Chaotic Image Encryption Algorithm Based on Fractional Order Scrambling Wavelet Transform and 3D Cyclic Displacement Operation

XIAOPENG YAN, XINGYUAN WANG, AND YONGJIN XIAN
School of Information Science and Technology, Dalian Maritime University, Dalian 116026, China
Corresponding authors: Xiaopeng Yan (karasyxp@126.com) and Xingyuan Wang (xywang@dlmu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61672124, in part by the Password Theory Project of the 13th Five-Year Plan National Cryptography Development Fund under Grant MMJJ20170203, in part by the Liaoning Province Science and Technology Innovation Leading Talents Program Project under Grant XLYC1802013, in part by the Key Research and Development Projects of Liaoning Province under Grant 2019020105-HJ2/103, and in part by the Jinan City 20 Universities Funding Projects Introducing Innovation Team Program under Grant 2019GXRC031.

ABSTRACT In order to deal with the security of image information, this paper proposes a chaotic image encryption method based on fractional wavelet transform. First, fractional wavelet transform is applied to preprocess the image as a signal. After three-level fractional wavelet decomposition, high-frequency components are obtained with low frequency components. Second, the hash function is used to generate the initial value of the chaotic system for the plaintext operation, and the initial value is brought into the Lorenz system to obtain the chaotic sequence, the chaotic sequence is scrambled for the high frequency components, and the 3D cyclic displacement scrambling is performed for the low frequency components. The high-frequency components and low-frequency components are reconstructed by fractional wavelet transform and restored to the final scrambled image. Last, in order to further deepen the encryption effect, cyclic displacement diffusion is performed on the scrambled image to obtain the final encrypted image. Experimental and simulation results show that the algorithm proposed in this paper has excellent encryption effect and can resist common attack methods.

INDEX TERMS Fractional wavelet transform, Lorenz chaotic system, cyclic displacement operation, hash function, reconstruction.

I. INTRODUCTION Image encryption is widely used in secure communication, information hiding and digital watermarking. The research on image encryption technology has high theoretical and practical significance [1]–[3]. With the development of the digital age, the amount of information that needs to be stored, transmitted and processed increases exponentially, and the mode of information transmission is also changing constantly. As an important information carrier, digital images are increasingly widely used in the field of information transmission [4]. Current research on digital image encryption method basic is divided into two categories: space transform encryption scheme and transform domain encryption method [5], more popular scheme mainly concentrated in the frequency domain or airspace simple encryption, the method has some deficiencies, the characteristics of image information is a huge amount of information, and need the information representation and transmission, image the difficulty lies in the transmission process of information security assurance image data and the security of [6], [7]. How to store and transmit image data safely and effectively has become an urgent need in modern information society. The original image data has strong correlation and a lot of redundancy [8]. Eliminating this redundancy can save storage space. In most images, there is a large correlation between adjacent pixels, namely, spatial redundancy. There is a great correlation between the adjacent frames of sequential images, namely, time redundancy. The method to eliminate the redundancy is called image compression, which improves the compression ratio of image encryption, thus improving the image encryption efficiency and facilitating the transmission and storage of ciphertext information. Reference [9] proposes to add confusion and diffusion of chaos to existing alternatives.
to completely eliminate correlation. First, TD-ERCS chaotic map is used to displace pixels, which breaks the autocorrelation of data. The final encrypted image is generated by using the logistic map to carry out bite-or to the diffusion. At present, almost all image encryption algorithms are single image encryption system, image encryption efficiency is low, and image compression encryption algorithms based on high-resolution wavelet analysis are few. Wavelet transform coding provides multi-scale and multi-resolution image transformation, which can effectively eliminate statistical redundancy and visual redundancy. It plays an important role in the field of image compression and coding. Image compression and encryption using wavelet transform can achieve the compression ratio of any compressed image theoretically, and it is also a relatively simple practice to achieve such a goal. Of course, any method is different from other methods in terms of its advantages, but also has some defects. Therefore, in order to achieve good image compression and encryption, a variety of technologies are used. Such image compression and encryption based on wavelet analysis is no exception. In most cases, wavelet analysis needs to be dynamically combined with other related technologies, such as dynamic S-box technology, neural network technology [10], Fourier transform and chaotic system [11], etc., Reference [12] proposes a new encryption scheme, which can transform the plaintext image pixels into visually meaningful encrypted images. In this scheme, confusion and diffusion of plaintext image are introduced by using entangled logic mapping, and finally the final ciphertext image is realized by using grayscale replacement Box (S-box). Reference [13] proposes to divide the plaintext image into several blocks, and then calculate the correlation coefficient of each block. Blocks with maximum relational values are processed pixel by pixel using random Numbers generated by skew maps based on predefined thresholds. Finally, two random sequences generated by TD-ERCS chaotic map are used to sort the whole image, so as to achieve more perfect results [14]. In view of this, in view of the large amount of image data and the characteristics of the high redundancy, proves that the traditional encryption algorithms are not suitable for image encryption, we put forward the improvement scheme based on wavelet and chaos theory, we select [15], [16] lorenz chaotic system combined with the wavelet analysis to image compression encryption research, image encryption based on chaos and wavelet technology not only has important theoretical value, but also has a certain application value.

Image coding technology began in the late 1940s, and early classical coding theories such as entropy coding, predictive coding and transform coding originated from Shannon’s information theory. The starting point of these coding theories is to eliminate statistical redundant information in images or visual redundancy, structural redundancy and knowledge redundancy in image data [17]. Since the 1990s, wavelet transform coding has provided a multi-scale and multi-resolution image transformation method, and can effectively eliminate statistical redundancy and visual redundancy, so wavelet change analysis method has begun to take an important position in the field of image compression coding. Since the first information hiding workshop was held in Cambridge University in 1996, the research on digital image encryption technology has made great progress. The second and third International Conferences on Information Hiding held in 1998 and 1999 in the United States and Germany led to a growing number of scholars working on image encryption. The fourth International Conference on Information Hiding was held in Pittsburgh in April 2001, and the fifth and sixth International Conferences on Information Hiding were held in The Netherlands and Canada in October 2002 and May 2004, respectively [18]. Research on image encryption, wavelet theory and chaos theory has become a hot research topic. At the same time, image encryption method is implemented in the frequency domain, which combines chaos theory and wavelet transform. Compressed and encrypted images can also be used as preprocessing before embedding digital watermarks. All of these have become research hotspots.

In recent years, the research on chaotic system is more and more in-depth. The research on stability, controllability and synchronous control of chaotic systems [19] has become extremely hot. In the field of image encryption, chaotic encryption is a typical representative of spatial transformation and release methods [20]–[23]. Due to the initial value of chaotic system Sensitivity, pseudo-randomness and other good properties have been widely used in the field of encryption [24]–[27]. Diaconu [28] proposed a chaotic image encryption method of pixel circular arrangement, Hua et al. [29] proposed a new two Dimensional sine logic modulation mapping is used for encryption. Sharma et al. [30] proposed a double random phase image coding encryption technology based on Lorenz system.

The typical representative method of the transform domain is the wavelet transform method [31]–[34]. The wavelet transform is very widely used in signal processing because of its good positioning characteristics and detailed processing functions [29], [35]–[37]. The wavelet transform is used in image processing [38]–[42]. The application in is also extremely popular. Tedmori [43] uses haar wavelet as the basis function to introduce a chaotic encryption method based on wavelet transform. Similarly, Ajish [44] discussed a new image encryption algorithm based on wavelet transform and dynamic S-box AES algorithm. Liu et al. [45] proposed an image encryption algorithm based on fractional Fourier transform domains. The combined use of wavelet transform and chaotic system can greatly enhance the encryption effect, and the encryption result has higher stability and resistance to attack [46].

