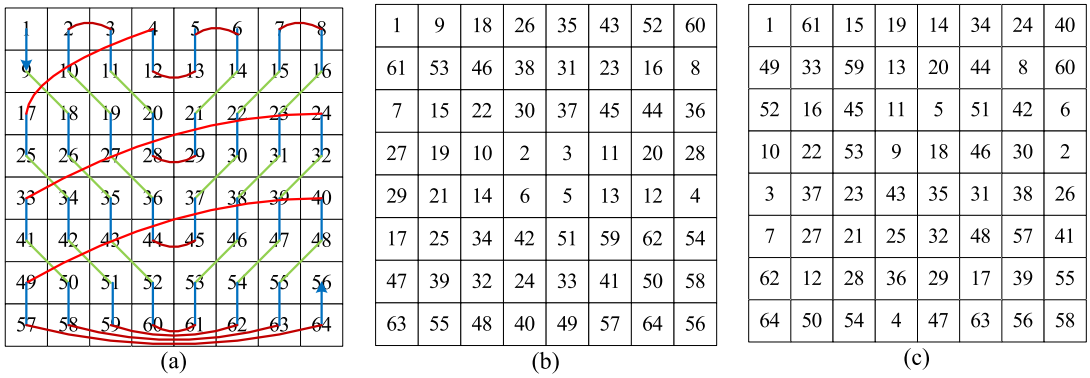


FIGURE 5. Scrambling schematic diagram using the Y-shaped curve.

the pixel value. To obtain pixel scrambling, we adopt a Y-index SFC. As shown in Figure 5(a), in an 8*8 square grid, the Y-index SFC scanning path starting from the upper right cell is a schematic.

Based on the scanning mode, the data matrix shown in Figure 5(a) is scanned, and the scrambled data matrix shown in Figure 5(b) was obtained. The scrambled data matrix shown in Figure 5(b) is scanned again to obtain the data matrix shown in Figure 5(c). Based on this scanning mode, the Lena image shown in Figure 6(a) is scrambled for the first scan. The scrambling results are shown in Figure 6(b), and the second scan results are shown in Figure 6(c). At the same time, the Hilbert curve and Square-wave curve are used to scan and scramble the Lena image, as shown in Figure 6(a). The results of the two scans of the Hilbert curve are shown in Figure 6(d) and 6(e), respectively, and the results of the two scans of the Square-wave curve are shown in Figure 6(f) and 6(h), respectively. Moreover, correlation analysis is carried out on the images after scrambling the three curves (see Section V(E) for the concept and calculation formula for the correlation), and the analysis results are shown in Table 1. The results show that in terms of image scrambling, the Y-index is significantly better than the Hilbert curve and the Square-Wave curve.

C. BIT SCRAMBLING

Pixel position scrambling breaks the correlation between adjacent pixels but does not change the pixel values, so it is not effective against statistical attacks. In this article, Josephus traversal is used for bit scrambling, which can effectively change the pixel value and further resist statistical attacks. This method combines Josephus traversal with a chaotic sequence and uses Josephus traversal to scramble the binary bits of each pixel by taking the pseudorandom sequence generated by the chaotic map as the parameter of the Josephus traversal. For Josephus traversal of each pixel, we use different starting points and step lengths, and the step lengths are automatically updated during the traversal. The elements of the pseudorandom sequences LS and LT generated by the chaotic map are processed according to formula (6) and converted into matrixes with the same size as the image, denoted as sequence matrixes $\boldsymbol{MS}$ and $\boldsymbol{MT}$, which are used as the starting point and step length of Josephus traversal, respectively. Figure 7 is an example of Josephus traversal, where the step length increment k=1. Figure 7(a) is a plaintext matrix, Figure 7(b) is the starting point matrix $\boldsymbol{MS}$, Figure 7(c) is the initial step length matrix $\boldsymbol{MT}$, and Figure 7(d) is the matrix after using Josephus to traverse Figure 7(a).

$$\begin{cases} ls_i^{'} = mod\left(floor\left(10^{15} \times ls_i\right), 8\right) + 1 \\ lt_i^{'} = mod\left(floor\left(10^{15} \times lt_i\right), 8\right) + 1 \end{cases} \tag{6}$$

D. PIXEL DIFFUSION

Ciphertext diffusion makes small changes in plaintext spread to the whole ciphertext, thus disrupting the relationship between plaintext and cipher images and effectively resisting the choice of cryptological attack means, such as plaintext. Combining a chaotic sequence and changing the current pixel value by using the previous pixel value, a small amount of plain image changes can be effectively propagated to the

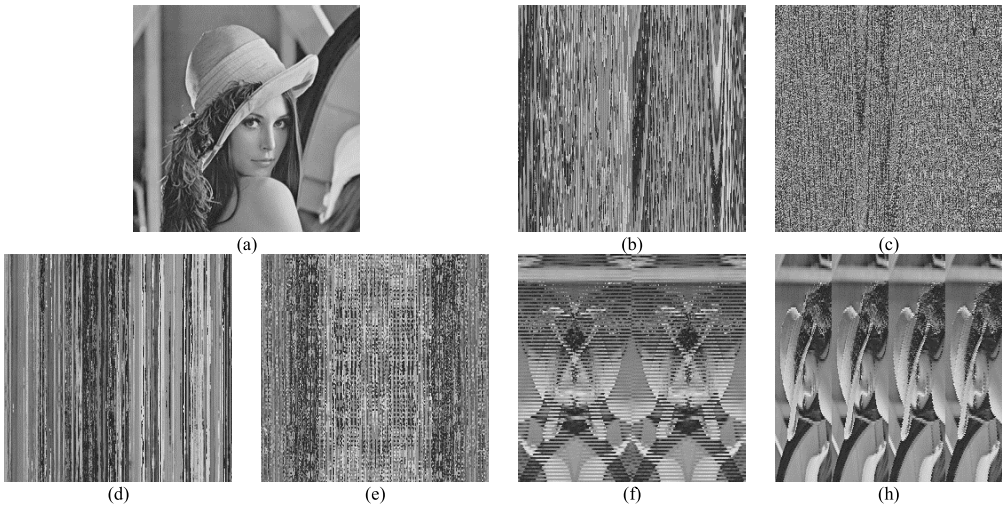


FIGURE 6. Simulation results of Pixel Scrambling. (a) The plain image of Lena; (b) Scrambled image with the Y-index for the first scan; (c) Scrambled image with the Y-index for the second scan; (d) Scrambled image with the Hilbert curve for the first scan; (d) Scrambled image with the Hilbert curve for the second scan; (e) Scrambled image with the Square-wave curve for the first scan; and (f) Scrambled image with the Square-wave for the second scan.