In this paper, the image is preprocessed by the fractional wavelet transform method. The fractional wavelet has better image processing effects and can reflect the image signal information in the time domain and the frequency domain. In this paper, combined with chaotic system, cyclic displacement scrambling and diffusion are designed. Scrambling is
based on a 3D cyclic displacement mechanism. Diffusion is based on sequential cyclic displacement. The image after fractional wavelet transform is scrambled by cyclic displacement. After diffusion, the final encrypted image is obtained. The structure of this paper is as follows: The second part introduces the pre-theoretical method of the encryption algorithm, the third part introduces the specific model of the circular displacement operation we proposed, the fourth part introduces the encryption and decryption process, and the fifth part is the simulation result, the sixth part is the security analysis, the seventh part is the conclusion.

II. PRELIMINARY WORKS

A. KEY GENERATION

Step 1: Generate Hash array \{H\} with SHA-512 Hash algorithm and plaintext image [47], [48], extract 12 elements from \{H\} as the initial value and control parameters of Lorenz chaotic map. Every three keys form a team of key groups, and the parameter calculation process is shown in Eq. (1):

\[
\begin{aligned}
x_1(0) &= Key_1, y_1(0) = Key_2, z_1(0) = Key_3 \\
x_2(0) &= Key_4, y_2(0) = Key_5, z_2(0) = Key_6 \\
x_3(0) &= Key_7, y_3(0) = Key_8, z_3(0) = Key_9 \\
x_4(0) &= Key_{10}, y_4(0) = Key_{11}, z_4(0) = Key_{12}.
\end{aligned}
\]

Step 2: The initial value composed of the key group generates 12 columns of chaotic sequences through the Lorenz system, and every three sequences are a group, the calculation process is shown in Eq. (2):

\[
K_i = [x_i, y_i, z_i] \quad i = [1, 2, 3, 4]
\]

Hash algorithms can compress messages of any length to a fixed length. Often used in cryptographic systems to defend against violent attacks. As Hash functions such as sha-1 are broken, some encryption algorithms are no longer safe. At present, the encryption algorithm based on sha-2 design has not been cracked. The second part of the key selects the SHA-512 hash algorithm in SHA-2 and combines the algorithm with the plaintext image \(P\) to obtain a hash array of length 128-bit. Even if you only change the 1-bit value of \(P\) pixel, the resulting hash array is completely different, and the complexity is \(2^{512}\), which is sufficient to resist violent attacks and achieve the purpose of one density at a time. For the convenience of the following description, here is given a 128-bit hexadecimal Hash array, denoted as \(H\):

\[
H = [h_1, h_2, \ldots, h_{128}]
\]

B. CHAOTIC SYSTEM

The Lorenz system is a strange attractor obtained by American meteorologists during the study of convection experiments, and then a dynamic system evolved [49]. During the rise of the system energy, the fluid orbit enters from the right outer edge, from inside to outside, it jumps to the left in a random state. During the entire change process, all behaviors are acyclic and unpredictable, and the orbits never intersect themselves, thus forming a chaotic behavior. As a three-dimensional chaotic system, Lorenz has a chaotic sequence that is more complex and unpredictable than a lower-dimensional chaotic system, and has a better pseudo-random effect. The three-dimensional system has more parameters and initial values, and has a greater secret key space.

The dynamic equation of Lorenz chaotic system is:

\[
\begin{align*}
\frac{dx}{dt} &= a(y - x) \\
\frac{dy}{dt} &= cx - yz - y \\
\frac{dz}{dt} &= xy - bz,
\end{align*}
\]

where \(a\), \(b\), and \(c\) are the parameters of the system, and the range of values is any real number. If \(a = 10\), \(b = 8/3\), and \(c = 28\), the system is in the best state of chaos, and the difference equation of the system can be expressed as:

\[
\begin{align*}
x_{k+1} &= x_k + a(y_k - x_k)dt \\
y_{k+1} &= y_k + (cx_k - y_k)dt \\
z_{k+1} &= z_k + (xy_k - bz_k)dt.
\end{align*}
\]

According to the difference equation, given initial values, \(x_0, y_0, z_0\), by solving the equation, three different chaotic sequences can be obtained simultaneously.

The phase diagram of the strange attractor of the Lorenz system is shown in the Fig. 1:

The sequence generated by the Lorenz system is a random sequence of chaos. During use, we need to standardize the chaotic sequence in order to be more suitable for the encryption system. The standardization steps are divided into two steps:

Step 1: In order to make the irregularity of the chaotic sequence more fully, we shift the sequence worth decimal point backward by \(n\) bits to obtain the sequence value containing integer bits.

Step 2: Since the pixel value range of the image is between (0,255), we map the sequence value into this range, and take the integer closest to the original value as the new sequence value.

The mathematical formula of the standardized model of chaotic sequence is:

\[
\begin{align*}
X(i) &= \text{floor}((x(i) \times 10^p) \mod 256) \\
Y(i) &= \text{floor}((y(i) \times 10^p) \mod 256) \\
Z(i) &= \text{floor}((z(i) \times 10^p) \mod 256).
\end{align*}
\]

C. FRACTIONAL WAVELET TRANSFORM

Mendlovic and Zalevsky first proposed the definition of fractional wavelet transform in 1997 [23]. The combination of fractional Fourier transform and traditional wavelet transform into a fractional discrete wavelet transform, FRWT definition formula is:

\[
W^{(\alpha)}(a, b) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} B_P(t, t') \chi(t') h^{(a)}(t) dt' dt,
\]

where
where \( B_p(t, t') \) is:

\[
B_p(t, t') = \sqrt{2} e^{-\pi (t^2 + t'^2)} \times \sum_{n=0}^{\infty} \frac{i^{-pn}}{2^{n!} n!} H_n(\sqrt{2\pi t}) H_n(\sqrt{2\pi t'}).
\] (8)

The fractional order change domain is:

\[
W(a_{mn}, b) = (a_m a_n)^{1/2} \int \int H^*(a_m u, a_n v) \times e^{(2j\pi a'_m u + 2j\pi a'_n v)}
\times F \{ F^{-1} \{ f(x, y)(x', y') \} (u, v) \} dudv.
\] (9)

in Eq. (9), \( h_{a,b}^s(x', y') \) represents the scale and translation function in the process of wavelet transform, \( a_{mn} = (a_m, a_n) \) represents the displacement scale, and \( p = [p_1, p_2] \) represents the order of wavelet transform.

Fractional wavelet reconstruction formula is:

\[
x(t) = \frac{1}{C} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \frac{1}{a^3} W(p)(a, b) B_p(t, t')
\times h(\frac{t' - b}{a}) dadbdt'.
\] (10)

The steps of performing a two-dimensional fractional wavelet transform on the initial image are: First performing a two-dimensional FRFT change on the matrix, the order is \( p_1, p_2 \), and then performing a two-dimensional wavelet decomposition to obtain the transformed image.