161 10100001	85 01010101	98 01100010	51 00110011
71 01000111	133 10000101	5 0000101	210 11010010
40 00101000	66 01000010	190 10111110	170 10101010
15 00001111	240 11110000	204 11001100	30 00011110

(a)

1	4	1	6
2	2	3	7
4	5	1	5
8	3	8	3

(b)

7	7	4	5
6	8	3	8
4	1	8	6
1	2	5	2

(c)

224 11100000	178 10110010	35 00100011	75 01001011
204 11001100	44 00101100	65 01000001	147 10010011
9 00001001	12 00001100	183 10110111	201 11001001
154 10011010	135 10000111	75 01001011	92 01011100

(d)

FIGURE 7. An example using Josephus scrambling. (a) plaintext matrix, (b)MS matrix, (c)MT matrix, (d) Josephus Traversal matrix.

whole encrypted image. The diffusion process includes forward diffusion and reverse diffusion. Given the initial chaotic value, the iterative chaotic map generates two sequences, LU and LV, with length $M\times N$. The sequence elements are processed according to formula (7) and converted into matrix form with size $M\times N$, denoted as sequence matrix $\boldsymbol{MU}$ and $\boldsymbol{MV}$.

$$\begin{cases} lu_i^{'} = mod\left(floor\left(10^{15}\times lu_i\right), 256\right) \\ lv_i^{'} = mod\left(floor\left(10^{15}\times lv_i\right), 256\right). \end{cases} \tag{7}$$

For the given image P, sequence matrix $\boldsymbol{MU}$, and sequence matrix $\boldsymbol{MV}$, the image matrix after forward diffusion is denoted as $\boldsymbol{C}$, and the matrix after reverse diffusion is denoted as $\boldsymbol{D}$. The forward diffusion process is shown in Formula (8):

$$C_{i,j} = \begin{cases} C_{i,j} \oplus P_{M,N} \oplus MU_{i,j} & for\ i = 1, j = 1; \\ C_{i-1,N} \oplus P_{i,j} \oplus MU_{i,j} & for\ i \neq 1, j = 1; \\ C_{i,j-1} \oplus P_{i,j} \oplus MU_{i,j} & other; \end{cases} \tag{8}$$

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

The reverse diffusion process is shown in Formula (9):

$$D_{i,j} = \begin{cases} \left(D_{i,j} + C_{1,1} + MV_{i,j}\right) mod 256 & for\ i = M,\ j = N; \\ \left(D_{i-1,N} + C_{i,j} + MV_{i,j}\right) mod 256 & for\ i \neq 1,\ j = 1; \\ \left(D_{i,j-1} + C_{i,j} + MV_{i,j}\right) mod 256 & other; \end{cases} \tag{9}$$

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

Figure 8 shows an example of using the pixel diffusion. Figure 8(a) shows the plaintext matrix and the ciphertext matrix, Figure 8(b) shows the plain image of Lena and the cipher image, and Figure 8(c) shows the histogram corresponding to Figure 8(b).

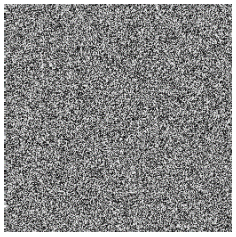
E. ENCRYPTION PROCESS

This algorithm contains three main operations: first, in the pixel position scrambling, the index composed of Y-index SFC is used to scramble the image pixel position globally. Second, the bit position scrambling; the bits of each pixel are traversed through Josephus traversal to achieve the replacement and encryption of pixel values. Finally, the pixels are further diffused through ciphertext feedback.

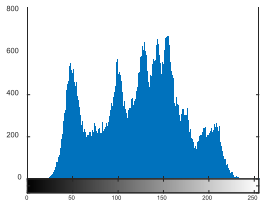
1	9	17	25	33	41	49	57
2	10	18	26	34	42	50	58
3	11	19	27	35	43	51	59
4	12	20	28	36	44	52	60
5	13	21	29	37	45	53	61
6	14	22	30	38	46	54	62
7	15	23	31	39	47	55	63
8	16	24	32	40	48	56	64

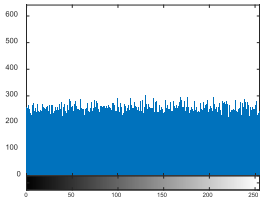
111	206	145	173	110	225	221	176
226	42	158	220	146	232	247	97
3	249	189	86	109	126	127	246
47	226	230	243	63	161	166	202
17	215	98	177	46	176	30	117
101	87	129	135	214	27	60	145
172	183	110	25	210	12	183	168
203	18	210	63	195	53	102	110

(a)



(b)





(c)

FIGURE 8. Pixel diffusion. (a) A 10*10 data matrix and its diffusion result; (b) the original and diffusion images; (c) the histograms image in (b).

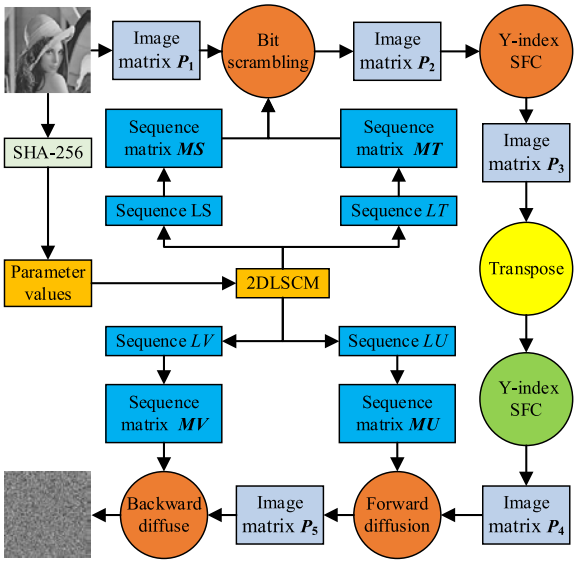


FIGURE 9. Flow chart of encryption process.

The encryption flow chart is shown in Figure 9, and the specific steps are as follows:

Step 1: Transform grayscale image P into a 2-dimensional image matrix $\boldsymbol{P_1}$ with size $M \times N$;

Step 2: Using the SHA-3 algorithm, calculate the hash value of image matrix $\boldsymbol{P_1}$ to obtain the 256-bit binary sequence K; According to formula (4) and formula (5), the parameter values of two sets of chaotic maps are obtained.

Step 3: According to the initial value obtained by formula (4), the chaotic system is iterated to obtain the chaotic sequences LS and LT. According to formula (6), the chaotic sequences LS and LT are processed and reconstructed into matrix form to obtain the sequence matrices $\boldsymbol{MS}$ and $\boldsymbol{MT}$, respectively. Take $\boldsymbol{MS}$ as the starting point matrix and $\boldsymbol{MT}$ as the initial step length matrix and use Josephus to traverse the binary bits of each pixel to obtain the image matrix $\boldsymbol{P_2}$.

Step 4: Scrambling image matrix $\boldsymbol{P_2}$ with the Y-index SFC for the first time to obtain image matrix $\boldsymbol{P_3}$;

Step 5: Transpose the image matrix $\boldsymbol{P_3}$ and use the Y-index SFC to perform a second scrambling operation to obtain the image matrix $\boldsymbol{P_4}$.