III. CYCLIC DISPLACEMENT OPERATION
A. 3D CYCLIC DISPLACEMENT SCRAMBLING
Treat the image as a 3D stereo model. Each pixel of the image matrix contains a pair of position information and a pixel value information. The three information form a spatial position. The position information of the matrix is scrambled by rotating the rows and columns of the matrix. Specific operations Proceed as follows:

Step 1: To obtain the chaotic sequence \( Z \), first normalize the chaotic sequence according to Eq. (11):

\[
Z(i) = \text{floor}(z(i) \times 10^n \mod 256).
\] (11)

Step 2: The step size following the back displacement is given by Eq. (12):

\[
H(i) = Z(i) \mod M.
\] (12)

\( H \) is a mapping matrix, and the position information and pixel values of the original image matrix are regarded as three-dimensional position information, namely:

\[
p = [i, j, k],
\] (13)

where \( p \) is the image matrix, \( i \) is the row coordinate, \( j \) is the column coordinate, \( k \) is the pixel information, and the scrambling formula is:

\[
p' = H \times p.
\] (14)
The horizontal circular diffusion formula is:

\[
\begin{pmatrix}
a_{11} & a_{12} & a_{13} \\
a_{21} & a_{22} & a_{23} \\
a_{31} & a_{32} & a_{33}
\end{pmatrix}
\rightarrow
\begin{pmatrix}
a_{13} & a_{11} & a_{12} \\
a_{22} & a_{23} & a_{21} \\
a_{33} & a_{31} & a_{32}
\end{pmatrix}
\]

**FIGURE 2.** Line loop transformation.

\[
\begin{pmatrix}
a_{13} & a_{11} & a_{12} \\
a_{22} & a_{23} & a_{21} \\
a_{33} & a_{31} & a_{32}
\end{pmatrix}
\rightarrow
\begin{pmatrix}
a_{13} & a_{23} & a_{32} \\
a_{13} & a_{21} & a_{31} \\
a_{22} & a_{11} & a_{21}
\end{pmatrix}
\]

**FIGURE 3.** Column cyclic transformation.

Taking the 3 × 3 matrix as an example, set the cyclic displacement variable to [2, 1, 2], the scrambling process is shown in the Fig. 2 and Fig. 3:

**B. CYCLIC DISPLACEMENT DIFFUSION**

The steps of cyclic displacement diffusion are:

**Step 1:** For the chaotic sequence \(x_4, y_4\), standardize the sequence:

\[
Z_3(i) = \text{floor}(x_4(i) \times 10^6) \mod 256
\]

\[
Z_4(i) = \text{floor}(y_4(i) \times 10^6) \mod 256.
\]  

(15)

**Step 2:** For \(M \times N\) images, convert the intermediate ciphertext matrix into a list of vectors as shown in the Eq. (16):

\[
P = [P_1, P_2, \cdots, P_{M \times N}].
\]  

(16)

**Step 3:** To diffuse the intermediate ciphertext vector, the horizontal circular diffusion formula is:

\[
P'_1 = P_1 \oplus Z_3(1) \oplus \text{sum}(P)
\]

\[
P'_i = P_i \oplus Z_3(i) \oplus P'_{i-1} \quad i = [2, 3, \ldots, M \times N],
\]  

(17)

convert the diffused vector \(P'\) into a matrix of \(M \times N\), and convert it vertically into a column of vectors \(E\), and perform vertical circular diffusion on the vector. The diffusion formula is:

\[
E'_1 = E_1 \oplus Z_4(1) \oplus \text{sum}(E)
\]

\[
E'_i = E_i \oplus Z_4(i) \oplus E'_{i-1} \quad i = [2, 3, \ldots, M \times N].
\]  

(18)

**IV. IMAGE ENCRYPTION STRATEGY**

The algorithm in this paper consists of two parts: Scramble and diffusion. Next, the algorithm is described in detail.

Image scrambling is used to change the position of image pixels. However, scrambling only completes the change of the pixel position of the image. In order to further deepen the effect of encryption, it is worth to modify the pixel using the diffusion method to enhance the encryption effect of the image. In this paper, \(M \times M\) images are taken as an example. The encryption process is as follows:

**Step 1.** The plaintext image is converted into a pixel matrix \(P\), and the matrix \(P\) is subjected to a fractional wavelet transform, and the transformed components are retained. The transform result is:

\[
[CA, CH, CV, CD] = \text{FRWT}(P).
\]  

(19)

**Step 2.** In Eq. (19), \(CA\) is the low-frequency component, and \(CH, CV, CD\) are the high-frequency components. This is the first-order wavelet transform. The low-frequency component is used as the new matrix to continue the wavelet transform, and the matrix is subjected to three-level wavelet decomposition. The result is:

\[
[CA_1, CH_{11}, CV_{11}, CD_{11}] = \text{FRWT}(CA)
\]

\[
[CA_{11}, CH_{111}, CV_{111}, CD_{111}] = \text{FRWT}(CA_1).
\]  

(20)

Finally, one low-frequency component \(CA_{11}\) and nine high-frequency components \(CH, CV, CD, CH_1, CV_1, CD_1, CH_{11}, CV_{11}, CD_{11}\) are obtained. The wavelet decomposition results are shown in the Fig. 4.

**Step 3.** Chaotic scrambling of all high-frequency components during the three-level wavelet transform of matrix \(P\), selecting the three pairs of key groups in Eq. (1), and iterating the Lorenz system to obtain three sets of chaotic sequences: \(x_1, y_1, z_1; x_2, y_2, z_2; x_3, y_3, z_3\), converting all high-frequency components into one row Vector, use chaotic sequence to chaotically scramble the high frequency components of matrix \(P\), first sort the chaotic sequence:

\[
\begin{align*}
C_1 &= \text{sort}(x_1); C_2 = \text{sort}(y_1); C_3 = \text{sort}(z_1) \\
C_{11} &= \text{sort}(x_2); C_{21} = \text{sort}(y_2); C_{31} = \text{sort}(z_2) \\
C_{12} &= \text{sort}(x_3); C_{22} = \text{sort}(y_3); C_{32} = \text{sort}(z_3)
\end{align*}
\]  

(21)
use the sorted chaotic sequence for chaotic scrambling of high frequency components:

\[
\begin{align*}
CH' &= CH(C_1(i)); \quad CD' = CD(C_2(i)); \quad CV' = CV(C_3(i)) \\
CH'_1 &= CH_1(C_{11}(i)); \quad CD'_1 = CD_1(C_{21}(i)); \quad CV'_1 = CV_1(C_{31}(i)) \\
CH'_{11} &= CH(C_{12}(i)); \quad CD'_{11} = CD_1(C_{22}(i)); \quad CV'_{11} = CV_1(C_{32}(i))
\end{align*}
\]

(22)

the scrambled high-frequency component sequence is converted into a matrix, which is the scrambled high-frequency component matrix:

\[
\begin{align*}
CH' &\rightarrow g_2, \quad CV' \rightarrow g_3, \quad CD' \rightarrow g_4 \\
CH'_1 &\rightarrow g_{21}, \quad CV'_1 \rightarrow g_{31}, \quad CD'_1 \rightarrow g_{41} \\
CH'_{11} &\rightarrow g_{22}, \quad CV'_{11} \rightarrow g_{32}, \quad CD'_{11} \rightarrow g_{42}.
\end{align*}
\]

(23)

**Step 4.** Perform a 3D cyclic displacement scrambling operation on the low-frequency component \(CA_{11}\) of the third-order fractional wavelet decomposition of the matrix \(P\). Use the key group: \(x_4(0) = Key_{10}, y_4(0) = Key_{11}, z_4(0) = Key_{12}\) to iterate the Lorenz system to obtain the chaotic sequence \(x_4, y_4, z_4\). Use the sequence \(z_4\) to set \(CA_{11}\) according to Eqs. (9)-(11) scrambling operation:

\[
g_{12} = H_4 \times CA_{11}.
\]

(24)

**Step 5.** The low-frequency component and the high-frequency component of the scrambled wavelet are reconstructed. The reconstruction process is:

\[
\begin{align*}
FF_1 &= iFRWT(g_{12}, g_{22}, g_{32}, g_{42}) \\
FF_2 &= iFRWT(FF_1, g_{21}, g_{31}, g_{41}) \\
FF_3 &= iFRWT(FF_2, g_{2}, g_{3}, g_{4}).
\end{align*}
\]

(25)

\(FF_3\) is the final scrambled image.

**Step 6.** Using the chaotic sequence \(x_4, y_4\), the scrambled image \(FF_3\) is subjected to cyclic displacement diffusion according to Eqs. (12)–(15), and the final encrypted image \(ee\) is output.

The encryption flowchart as shown in Fig. 5. The decryption algorithm is the inverse operation of the encryption algorithm.

**V. EXPERIMENTAL RESULTS**

In order to verify the effectiveness of the encryption strategy proposed in section 4, this paper simulates the algorithm through multiple images, and the image data used is the image in USC-SIPI 'Miscellaneous' image dataset. The simulation software is MATLAB R2017a, the computer is configured with 8GB RAM, Intel (R) Core (TM)i5 Duo CPU e6550@2.33 GHz, and the operating system is Windows 10.

In this section, grayscale images “Lena” (256 × 256), “Elaine” (512 × 512), “Airport” (1024 × 1024), color image “Lena” (512 × 512 × 3) and binary image were used for simulation experiments, and the resulting encrypted and decrypted images are shown in Fig. 6. In addition, images in the USC-SIPI ‘Miscellaneous’ image dataset are selected for data comparison and analysis.
FIGURE 6. Experimental results. Fig. 6(a)-6(c) are the original image, encrypted image and decrypted image of the grayscale image Lena, respectively. Fig. 6(d)-6(f) are the original image, encrypted image and decrypted image of the grayscale image Elaine, Fig. 6(g)-6(i) are the original image, encrypted image and decrypted image of the grayscale image Airport, Fig. 6(j)-6(l) are the original image, encrypted image and decrypted image of the color image Lena, Fig. 6(m)-6(o) are the original image, encrypted image and decrypted image of the binary image, respectively.
VI. SECURITY ANALYSIS

A good encryption algorithm should be able to withstand a variety of attacks, such as violent attacks, statistical attacks, differential attacks, selective plaintext attacks, and so on. In order to verify the security of this algorithm, this section will conduct theoretical analysis and numerical simulation of this algorithm from these aspects, and make a comparison with.

### TABLE 1. Uniformity evaluation by chi-square test.

| Image     | Chi-square | $H (0 \text{ or } 1)$ | Decision |
|-----------|------------|-----------------------|----------|
| 5.1.10    | 267.929    | 0                     | success  |
| 5.1.11    | 281.461    | 0                     | success  |
| 5.1.12    | 261.774    | 0                     | success  |
| 5.1.13    | 263.118    | 0                     | success  |
| 5.1.14    | 247.297    | 0                     | success  |
| 5.2.08    | 247.334    | 0                     | success  |
| 5.2.09    | 275.784    | 0                     | success  |
| 5.2.10    | 249.539    | 0                     | success  |
| 7.1.01    | 265.975    | 0                     | success  |
| 7.1.02    | 290.079    | 0                     | success  |
| 7.1.03    | 238.131    | 0                     | success  |
| 7.1.04    | 238.131    | 0                     | success  |
| 7.1.05    | 252.915    | 0                     | success  |
| 7.1.06    | 264.534    | 0                     | success  |
| 7.1.07    | 254.215    | 0                     | success  |
| 7.1.08    | 276.981    | 0                     | success  |
| 7.1.09    | 191.782    | 0                     | success  |
| 7.1.10    | 266.536    | 0                     | success  |
| boat512   | 246.807    | 0                     | success  |
| gray21512 | 251.293    | 0                     | success  |
| ruler512  | 256.292    | 0                     | success  |
| 5.3.01    | 250.814    | 0                     | success  |
| 5.3.02    | 247.825    | 0                     | success  |
FIGURE 9. Correlation of adjacent pixels. Fig. 9(a)-9(c) are the horizontal, vertical and diagonal correlations of the plaintext image Lena, respectively; Fig. 9(d)-9(f) are the horizontal, vertical and diagonal correlations of the ciphertext image Lena, respectively.

FIGURE 10. Ciphertext correlation line graph.

A. KEY SPACE ANALYSIS
The key space contains the keys used by the algorithm during encryption. The size of the key space directly determines the ability of an encryption algorithm to resist violent attacks. The valid keys of this algorithm are as follows: the number of iterations $TNI$; Random key sequence $Key$ with size 12, and key space $2^{12}$; The Hash array $H$ is generated by SHA-512 Hash algorithm from the plaintext image, and the key space is $2^{512}$; If the precision of the computer is $10^{-14}$, then the key space of this algorithm is $2^{12} \times 2^{512} \times 10^{14\times6}$, which is much larger than $2^{10^{20}}$ [50] required by the cryptographic system, so the algorithm in this paper can resist violent attacks.

B. KEY SENSITIVITY ANALYSIS
If a cryptographic system is highly sensitive to keys, it is a good key system. We take key $Key_1$ as an example to test the sensitivity of image “Lena”. The experimental results are shown in Fig. 6. When other keys remain unchanged, Fig. 7 (a) is the ciphertext image generated according to key $x_1(0)$; If the key $x_1(0)$ is changed from 0.3801 to 0.38010000000001, the generated ciphertext image is shown in Fig. 7(b). Fig. 7(c) is the difference between Fig. 7(a) and Fig. 7(b). Fig. 7(d) is the decrypted image obtained by decrypting Fig. 7(a) with the modified key $x_1(0)$; When the key changes slightly, the encryption image generated by the algorithm in this paper is completely different from the encryption image generated based on the original key in Fig. 7(b). It can be seen that the algorithm in this paper is extremely sensitive to the key.