Step 6: According to the initial value obtained by formula (5), the chaotic system is iterated to obtain the chaotic sequences LU and LV. According to formula (7), the elements of sequences LU and LV are processed, and the matrix form is reconstructed again to obtain the two sequence matrices $\boldsymbol{MU}$ and $\boldsymbol{MV}$, respectively. Using the sequence matrix $\boldsymbol{MU}$ to carry out forward diffusion of the image, according to formula (8), the image matrix $\boldsymbol{P_5}$ is obtained

Step 7: Use sequence matrix $\boldsymbol{MV}$ to backward diffuse image matrix $\boldsymbol{P_5}$ according to formula (9) to obtain cipher images.

The decryption process of this algorithm is the inverse process in the encryption process, which is not described in this article.

V. ENCRYPTION RESULTS AND SECURITY ANALYSIS

To verify the feasibility and effectiveness of the method, the *Matlab2018a* tool was used to simulate the encryption algorithm. Given the initial value $x_0' = y_0' = \sigma' = 0.1$, this algorithm is used to encrypt the commonly used 256*256 *Lena, Baboon, Boat, Pepper* and *Elaine* grayscale images. The original images, cipher images, and decrypted images are shown in Figure 10. Clearly, the decrypted image is the same as the original image, without distortion, noise, or data loss. The cipher image has completely lost the features of the original image, and the original image cannot be identified from the cipher image, which indicates that the encryption algorithm has a good effect.

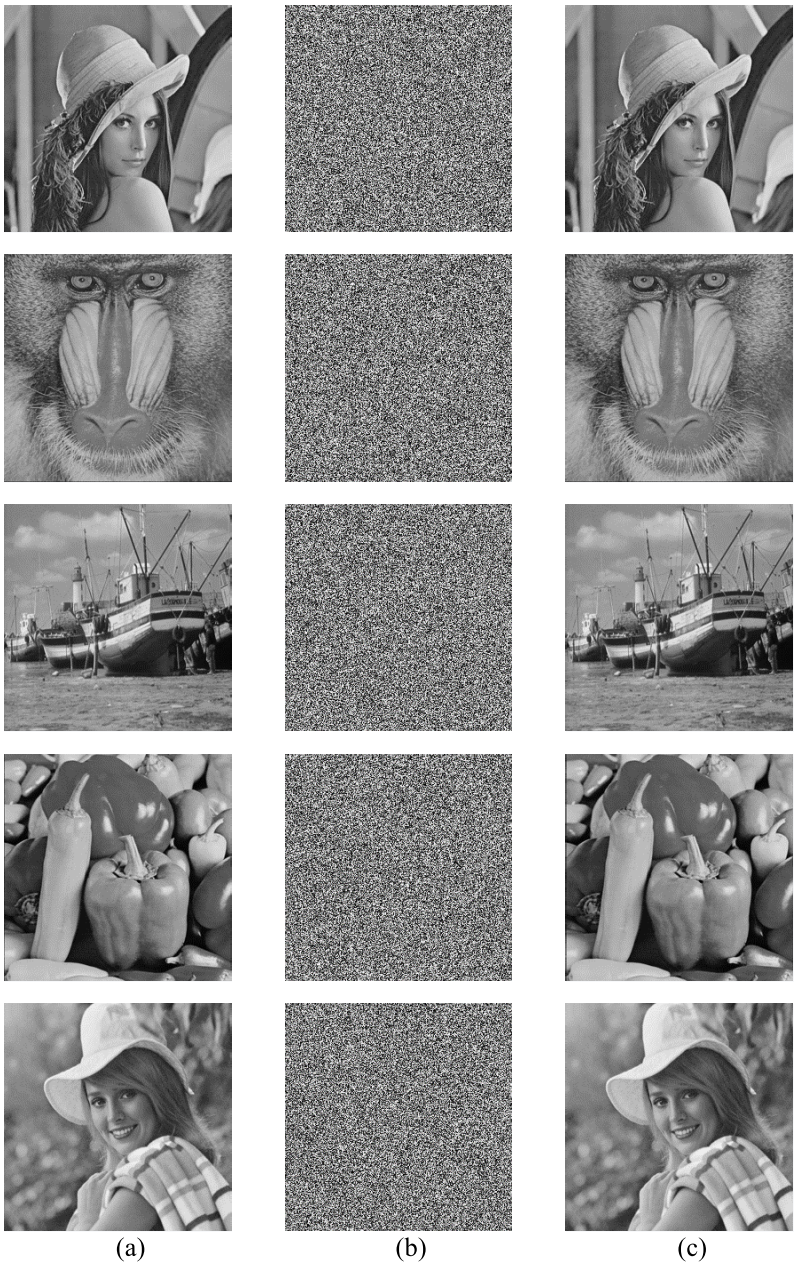


FIGURE 10. Experimental results. (a) The original images, (b) the cipher images, and (c) the decryption images.

To measure the security of the encryption algorithm, many evaluation indexes are used, including key security analysis, pixel analysis, and antijamming analysis. Among them, key analysis mainly analyzes the security of encryption algorithm keys, pixel analysis mainly analyzes the mathematical statistical relationship among pixels, and anti-interference analysis mainly analyzes the degree of the encryption algorithm's recovery of the image information when the cipher image encounters noise damage or data loss in the process of transmission. Below, we use these analysis indicators to analyze the security of the encryption algorithm proposed in this article.

A. KEY SPACE ANALYSIS

Key space is considered an important feature of any cryptographic system. The key space should be large enough to resist brute force attacks. The encryption algorithm's key is generated by the SHA-3(256) function, which converts data of an arbitrary length into a set of fixed-length binary sequences with an algorithmic complexity of 2^{128}. In addition, the encryption algorithm uses three initial values of the chaotic system, each with a key space of 10^{15}. According to the discussion in literature [23], the key space of an image encryption algorithm based on chaos should be greater than 2^{100} to resist brute force attacks. Therefore, the key space of

TABLE 2. Encryption key sensitivity (%) analysis.

The parameter value	NPCR	UACI
$x_0'+10^{-10}$	99.6002	33.4321
$y_0'+10^{-10}$	99.6368	33.4287
$\sigma'+10^{-10}$	99.6277	33.5921

the encryption algorithm is large enough to resist exhaustive attacks on the key.

B. KEY SENSITIVITY ANALYSIS

A high degree of key sensitivity is an important characteristic of any good encryption system. If the key is extremely sensitive to minor changes: (1) in the encryption process, the cipher images formed before and after the minor changes in the key will be greatly different, and the wrong ciphertext will not be able to restore the effective information through the correct key; (2) in the decryption process, the decryption images formed before and after minor changes in the key will have great differences. If the key is destroyed, the original information cannot be decrypted.

The NPCR (pixel change rate) and UACI (normalized change intensity) are commonly used to measure the sensitivity of the key [24]. The NPCR can be used to calculate the percentage of pixel changes, and when the detection result of the NPCR is close to 100%, then the pixels in the two images are greatly different. The UACI can adequately detect the difference of the samples as a supplementary explanation. The maximum theoretical value of the NPCR is 100%, and the ideal value of the UACI is 33.4635%. The calculations of the NPCR and UACI are shown in formulas (10)-(11).

$$NPCR(C_1, C_2) = \frac{\sum_{i,j} D(i,j)}{M \times N} \times 100\% \tag{10}$$

$$UACI(C_1, C_2) = \frac{\sum_{i,j} |C_1(i,j) - C_2(i,j)|}{255 \times M \times N} \times 100\% \tag{11}$$

C_1 and C_2 represent two cipher images, for pixel (i, j); if $C_1(i,j) \neq C_2(i,j)$, then $D(i,j) = 1$; otherwise, the $D(i,j) = 0$.