C. HISTOGRAM ANALYSIS
An important measure of a cryptographic algorithm’s resistance to statistical analysis is to calculate the histogram of ciphertext images. Histogram is used to describe the pixel
| Image       | Type | Horizontal | Vertical | The diagonal |
|-------------|------|------------|----------|--------------|
| Cameraman   | clear| 0.9335     | 0.9591   | 0.9084       |
|             | cipher| 0.002104122| -0.00191233| 0.000779237 |
|             | clear| 0.9696     | 0.9733   | 0.9434       |
|             | cipher| -0.006430639| 0.000770385| -0.000602089 |
| Peppers     | clear| 0.9779     | 0.9651   | 0.9480       |
|             | cipher| 0.005502313| -0.00669286| 0.007525303 |
|             | clear| 0.9756     | 0.9730   | 0.9693       |
|             | cipher| -0.00075116| -0.00087152| -0.002021245 |
| House       | clear| 0.7593     | 0.7792   | 0.6978       |
|             | cipher| -0.000281916| 0.000535188| 4.24062e-05 |
| Elaine      | clear| 0.9410     | 0.9647   | 0.9143       |
| Testpart.1k | cipher| -0.003755791| 0.008675043| -0.001064676 |
| 5.1.09      | clear| 0.9059     | 0.8602   | 0.8244       |
|             | cipher| 0.001545614| 0.003576023| 0.000872724 |
|             | clear| 0.9506     | 0.9669   | 0.9197       |
|             | cipher| -0.000182157| -0.00566253| 0.003209531 |
| 5.1.11      | clear| 0.9565     | 0.9741   | 0.9143       |
|             | cipher| -0.002555087| -0.00040431| -0.003547135 |
| 5.1.12      | clear| 0.8719     | 0.8665   | 0.7559       |
|             | cipher| -0.004860921| 8.43753E-05| 0.000756657 |
|             | clear| 0.9462     | 0.8983   | 0.8529       |
|             | cipher| -0.002554441| -0.0010326| 0.000511298 |
| 5.1.14      | clear| 0.9364     | 0.8906   | 0.8533       |
|             | cipher| 0.000155616| 0.000738565| 0.001981233 |
|             | clear| 0.9000     | 0.8675   | 0.8102       |
|             | cipher| -0.000344055| 0.00077819| 0.002163414 |
| 5.2.08      | clear| 0.9402     | 0.9284   | 0.8983       |
|             | cipher| 0.002211752| -0.00320931| -0.002336888 |
|             | clear| 0.9774     | 0.9813   | 0.9671       |
|             | cipher| 0.002213779| -0.00154602| -0.002029902 |
| 5.3.01      | clear| 0.9099     | 0.9034   | 0.8591       |
|             | cipher| 0.000688093| 0.001897935| -0.000424347 |
|             | clear| 0.9619     | 0.9217   | 0.9086       |
|             | cipher| 0.004460742| 0.003880249| 0.003774306 |
| 7.1.01      | clear| 0.9647     | 0.9472   | 0.9452       |
|             | cipher| 0.00150668| 0.002688341| -0.002284199 |
TABLE 2. (Continued.) Correlation between plaintext image and adjacent pixels of ciphertext image.

|    | clear  | cipher    |    |    |
|----|--------|-----------|----|----|
| 7.1.03 | 0.9458 | 0.9322 | 0.9019    |   |
| 7.1.04 | 0.9769 | 0.9683 | 0.9573    |   |
| 7.1.05 | 0.9421 | 0.9122 | 0.8935    |   |
| 7.1.06 | 2.73305e-05 | 0.000913432 | -0.004015799 |   |
| 7.1.07 | 0.9403 | 0.9064 | 0.8862    |   |
| 7.1.08 | -0.001492885 | 0.000209264 | 3.108666e-05 |   |
| 7.1.09 | 0.8863 | 0.8779 | 0.8393    |   |
| 7.1.10 | -0.001273114 | 0.002897959 | -0.005358981 |   |
| 7.1.11 | 0.9577 | 0.9295 | 0.9221    |   |
| 7.1.12 | 0.9657 | 0.9308 | 0.9172    |   |
| 7.1.13 | 0.000734144 | -0.00307496 | 5.34782e-05 |   |
| 7.1.14 | 0.9644 | 0.948363 | 0.932196 |   |
| 7.1.15 | -0.000953789 | 0.000304602 | -0.000793932 |   |

FIGURE 11. Noise experiment. “Lena” of gray scale image of noise experiment with noise intensity (a) 0.01, (b) 0.05, (c) 0.1(d) 0.05 color image encrypted image after, Fig. 10(e)-(f) are corresponding decrypted images.

value distribution of the image. Statistical analysis attacks are easy to obtain a certain amount of information from images with uneven pixel distribution. If the histogram of the image is flat, that is, the pixel distribution is even, it is difficult to get information from statistical analysis attacks. In other words, the flatter the histogram of ciphertext images generated by a good encryption strategy, the better.
| Image       | Type     | New scheme | Ref. [35] | Ref. [36] | Ref. [29] | Ref. [37] |
|------------|----------|------------|-----------|-----------|-----------|-----------|
|            |          |            | 256×256   |           |           |           |
| Horizontal |          | 0.002104422| -0.0211  | 0.0063   | -0.0047  | -0.0009  |
| Cameraman   | vertical | -0.001912326| -0.0103 | -0.0142  | -0.0195  | -0.0223  |
| The diagonal|          | 0.000779237| 0.0054   | 0.0168   | 0.0279   | 0.0025   |
| 5.1.09     | Horizontal| -0.003755791| -0.0086 | -0.0066  | 0.0011   | -0.0063  |
|            | vertical | 0.008675043| -0.102   | -0.0089  | 0.0098   | 0.0109   |
| The diagonal|          | -0.001064676| 0.0125   | 0.0424   | -0.0227  | -0.0154  |
| 512×512    |          |            |          |          |          |          |
| Horizontal |          | -0.00075116| -0.0065  | -0.0191  | -0.0066  | -0.0062  |
| Elaine     | vertical | -0.00087152| -0.0096  | -0.0130  | -0.0019  | -0.0197  |
| The diagonal|          | -0.002021245| -0.0148 | -0.0096  | 0.0007   | -0.0169  |
| 1024×1024  |          |            |          |          |          |          |
| Horizontal |          | 0.002213779| 0.0040   | -0.0061  | 0.0094   | -0.0142  |
| 5.3.02     | vertical | -0.00154602| -0.0174  | -0.0079  | -0.0107  | -0.0141  |
| The diagonal|          | -0.002029902| -0.0135 | -0.0001  | -0.0007  | 0.0002   |
| Average    | vertical | 0.001086294| -0.034825| -0.011   | -0.0107  | -0.0113  |
| The diagonal|          | -0.001084147| -0.0026 | 0.012375 | 0.0013   | -0.0074  |

**Color image and binary image**

| Image         | Type     | New scheme | Ref. [52] | Ref. [12] |
|---------------|----------|------------|-----------|-----------|
| Horizontal    |          | 0.0034     | 0.0542   | -0.0163  |
| Color image (red) vertical |          | -3.5725e-05| -0.0068  | 0.0185   |
| The diagonal  |          | 0.0021     | 0.0319   | -0.0129  |
| Horizontal    |          | 9.9482e-04| -0.0331  | -0.0163  |
| Color image (blue) vertical |          | 8.1869e-04| 0.0125   | 0.0185   |
| The diagonal  |          | -2.5193e-04| -0.0236 | -0.0129  |
| Horizontal    |          | 0.0015     | 0.0562   | -0.0163  |
| Color image (green) vertical |          | -0.0012   | -0.0186  | 0.0185   |
| The diagonal  |          | 6.7506e-04| 0.0205   | -0.0129  |
| Horizontal    |          | 0.0018     | 0.0465   | 0.1007   |
| Binary image  | vertical | 0.0022     | -0.0087  | 0.1081   |
| The diagonal  |          | 3.7169 e-04| 0.0226  | 0.067    |
TABLE 4. Information entropy of ciphertext images.

| Image  | Information entropy |
|--------|---------------------|
| 5.1.09 | 7.9969              |
| 5.1.10 | 7.9971              |
| 5.1.11 | 7.9969              |
| 5.1.12 | 7.9972              |
| 5.1.13 | 7.9971              |
| 5.1.14 | 7.9973              |
| 5.2.08 | 7.9994              |
| 5.2.09 | 7.9993              |
| 5.2.10 | 7.9994              |
| 7.1.01 | 7.9993              |
| 7.1.02 | 7.9993              |
| 7.1.03 | 7.9994              |
| 7.1.04 | 7.9994              |
| 7.1.05 | 7.9994              |
| 7.1.06 | 7.9994              |
| 7.1.07 | 7.9993              |
| 7.1.08 | 7.9994              |
| 7.1.09 | 7.9993              |
| 7.1.10 | 7.9995              |
| boat.512 | 7.9993        |
| gray21.512 | 7.9994    |
| ruler.512 | 7.9993       |
| 5.3.01 | 7.9999              |
| 5.3.02 | 7.9999              |

D. \(\chi^2\) TEST

In order to analyze the distribution situation of ciphertext images more accurately, we exploit the chi-square test to obtain the quantitative numerical results, avoiding visual spoofing. In order to describe the experimental results more intuitively, we use the formula given in [50] for comparative analysis as Eq. (26).

\[
\chi^2 = \sum_{i=0}^{255} \frac{(q_i - q)^2}{q},
\]  

(26)

where, \(q_i\) represents the number of times the pixel value \(i\) appears in the image, and \(q\) is defined as:

\[
q = \frac{M \times N}{256}.
\]

(27)

Fig. 8 shows the histogram of clear text and ciphertext for “Lena”, “Elaine” and “Airport”, respectively. It can be seen that the histogram pixel distribution of the plaintext image is not uniform. On the contrary, histogram pixel distribution of ciphertext image is more uniform. Therefore, the ciphertext image obtained by the encryption algorithm in this paper can resist the attack of statistical analysis.

E. CORRELATION ANALYSIS

The high correlation of adjacent pixels indicates that the plaintext image is vulnerable to statistical analysis. Therefore, it is necessary to reduce the correlation between adjacent pixels. A good encryption algorithm can reduce the correlation between adjacent pixels. We randomly selected 10000 pixels from the plaintext and ciphertext images of the grayscale image “Lena”, and calculated the horizontal, vertical and diagonal correlations of adjacent pixels by using Eq. (28) and Eq. (29) [51], respectively. The calculated results are shown in Fig. 9.

\[
r_{xy} = \frac{\text{cov}(x, y)}{\sqrt{D(x)D(y)}},
\]

(28)

here:

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

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

(29)

In addition, the correlation coefficients of experimental images (plaintext and ciphertext) during the experiment and the comparison results with the correlation coefficients of ciphertext images in algorithm [29], [35]–[37] and algorithm [52], [53] are also given in Fig. 10, Table 2 and Table 3. The correlation between plaintext and ciphertext images is strong. However, the correlation between pixels of ciphertext images encrypted by the algorithm in this paper is much lower than the correlation coefficients of plaintext images. It can be seen from the comparison results between the algorithm in this paper and the other algorithm, that all the correlation coefficients of the algorithm in this paper are basically smaller than other algorithms. Therefore, this algorithm has a good ability to resist statistical analysis attacks.

F. INFORMATION ENTROPY ANALYSIS

1) GLOBAL INFORMATION ENTROPY

Information entropy is a commonly used index to reflect the randomness of information, and its calculation is...
TABLE 5. Comparison between ciphertext image information entropy and other literature results.

| Image    | New scheme | Ref. [35] | Ref. [36] | Ref. [29] | Ref. [37] |
|----------|------------|-----------|-----------|-----------|-----------|
|          |            | 256×256    |           |           |           |
| 5.1.09   | 7.9969     | 7.9991    | 7.9951    | 7.9965    | 7.9964    |
| Cameraman | 7.9973     | 7.9966    | 7.9955    | 7.9964    | 7.9990    |
|          |            | 512×512    |           |           |           |
| Elaine   | 7.9993     | 7.9972    | 7.9939    | 7.9971    | 7.9951    |
| 5.2.08   | 7.9994     | 7.9987    | 7.9951    | 7.9980    | 7.9956    |
|          |            | 1024×1024  |           |           |           |
| 5.3.02   | 7.9999     | 7.9966    | 7.9966    | 7.9969    | 7.9957    |
| Average  | 7.99856    | 7.99764   | 7.99524   | 7.99698   | 7.99636   |

Color image and binary image

| Image      | New scheme | Ref. [52] | Ref. [12] |
|------------|------------|-----------|-----------|
| Color image| 7.9998     | 7.9959    | 7.9925    |
| Binary image| 7.9969 | 7.9885    | 7.9921    |

Eq. (30). Theoretically, the possibility of information leakage decreases gradually as the information entropy approaches 8. Table 4 lists the information entropy of experimental images during the experiment. Table 5 shows the gray image entropy comparison results of this algorithm and algorithm [32]–[35]. The color image and binary image are compared with algorithm [49], [50] respectively.

$$H(s) = \sum_{i=0}^{2^n-1} p(m_i) \log_2 \frac{1}{p(m_i)}, \quad (30)$$

here $p(s_i)$ is the probability of $s_i$ happening.

The information entropy of the encrypted ciphertext image should be close to 8. As can be seen from Table 4, in the test images, the information entropy of ciphertext images obtained by the algorithm in this paper is close to 8. As can be seen from Table 5, the information entropy of both the algorithm in this paper and the algorithm [29], [35]–[37] is close to 8. In other words, the ciphertext images obtained in this paper are less likely to leak information and have better ability to resist statistical analysis attacks [45]–[47].

2) LOCAL SHANNON ENTROPY

For better analysis, we give experimental comparisons of local information entropy in this part. The calculation of local information entropy is Eq. (31).

$$H_{k,T_B}(S, L) = \frac{\sum_{i=1}^{k} H(S_{T_B}, L)}{k}, \quad (31)$$

where, $H(S_{T_B}, L)$ is the information entropy of non-overlapping image blocks $S_{T_B}$, $k$ and $T_B$ denote the number of blocks and pixels in each block. The $k$ is chosen as 30, $T_B$ is set as 1936, the significant $\alpha$ is selected as 0.05.

TABLE 6. Local Shannon entropy.

| Image          | Size       | Proposed |
|----------------|------------|----------|
| 5.1.10         | 256×256    | 7.9030089 |
| 5.1.11         | 256×256    | 7.9031685 |
| 5.1.12         | 256×256    | 7.9025458 |
| 5.1.13         | 256×256    | 7.9017190 |
| 5.1.14         | 256×256    | 7.9032276 |
| 5.2.08         | 512×512    | 7.9076276 |
| 5.2.09         | 512×512    | 7.9015359 |
| 5.2.10         | 512×512    | 7.9017094 |
| 7.1.01         | 512×512    | 7.8996862 |
| 7.1.02         | 512×512    | 7.9028140 |
| 7.1.03         | 512×512    | 7.9018914 |
| 7.1.04         | 512×512    | 7.9036226 |
| 7.1.05         | 512×512    | 7.9033111 |
| 7.1.06         | 512×512    | 7.9026073 |
| 7.1.07         | 512×512    | 7.9032934 |
| 7.1.08         | 512×512    | 7.9013717 |
| 7.1.09         | 512×512    | 7.9025729 |
| 7.1.10         | 512×512    | 7.9019517 |
| boat.512       | 512×512    | 7.9020965 |
| gray21.512     | 512×512    | 7.9030425 |
| ruder.512      | 512×512    | 7.9033066 |
| 5.3.01         | 1024×1024  | 7.9030218 |
| 5.3.02         | 1024×1024  | 7.9029454 |

PASS/ALL 22/24
TABLE 7. Comparison of NPCR(%) values in ciphertext images.