Taking Cameraman as an example, this article uses the sensitivity of the initial value of the chaotic system as the method to analyze the security of the key. The cipher images after each initial value of the chaotic system are increased by 10^{-10}, and the cipher images before the increase are compared and analyzed. The results are shown in Table 2. Table 2 shows that both the NPCR and UACI are close to their ideal values. The results show that the key generated by the algorithm has good encryption sensitivity.

In the decryption process, the NPCR and UACI can also be used to reflect the difference between the decrypted images. While the NPCR and UACI are used to measure the difference of decrypted images, the MSE (mean square deviation) and PSNR (peak signal-to-noise ratio) are added as indicators to detect the visual difference between the decrypted images. The calculation methods of the MSE and PSNR are shown in formula (12) and formula (13), respectively.

TABLE 3. Decryption key sensitivity (%) analysis.

The parameter value	NPCR	UACI	MSE	PSNR
$x_0'+10^{-10}$	99.6383	34.7546	11759.7	7.4269
$y_0'+10^{-10}$	99.6139	34.6219	11689.6	7.4529
$\sigma'+10^{-10}$	99.6002	34.7856	11740.9	7.4337

$$MSE = \frac{1}{M \times N} \sum_{i=1}^{N} \sum_{j=1}^{M} |P_1(i,j) - P_2(i,j)|^2, \tag{12}$$

$$PSNR = 20log_{10}\left(\frac{255}{\sqrt{MSE}}\right). \tag{13}$$

When the calculated MSE value is greater than or equal to 30 or the PSNR value is smaller, the difference between the two images is more obvious. The cipher images were decrypted and compared using the correct key and the slightly changed key. The results of the NPCR, UACI, MSE, and PSNR are shown in Table 3. These results show that the decryption process of the encryption algorithm is extremely sensitive to the key, and even a small change of the key will result in a large change in the decrypted image. Therefore, the encryption algorithm has strong key sensitivity, enough to resist an attack on the key.

C. HISTOGRAM ANALYSIS

The statistical information of the image can reflect the distribution law of the gray value of the original image to a certain extent. When the encryption algorithm only performs a pixel scrambling operation and no pixel displacement operation, the histogram distribution of the pixels remains unchanged. The purpose of bit substitution and pixel diffusion in this article is to defend against pixel value statistics attacks. A good encryption algorithm can make the statistical characteristics of the pixel values approach a uniform distribution by substitution, and the more even the distribution of the pixel values is, the better the security of the encryption algorithm will be. Figure 11 shows the histograms of the plain image and its corresponding cipher image. Through comparison, it can be seen that the cipher image is evenly distributed within the statistical range of the histogram, which breaks the statistical rule of the histogram of the original image.

The distribution of pixel histograms can be measured by χ^2, and χ^2 is calculated by formula (14), where $hist_i(i = 0, 2, 3, \ldots, 255)$ represents the distribution number of pixel values in an image in a range of values.

$$\chi^2 = \frac{1}{256} \sum_{i=0}^{255} (hist_i - \frac{1}{256} \sum_{i=0}^{255} hist_i)^2 \tag{14}$$

Given a significant level $P\{\chi^2 \geq \chi_\alpha^2(n-1)\} = \alpha$, $\chi^2 < \chi_\alpha^2(n-1)$ meets the desired condition. When $\alpha = 0.01, 0.05$, and 0.1, $\chi_{0.01}^2(255) = 310.45739$, $\chi_{0.05}^2(255) = 293.24783$, and $\chi_{0.1}^2(255) = 284.33591$, respectively. The statistics of the chi-square distribution in the original and

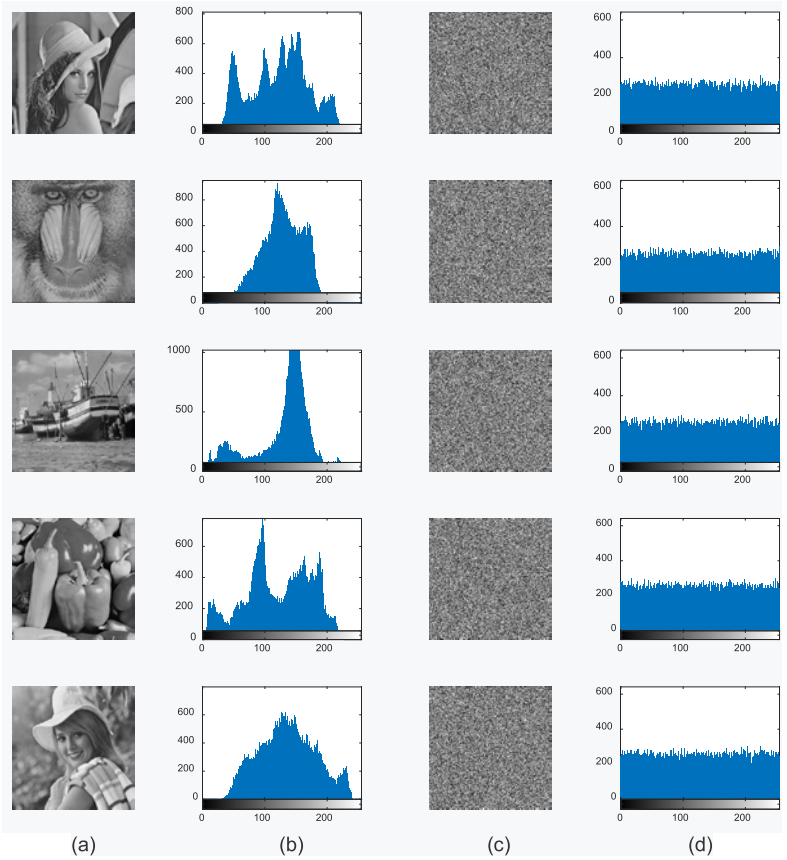


FIGURE 11. Histogram plots of several images. (a) original images; (b)Histograms of the original images (a); (c) cipher images; (d)histograms of the related cipher images (c).

TABLE 4. Statistics of $\chi 2$ in ciphertext images.

Image	Plaintext	Ciphertext	Test results
Lena	39851	271.12	pass
Baboon	79057	250.34	pass
Boat	100853	271.99	pass
Elaine	36282	246.73	pass
Pepper	31627	255.40	pass

encrypted images are shown in Table 4. All the cipher images passed the test when the significance level $\alpha = 0.05$.

D. INFORMATION ENTROPY ANALYSIS

The entropy of information can reflect the randomness of information. The closer the entropy of the information is to the theoretical value, the stronger the randomness of the information and the higher its ability to resist attacks will be. The formula to calculate information entropy is shown in Equation (15).

$$H = -\sum_{i=0}^{255} p\left(m_i\right) log_2 p\left(m_i\right), \tag{15}$$

where, $p(m_i)$ represents the probability distribution of the gray value m_i. The closer the information entropy of the grayscale image is to 8, the better the image randomness is. The information entropy of some images commonly used in the field of image encryption and their corresponding encrypted images are shown in Table 5. The information entropy of the encrypted images after the algorithm is close to 8, indicating that the algorithm has good randomness.

TABLE 5. Information entropy and local information entropy.

Image	Information entropy		Local information entropy
	Plaintext	Cipher	Cipher
Lena	7.4532	7.9971	7.9038
Baboon	7.0092	7.9972	7.9036
Boat	7.1572	7.9970	7.9032
Elaine	7.4874	7.9973	7.9000
Pepper	7.5797	7.9972	7.9039

The overall information entropy can reflect the randomness of the whole image, but it is not enough to reflect the local randomness of the image. Therefore, while using information entropy to reflect the overall randomness of the image, this article also adds local information entropy to reflect the randomness of the local area of the image. The statistical method of local information entropy is shown in formula (16).

$$\overline{H_{(k,T_B)}(S)} = \frac{1}{k}\sum_{i=1}^{k} H(S_i), \tag{16}$$

TABLE 6. Correlation coefficients of each direction of plain image and cipher image.

Image		Level	Vertical	Positive diagonal	Antidiagonal
Lena	plain image	0.9660	0.9371	0.9024	0.9270
	Cipher image	0.0071	-0.0052	0.0013	0.0055
Baboon	Original image	0.8227	0.8747	0.7894	0.7761
	Encryption image	0.0037	0.0072	0.0038	0.0119
Boat	Original image	0.9462	0.9233	0.8811	0.8919
	Encryption image	0.0128	0.0040	0.0050	0.0120
Elaine	Original image	0.9597	0.9725	0.9566	0.9553
	Encryption image	0.0012	0.0025	0.0233	0.0006
Pepper	Original image	0.9695	0.9652	0.9376	0.9390
	Encryption image	0.0053	0.0221	0.0035	0.0010

where k represents the total number of selected regions and T_B represents the number of pixels in the selected regions. When the significant level of local information entropy $\alpha = 0.05$, the value interval is [7.900573, 7.904227]. The local information of cipher images commonly used in the field of image encryption is shown in Table 5. It can be seen from the comparison that the encryption algorithm can make cipher images have good local randomness.

E. CORRELATION ANALYSIS

Plain images have an obvious rule in histogram statistics, and there is a strong correlation among their pixels. As shown in the statistics of the adjacent pixels of the plain image in Figure 12, there is a strong correlation between adjacent pixels. Breaking the correlation of the adjacent pixels is of great significance for resisting statistical attacks. The method used to calculate the correlation coefficient is shown in formula (17).

$$\begin{cases} E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \\ D(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))^2 \\ cov(x,y) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))(y_i - E(y)) \\ r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \end{cases} \tag{17}$$

Ten thousand pixels in the image were randomly selected, and the correlation of adjacent pixels is calculated from the horizontal, vertical, and diagonal directions. The statistical results are shown in Table 6. Figure 12 shows the correlation of plaintext and the cipher of Lena image in all directions. It can be seen from the comparison that the correlation of the adjacent pixels of the original image is very strong and tends to 1, while the correlation of the cipher image is very weak and tends to 0. Therefore, the encryption algorithm has a strong ability to break the correlation between adjacent pixels.

F. DIFFERENTIAL ATTACK ANALYSIS

Differential attack analysis aims to make small changes to the original image, encrypt it, and analyze the cipher image to analyze its sensitivity to plaintext. If the change of a pixel value of the plain image can change the cipher image to a large extent, it shows that the encryption algorithm has a strong ability to resist differential attack. The NPCR and UACI are used to measure its ability to resist differential attacks. The analysis of the NPCR and UACI values of the proposed algorithm are shown in Table 7. The data shown in the table are close to the theoretical value, which reflects the use of the encryption algorithm to encrypt the cipher image. There is a strong correlation between the original image and the encrypted image, and even if 1 bit is changed in the original image, it will completely change cipher images.

TABLE 7. Values of NPCR and UACI between corresponding cipher images (%) when the plain image is slightly changed.

Image	NPCR	UACI
Lena	99.6337	33.6050
Baboon	99.6629	33.5570
Boat	99.5850	33.4178
Elaine	99.6155	33.5286
Pepper	99.5865	33.6048

G. DATA LOSS ATTACK ANALYSIS

In the process of data transmission, data loss may occur whether the transmission is affected or attacked. When data are lost to different degrees, a good encryption method can restore the features of the original image to a great extent when decrypting the cipher image, which is also the role of the scrambling process in the encryption algorithm. To test the degree of recovery after partial data loss of the image for the method proposed in this article, the cipher image Cameraman was clipped 1/64, 1/16, and 1/4 and decrypted. The areas of lost data were recovered to different degrees. The specific results are shown in Figure 13. The results show that the algorithm can recover the original image features to some extent when the cipher image is lost to some extent.

H. ANALYSIS OF ANTINOISE ATTACK

In the process of signal transmission, the cipher image affected by noise will show different degrees of distortion in the decryption process. Pepper and salt noise, white Gaussian

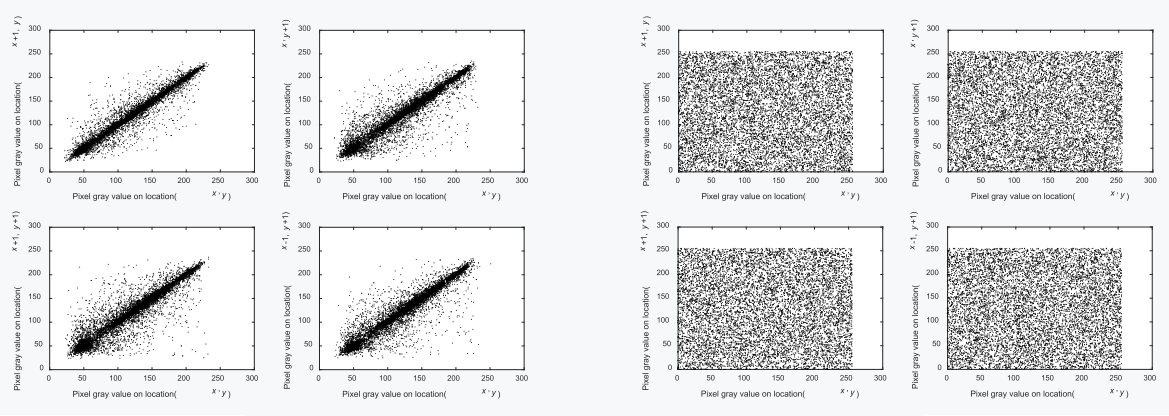


FIGURE 12. Correlation analysis (a) Correlation of the plain Lena in horizontal, vertical, positive diagonal, and anti-angular directions; (b) Correlation of cipher image Lena in horizontal, vertical, positive diagonal, and anti-angular directions.