| Image    | New     | Ref. [53] | Ref. [54] | Ref. [55] | Ref. [56] | Ref. [57] | Ref. [48] |
|----------|---------|-----------|-----------|-----------|-----------|-----------|-----------|
| 256×256  | N_{655} ≥ 99.5693 |
| 5.1.09  | 99.6096 | 99.6338   | 99.5575   | 99.6109   | 99.6124   | 99.6170   | 99.6506   |
| 5.1.10  | 99.6118 | 99.6017   | 99.5544   | 99.5987   | 99.5972   | 99.6017   | 99.6063   |
| 5.1.11  | 99.6107 | 99.5956   | 99.8123   | 99.5697   | 99.5956   | 99.6460   | 99.6490   |
| 5.1.12  | 99.6098 | 99.6292   | 99.6109   | 99.6338   | 99.6017   | 99.6048   | 99.6170   |
| 5.1.13  | 99.6126 | 99.6109   | 99.7421   | 99.6201   | 99.6552   | 99.5758   | 99.5605   |
| 5.1.14  | 99.6170 | 99.6185   | 99.6933   | 99.6094   | 99.6002   | 99.5621   | 99.6216   |
| 512×512  | N_{655} ≥ 99.5893 |
| 5.2.08  | 99.6128 | 99.6059   | 99.6101   | 99.6056   | 99.6220   | 99.6292   | 99.5987   |
| 5.2.09  | 99.6131 | 99.6029   | 99.7025   | 99.6410   | 99.6208   | 99.6048   | 99.6220   |
| 5.2.10  | 99.6109 | 99.6010   | 99.6120   | 99.6067   | 99.5968   | 99.6155   | 99.6162   |
| 7.1.01  | 99.6212 | 99.6040   | 99.5190   | 99.6132   | 99.6161   | 99.6166   | 99.6166   |
| 7.1.02  | 99.617 | 99.5941   | 99.7200   | 99.6124   | 99.6140   | 99.5968   | 99.6109   |
| 7.1.03  | 99.6192 | 99.6132   | 99.4072   | 99.5983   | 99.6166   | 99.6075   | 99.6216   |
| 7.1.04  | 99.6131 | 99.6098   | 99.6037   | 99.6052   | 99.6227   | 99.6277   | 99.6090   |
| 7.1.05  | 99.6101 | 99.5995   | 99.4572   | 99.5842   | 99.5960   | 99.6284   | 99.6063   |
| 7.1.06  | 99.6135 | 99.6109   | 99.5213   | 99.6086   | 99.6212   | 99.6025   | 99.6101   |
| 7.1.07  | 99.6208 | 99.5934   | 99.5007   | 99.6193   | 99.6113   | 99.6037   | 99.6220   |
| 7.1.08  | 99.6315 | 99.6098   | 99.6902   | 99.5869   | 99.5914   | 99.6304   | 99.6101   |
| 7.1.09  | 99.6208 | 99.5815   | 99.7181   | 99.6094   | 99.6067   | 99.6193   | 99.5861   |
| 7.1.10  | 99.6143 | 99.5850   | 99.5163   | 99.6063   | 99.6056   | 99.6380   | 99.6120   |
| boat.512| 99.6265 | 99.6284   | 99.7128   | 99.6132   | 99.6021   | 99.6109   | 99.6086   |
| gray21.512| 99.6139 | 99.6014   | 99.6120   | 99.6162   | 99.6230   | 99.6109   | 99.6040   |
| ruler.512| 99.6135 | 99.6151   | 99.3118   | 99.6189   | 99.5930   | 99.6082   | 99.6227   |
| 1024×1024| N_{655} ≥ 99.5994 |
| 5.3.01  | 99.6106 | 99.6132   | 99.6040   | 99.6169   | 99.6100   | 99.6173   | 99.6099   |
| 5.3.02  | 99.6197 | 99.6107   | 99.4789   | 99.6010   | 99.5964   | 99.5981   | 99.6099   |
| Pass rate| 24/24 | 22/24 | 15/24 | 22/24 | 23/24 | 22/24 | 22/24 |

so an image passes the test if the obtained Local Shannon entropy falls within [7.901515698, 7.903422936]. Based on these settings, we can get the experimental results shown in Table 6.

G. DIFFERENTIAL ATTACK

Differential attack is a way to choose the plaintext attack. The attacker first makes a small change to the plaintext image, then uses the encryption algorithm to encrypt the two plaintext images respectively, and finally finds the correlation between the plaintext image and the ciphertext image by comparing the two ciphertext images. The Number of Pixels Change Rate (NPCR) and the Unified Average Changing Intensity (UACI) are two important indicators to evaluate the difference attack. Their formulas are respectively Eqs. (32)-(34) [42], as follows:

\[
D(i,j) = \begin{cases} 
0, & c_1(i,j) = c_2(i,j) \\
1, & c_1(i,j) \neq c_2(i,j).
\end{cases}
\]

(32)

\[
NPCR = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} D(i,j) \times 100\%.
\]

(33)
TABLE 8. Comparison of UACI(%) values in ciphertext images.

| Image     | New   | Ref. [53] | Ref. [54] | Ref. [55] | Ref. [56] | Ref. [57] | Ref. [48] |
|-----------|-------|-----------|-----------|-----------|-----------|-----------|-----------|
| 256x256   |       |           |           |           |           |           |           |
| 5.1.09    | 33.5167 | 33.5119   | 33.7574 | 33.5527   | 33.4652   | 33.2580   | 33.4387   |
| 5.1.10    | 33.4908 | 33.4263   | 33.1739 | 33.4381   | 33.5448   | 33.2845   | 33.4701   |
| 5.1.11    | 33.5008 | 33.4192   | 33.3198 | 33.4390   | 33.4161   | 33.5448   | 33.4150   |
| 5.1.12    | 33.4754 | 33.2672   | 33.6656 | 33.4373   | 33.2152   | 33.3578   | 33.5082   |
| 5.1.13    | 33.4708 | 33.4252   | 34.3306 | 33.3488   | 33.6499   | 33.7371   | 33.4939   |
| 5.1.14    | 33.5672 | 33.2919   | 33.1888 | 33.5133   | 33.5554   | 33.2361   | 33.7240   |
| 512x512   |       |           |           |           |           |           |           |
| 5.2.08    | 33.5492 | 33.4509   | 32.7443 | 33.4377   | 33.4575   | 33.2432   | 33.4694   |
| 5.2.09    | 33.5607 | 33.4543   | 34.0963 | 33.4939   | 33.4175   | 33.5176   | 33.4704   |
| 5.2.10    | 33.5461 | 33.4365   | 33.4982 | 33.3888   | 33.4315   | 33.5656   | 33.5688   |
| 7.1.01    | 33.5472 | 33.4811   | 33.5512 | 33.5553   | 33.5150   | 33.1477   | 33.4531   |
| 7.1.02    | 33.5181 | 33.4762   | 33.0872 | 33.4342   | 33.5221   | 33.4418   | 33.3931   |
| 7.1.03    | 33.5226 | 33.5346   | 33.7230 | 33.4585   | 33.4777   | 33.2279   | 33.4599   |
| 7.1.04    | 33.5269 | 33.3450   | 33.6036 | 33.4830   | 33.4721   | 33.1993   | 33.4471   |
| 7.1.05    | 33.5054 | 33.5380   | 33.1520 | 33.4393   | 33.4757   | 33.2974   | 33.3758   |
| 7.1.06    | 33.5188 | 33.4766   | 33.8290 | 33.5634   | 33.5035   | 33.3352   | 33.4942   |
| 7.1.07    | 33.5442 | 33.4695   | 33.5833 | 33.5241   | 33.4317   | 33.2157   | 33.4876   |
| 7.1.08    | 33.5134 | 33.4258   | 33.4212 | 33.4251   | 33.4274   | 33.2077   | 33.5078   |
| 7.1.09    | 33.5358 | 33.4954   | 32.8751 | 33.4606   | 33.4452   | 33.2849   | 33.4584   |
| 7.1.10    | 33.5147 | 33.4389   | 32.9976 | 33.4119   | 33.4434   | 33.1952   | 33.4332   |
| boat.512  | 33.5557 | 33.4693   | 33.9503 | 33.4993   | 33.4059   | 33.3673   | 33.4197   |
| gray21.512| 33.5007 | 33.4667   | 33.4646 | 33.4634   | 33.4554   | 33.4830   | 33.4906   |
| ruler.512 | 33.5151 | 33.5154   | 33.4628 | 33.5090   | 33.4795   | 33.7365   | 33.5193   |
| 1024x1024 |       |           |           |           |           |           |           |
| 5.3.01    | 33.5029 | 33.4973   | 32.9559 | 33.4698   | 33.4516   | 33.4406   | 33.4413   |
| 5.3.02    | 33.4880 | 33.5109   | 33.9252 | 33.4820   | 33.4579   | 33.3072   | 33.5189   |
| Pass rate | 24/24  | 21/24     | 6/24     | 22/24     | 23/24     | 7/24      | 21/24     |