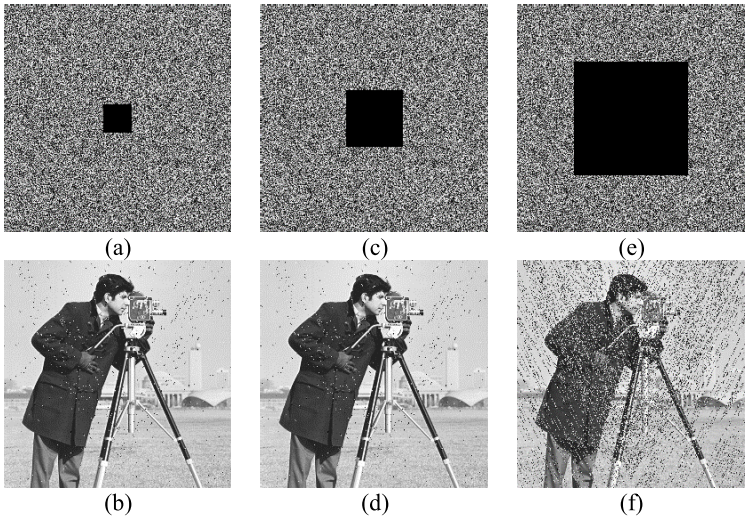


FIGURE 13. Data loss attack analysis. (a) cipher image with 1/64 data lost; (b) Decrypted image of (a); (b) Decrypted image of (a); (c) cipher image with 1/16 data lost; (d) Decrypted image of (c); (e) cipher image with 1/4 data lost; (f) Decrypted image of (e).

noise, and Poisson noise are common noise signals in information transmission. To analyze the antinoise attack capability of the encryption algorithm, pepper and salt noise with strengths of 0.01, 0.05 and 0.1 were selected to process and decrypt the Cameraman cipher image. The results are shown in Figure 14.

Based on the correlation coefficient of the cipher image, the NPCR and UACI values of the decrypted image with added noise and the original image are compared. The results are shown in Table 8. The statistical results show that the algorithm has strong recoverability when the image is affected by noise, and it can effectively resist the influence of noise.

TABLE 8. Resistance of the algorithm to noise attack.

Noise intensity	Correlation coefficient			NPCR	UACI
	Level	Vertical	Diagonal		
0	0.9573	0.9134	0.9066	0	0
0.01	0.9102	0.8766	0.8455	1.8081	0.6357
0.05	0.8644	0.8252	0.8116	3.8132	1.3515
0.1	0.5905	0.5601	0.5463	18.0923	6.2890

TABLE 9. The performance assessment results of all-black and white images.

Cipher image	Correlation coefficient			Entropy
	Horizontal	Vertical	Diagonal	
Full Black	0.00272167	0.00110687	0.00610290	7.9973840
Full white	0.00111899	0.00982387	0.00593937	7.997392

I. CHOSEN PLAINTEXT ATTACK

Attackers mostly use special images, such as white and all black, to attack image encryption methods and find the key. However, in this method, the initial value of the chaotic systems is associated with the hash value of the image being encryption, and different image encryptions using different chaotic sequences can achieve the result of "One-Time-Pad" and can resist the chosen plaintext attack.

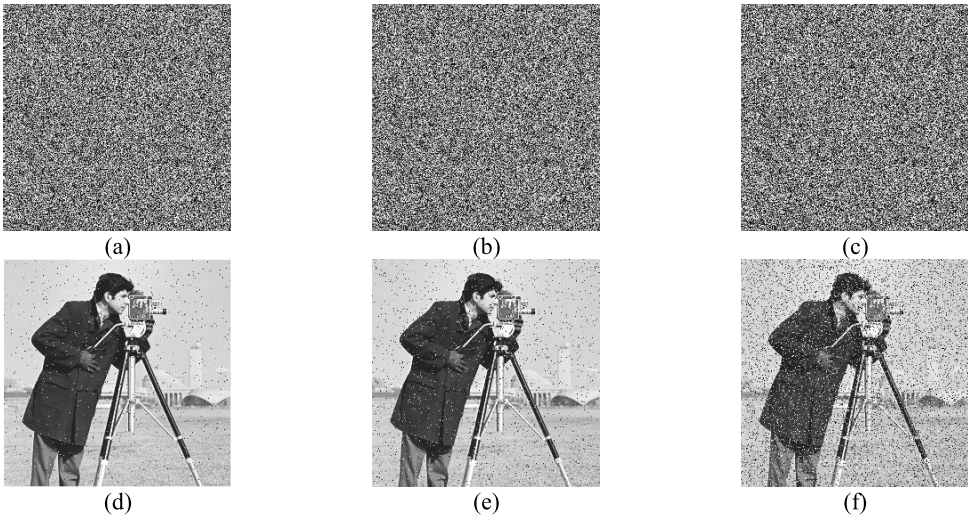


FIGURE 14. Cipher images subjected to noise attack and corresponding decrypted images. (a) Cipher image with noise intensity 0.01; (b) Cipher image with noise intensity 0.05; (c) Cipher image with noise intensity 0.1; (d) Decrypted image from (a); (e) Decrypted image from (b); (f) Decrypted image from (c).

TABLE 10. The performance evaluation of other images.

Image	Information entropy	NPCR	UACI	Correlation coefficient			
				Horizontal	Vertical	Diagonal	Diagonal
5.1.09	7.9968994	99.595642	33.443190	-0.0129	0.0116	0.00600	-0.0008
5.1.10	7.9969899	99.581909	33.508683	-0.0032	-0.0015	0.0027	0.0027
5.1.11	7.9970258	99.600219	33.383250	-0.0057	0.0041	-0.0021	0.0113
5.1.12	7.9974092	99.583435	33.483054	0.0407	0.0074	0.0015	0.0173
5.1.13	7.9975222	99.606323	33.510760	0.00004	-0.0178	-0.0328	0.0004
5.1.14	7.9975803	99.610900	33.434621	-0.0041	-0.0029	-0.0177	-0.0007
5.2.08	7.9992716	99.591064	33.551791	0.0038	0.0094	0.0073	-0.0147
5.2.09	7.9993891	99.619674	33.455086	0.0151	0.0019	0.011	0.0044
5.2.10	7.9993010	99.597167	33.446369	-0.0009	-0.0128	0.0051	-0.010
7.1.01	7.9993059	99.600219	33.487816	-0.0007	-0.0007	0.0096	0.0009
7.1.02	7.9993468	99.618911	33.593159	0.0017	0.00030	0.0029	0.0053
7.1.03	7.9993612	99.612808	33.432213	0.0065	-0.0004	-0.0105	-0.0006
7.1.04	7.9993449	99.590682	33.451413	-0.0089	-0.0038	-0.0070	-0.0101
7.1.05	7.9991923	99.581146	33.437395	0.0045	-0.0063	0.0040	-0.0122
7.1.06	7.9993197	99.629211	33.491296	0.0111	-0.0025	-0.00070	-0.0165
7.1.07	7.9992833	99.586868	33.397773	-0.0073	0.017	-0.0042	-0.0032
7.1.08	7.9992606	99.590682	33.408604	0.0141	-0.0073	-0.0246	0.0051
7.1.09	7.9992235	99.606704	33.536219	0.0113	0.0084	0.00010	-0.0109
7.1.10	7.9993525	99.621582	33.507982	-0.0081	0.0158	-0.0073	0.0011
boat.512	7.9993403	99.614334	33.440083	-0.0097	0.019	0.0016	0.0138
gray21.512	7.9993519	99.618911	33.469947	0.0023	-0.0165	0.0050	-0.0060
ruler.512	7.9993449	99.615859	33.428500	0.0156	-0.0142	0.0124	-0.0018

TABLE 11. Performance of the proposed scheme and other methods for Lena in size 256*256.

Schemes	Entropy	NPCR	UACI	Correlation coefficient		
				Horizontal	Vertical	Diagonal
Proposed Scheme	7.9971	99.6337	33.6050	0.0071	-0.0052	0.0013
Ref. [25]	7.9963	99.5040	31.6551	-0.0074	0.0069	-0.0191
Ref. [26]	7.9974	99.4062	37.6389	-0.0063	0.0095	0.0089
Ref.[27]	7.9970	99.6093	33.4597	0.0022	0.0013	0.0008
Ref.[28]	7.9974	99.6114	33.4636	−0.0223	−0.0084	−0.0086

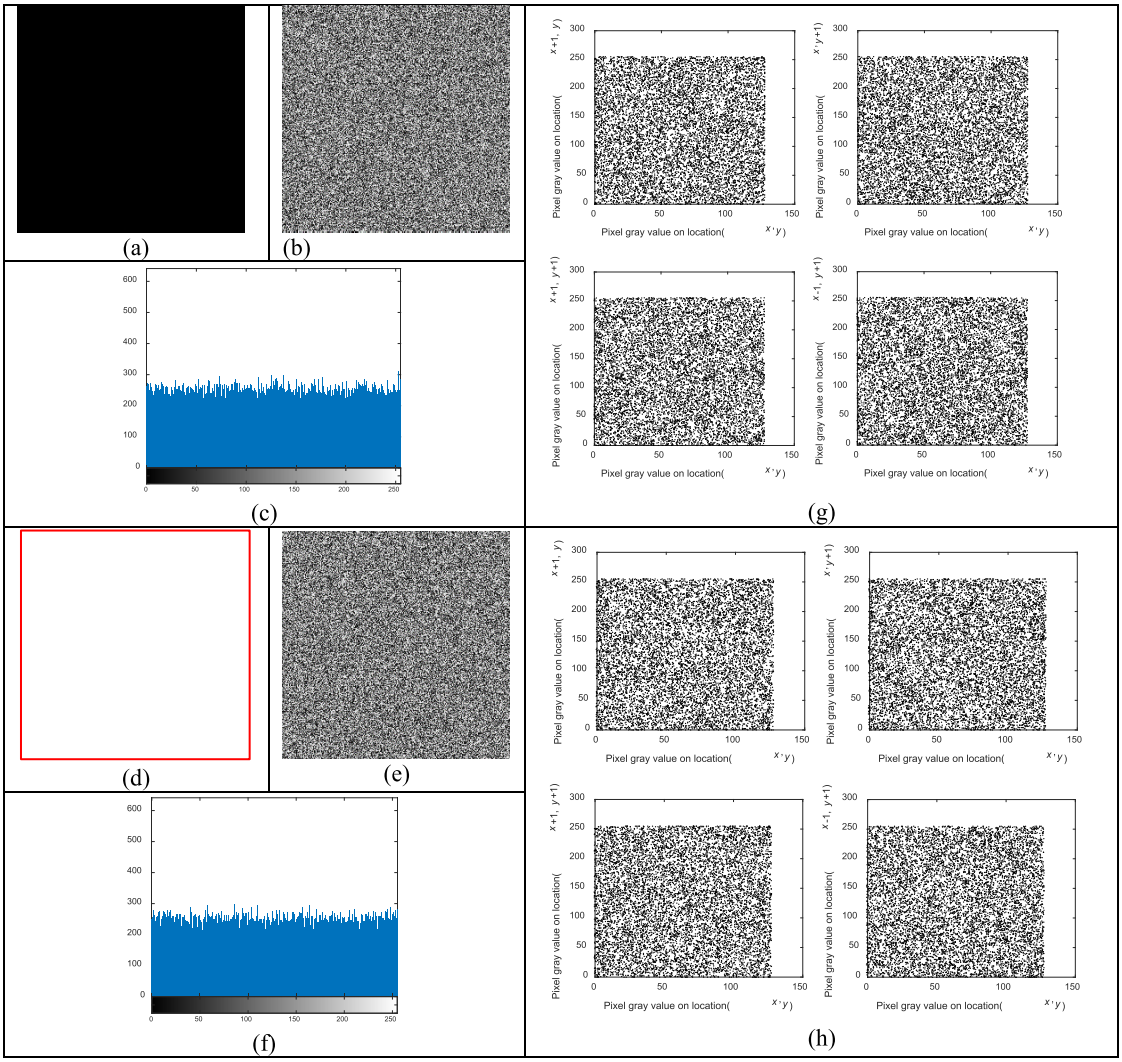


FIGURE 15. Results of all black and all white image analysis. (b) and (e) are cipher images corresponding to all-black and all-white images, respectively; (c) and (f) histograms of cipher images for all-black and all-white images, respectively; (g) and (h) are correlation analyses for four directions of all-black and all white cipher images, respectively.

Figure 15 shows the encryption results of all-black and all-white images of size 256*256. The correlation distribution and histogram analysis are shown in Figure 12 and Figure 13, respectively. Table 9 provides the information entropy and three kinds of correlation of images. Experimental analysis shows that this method can resist both known attacks and chosen plain attacks.

J. TEST OTHER IMAGES AND THEIR COMPARATIVE ANALYSIS

To further verify the performance of this scheme, we test the images in the USC-SIPI Miscellaneous database. The performance evaluation includes differential attack, the correlation coefficient and information entropy. The results are shown in Table 10.

At the same time, the performance of the scheme is compared with the existing literatures. Table 11 shows the comparison and analysis results of Lena images with size 256*256 in terms of the correlation, NPCR, UACI and information entropy. From these results, we can see that our scheme has better security than other methods.

VI. CONCLUSION

This article presents a simple and effective image encryption algorithm based on a filling curve and Josephus scrambling. First, the Josephus traversal function is improved by adding a variable step length parameter that expands the diversity of the Josephus traversal function. The sequence generated by chaotic map is used as the starting point and initial step of Josephus traversal to scramble pixels. Second, a novel Y-index SFC is proposed, and a pixel scrambling method based on the Y-index is designed. By comparing our method with the Hilbert curve and square wave curve, the Y-index is found to have obvious advantages in scrambling. Finally,

the proposed algorithm combines with a chaotic sequence to realize bidirectional feedback of the ciphertext. To enhance the security of the scheme, different chaotic sequences are used in the permutation and diffusion processes. In addition, the diffusion initial values and parameters related to chaotic sequence generation are generated by the hash function of the plain image, which enhances the correlation between the encryption process and the plain image, achieves the effect of ''one-time pad'', and improves the security of the system. We will study its application in video encryption in the future.

REFERENCES

[1] X. Chai, H. Wu, Z. Gan, Y. Zhang, Y. Chen, and K. W. Nixon, ''An efficient visually meaningful image compression and encryption scheme based on compressive sensing and dynamic LSB embedding,'' *Opt. Lasers Eng.*, vol. 124, Jan. 2020, Art. no. 105837.
[2] Y. Niu, Z. Zhou, and X. Zhang, ''An image encryption approach based on chaotic maps and genetic operations,'' *Multimedia Tools Appl.*, vol. 79, nos. 35–36, pp. 25613–25633, Sep. 2020.
[3] X. Zhang, Z. Zhou, and Y. Niu, ''An image encryption method based on the Feistel network and dynamic DNA encoding,'' *IEEE Photon. J.*, vol. 10, no. 4, pp. 1–14, Aug. 2018.
[4] Z. Hua, F. Jin, B. Xu, and H. Huang, ''2D Logistic-Sine-coupling map for image encryption,'' *Signal Process.*, vol. 149, pp. 148–161, Aug. 2018.
[5] H. Natiq, N. M. G. Al-Saidi, M. R. M. Said, and A. Kilicman, ''A new hyperchaotic map and its application for image encryption,'' *Eur. Phys. J. Plus*, vol. 133, no. 1, Jan. 2018, Art. no. 6.
[6] S. Vaidyanathan, A. Akgul, S. Kaçar, and U. Çavuşoğlu, ''A new 4-D chaotic hyperjerk system, its synchronization, circuit design and applications in RNG, image encryption and chaos-based steganography,'' *Eur. Phys. J. Plus*, vol. 133, no. 2, Feb. 2018.
[7] A. K. Farhan, N. M. G. Al-Saidi, A. T. Maolood, F. Nazarimehr, and I. Hussain, ''Entropy analysis and image encryption application based on a new chaotic system crossing a cylinder,'' *Entropy*, vol. 21, no. 10, Sep. 2019, Art. no. 958.
[8] X.-J. Tong, ''The novel bilateral–diffusion image encryption algorithm with dynamical compound chaos,'' *J. Syst. Softw.*, vol. 85, no. 4, pp. 850–858, Apr. 2012.
[9] P. Ping, F. Xu, Y. Mao, and Z. Wang, ''Designing permutation–substitution image encryption networks with henon map,'' *Neurocomputing*, vol. 283, no. 29, pp. 53–63, Mar. 2018.
[10] X. Zhang, L. Wang, Z. Zhou, and Y. Niu, ''A chaos-based image encryption technique utilizing Hilbert curves and H-Fractals,'' *IEEE Access*, vol. 7, pp. 74734–74746, 2019.
[11] X. Wang, Y. Wang, X. Zhu, and S. Unar, ''Image encryption scheme based on chaos and DNA plane operations,'' *Multimedia Tools Appl.*, vol. 78, no. 18, pp. 26111–26128, Sep. 2019.
[12] S. Mozaffari, ''Parallel image encryption with bitplane decomposition and genetic algorithm,'' *Multimedia Tools Appl.*, vol. 77, no. 10, pp. 1–21, 2018.
[13] S. S. Maniccam and N. G. Bourbakis, ''Image and video encryption using SCAN patterns,'' *Pattern Recognit.*, vol. 37, no. 4, pp. 725–737, Apr. 2004.
[14] G. Bhatnagar, Q. M. J. Wu, and B. Raman, ''Image and video encryption based on dual space-filling curves,'' *Comput. J.*, vol. 55, no. 6, pp. 667–685, Jun. 2012.
[15] V. Suresh and C. Madhavan, ''Image encryption with space-filling curves,'' *Defence Sci. J.*, vol. 62, no. 1, pp. 46–50, Jan. 2012.
[16] K. S Kavya, Prabavathi, ''Image encryption using Hilbert space filling curve and Henon map,'' *Int. J. Innov. Res. Comput. Sci. Technol.*, vol. 3, no. 3, pp. 56–60, 2015.
[17] G. Bhatnagar and Q. M. Jonathan Wu, ''Selective image encryption based on pixels of interest and singular value decomposition,'' *Digit. Signal Process.*, vol. 22, no. 4, pp. 648–663, Jul. 2012.
[18] T. Sivakumar and R. Venkatesan, ''A novel approach for image encryption using dynamic SCAN pattern,'' *IAENG Int. J. Comput. Sci.*, vol. 41, no. 2, pp. 11–21, 2014.
[19] P. Murali and V. Sankaradass, ''An efficient space filling curve based image encryption,'' *Multimedia Tools Appl.*, vol. 78, no. 2, pp. 2135–2156, Jan. 2019.
[20] K. U. Shahna and A. Mohamed, ''A novel image encryption scheme using both pixel level and bit level permutation with chaotic map,'' *Appl. Soft Comput.*, vol. 90, pp. 106–162, 2020.
[21] M. Markus and B. Hess, ''Lyapunov exponents of the logistic map with periodic forcing,'' *Comput. Graph.*, vol. 13, no. 4, pp. 73–78, 1988.
[22] Y. Zhou, L. Bao, and C. P. Chen, ''A new 1D chaotic system for image encryption,'' *Signal Process.*, vol. 97, no. 11, pp. 172–182, 2014.
[23] G. Alvarez and S. Li, ''Some basic cryptographic requirements for chaos-based cryptosystems,'' *Int. J. Bifurcation Chaos*, vol. 16, no. 8, pp. 2129–2151, Aug. 2006.
[24] Z.-H. Gan, X.-L. Chai, D.-J. Han, and Y.-R. Chen, ''A chaotic image encryption algorithm based on 3-D bit-plane permutation,'' *Neural Comput. Appl.*, vol. 31, no. 11, pp. 7111–7130, 2018.
[25] Y. Guo, L.-P. Shao, and L. Yang, ''Bit-level image encryption algorithm based on Josephus and Henon chaotic map,'' *Appl. Res. Comput.*, vol. 32, no. 4, pp. 1131–1137, Apr. 2015.
[26] A. Bakhshandeh and Z. Eslami, ''An authenticated image encryption scheme based on chaotic maps and memory cellular automata,'' *Opt. Lasers Eng.*, vol. 51, no. 6, pp. 665–673, Jun. 2013.
[27] C. Chen, K. Sun, and S. He, ''An improved image encryption algorithm with finite computing precision,'' *Signal Process.*, vol. 168, Mar. 2020, Art. no. 107340.
[28] Y. Zhang, ''The fast image encryption algorithm based on lifting scheme and chaos,'' *Inf. Sci.*, vol. 520, pp. 177–194, May 2020.

YING NIU received the master's degree from the Zhengzhou University of Light Industry, in 2009. She is currently an Associate Professor with the College of Architecture Environment Engineering, Zhengzhou University of Light Industry. Her research interests include DNA computing and information security.

XUNCAI ZHANG (Member, IEEE) was born in Zhoukou, China, in 1981. He received the Ph.D. degree from the Huazhong University of Science and Technology, in 2009. From 2010 to 2012, he accomplished the Postdoctoral Research at Peking University. He is currently an Associate Professor with the College of Electrical and Information Engineering, Zhengzhou University of Light Industry. His research interests include DNA computing and information security.