Color image and binary image

| Image     | New   | Ref. [49] | Ref. [12] |
|-----------|-------|-----------|-----------|
| Color image | 33.5041 | 33.1273   | 33.0341   |
| Binary image | 33.6221 | 33.1412   | 33.0021   |

The critical value of NPCR with the significance level $\alpha$ is marked as $N^*_a$, which can be calculated according to the following formula.

$$N^*_a = \frac{L - \Phi^{-1}(\alpha)\sqrt{L/MN}}{L + 1},$$

(35)

where, $\alpha$ is the significance level which is set as 0.05 in our experiment, $\Phi^{-1}(\alpha)$ is the inverse cumulative density function of normal distribution $N(0, 1)$.

The lower bound of UACI with the significance level $\alpha$ is marked as $U^*_a-$ and the upper bound is marked as $U^*_a+$, which can be calculated according to the following formulas.

$$U^*_a- = \mu_u - \Phi^{-1}(\alpha/2) \times \sigma_u,$$

$$U^*_a+ = \mu_u + \Phi^{-1}(\alpha/2) \times \sigma_u,$$

(36)

where,

$$\mu_u = \frac{L + 2}{3L + 3},$$

(37)

$$\sigma^2_u = \frac{(L + 2)(L^2 + 2L + 3)}{18MNL(L + 1)^2},$$

(38)
X. Yan et al.: Chaotic Image Encryption Algorithm

FIGURE 12. Cropping attack experiment. Block the attack experiment respectively block cipher text image “Lena” (a) 1.526%, (b) 6.25%, (c) 25%, (d) 25% of the information, Fig. 12(e)-12(h) is the corresponding decryption image.

and \( L \) is grey level, \( M \) and \( N \) represent the sizes of the image.

In theory, if NPCR and UACI values are close to 99.6093% and 33.4635%, respectively, the results are good. Under the condition that the key remains unchanged, \( c_1 \) and \( c_2 \) are encrypted by the encryption algorithm in this paper. Then the NPCR and UACI of the two ciphertext images were calculated. Table 7 and Table 8 respectively show the NPCR value and UACI value obtained by this algorithm and the comparison results with other algorithms, and the gray image is compared with the algorithm [51]–[56]. Color image and binary image are compared with algorithms [49], [50]. It can be seen that, compared with other algorithms, most of the NPCR and UACI values obtained by this algorithm are close to 99.6093% and 33.4635%.

H. ROBUSTNESS ANALYSIS

Robustness is an important index to measure the anti-interference ability of a cryptographic system. We test the robustness of this algorithm by noise attack and block attack respectively.

Noise interference is an important problem in practical communication. The common noise types include gaussian noise, pepper and salt noise, etc. Compared with other noises, salt and pepper noise has more influence on ciphertext image. Therefore, this paper mainly considers the effect of adding salt and pepper noise to the plaintext image. Under the condition of keeping the key unchanged, we added the pepper and salt noise to the plaintext image with different intensity, and encrypted and decrypted the image after adding the noise respectively with the algorithm in this paper. Fig. 11 shows the encrypted and decrypted images of the plaintext image “Lena” with pepper and salt noise intensity of 0.01, 0.05 and 0.1, respectively. It can be seen that, even if the noise intensity reaches 0.1, the plaintext image can still be distinguished from the decrypted image.

Therefore, the algorithm in this paper has a good ability to resist noise attack.

Cropping attacks is another test of robustness. The experimental method is to first block the information of different sizes of an encrypted ciphertext image, and then try to decrypt it with the algorithm. Obviously, with the increasing of blocking information, the quality of decrypted images gradually decreases. Fig. 12 shows the decrypted images obtained by blocking 1.5625%, 6.25%, 25% and color image 25% of the ciphertext image “Lena” respectively. It can be seen that the main information of the plaintext image “Lena” can still be distinguished from the decrypted image. Therefore, the algorithm in this paper has a good ability to resist blocking attacks. Therefore, the proposed algorithm is robust.

I. TIME COMPLEXITY ANALYSIS

As is known to all, speed is an important index of encryption algorithm, which not only verifies its security, but also verifies its feasibility. Encryption speed can reflect the utility of the algorithm. In this algorithm, \( O (MN) \) is the time complexity, and \( N \) is the width and height of the plane image. In the scrambling phase, we need order \( O (1/4 MN) \) time complexity. In the diffusion stage, the time complexity is \( O (3/4 MN) \), so the total time complexity of the proposed encryption algorithm is \( O (MN) \). In general, the encryption speed depends on CPU, GPU, compiler, etc. MATLAB is a
more appropriate simulation tool, but its simulation efficiency is low. We propose a more reliable image encryption algorithm rather than a concrete implementation, which does not violate the original intention of this paper. In this paper, all image encryption algorithms are implemented using MATLAB R2017a and run on a computer with 8.0GB memory and a CPU of Intel(R) Core (TM) i5-2.30hz. The comparison of time complexity between gray image, color image and binary image is shown in Table 9.

| Test image | Proposed | Ref. [49] | Ref. [50] |
|------------|----------|-----------|-----------|
| Gray       | 0.4222   | 0.55      | 10.84     |
| color      | 0.6503   | 0.65      | 16.03     |
| Binary     | 0.4304   | 0.55      | 10.45     |

VII. CONCLUSION

In this paper, a chaotic image encryption algorithm is proposed based on the fractional wavelet transform. First, the image is preprocessed based on the fractional wavelet transform, and the image after three-level wavelet decomposition is divided into two types of components, high-frequency components and low-frequency components. To scramble two types of components, this paper proposes a new 3D cyclic displacement scrambling method, which is used in conjunction with chaos scrambling to scramble the two types of components. After scrambling is completed, in order to deepen the effect of encryption, pixel diffusion of the image matrix is performed based on the cyclic shift operation, and finally an encrypted image is obtained. In this paper, the Lorenz system is used to ensure the chaotic characteristics of the chaotic sequence, and the initial value of the chaotic system is calculated according to the SHA-512 hash algorithm and the plain text image. Compared with other algorithms, it can be found that the algorithm in this paper can achieve very good encryption effects and can resist common attacks. The innovation of this paper lies in applying the fractional wavelet to the preprocessing of the image, treating the image as a type of signal, and proposing a new scrambling and diffusion method. Scrambling and diffusion based on cyclic displacement can complete the encryption process well. The encrypted result has good security.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

REFERENCES

[1] X. Wang, L. Feng, and H. Zhao. “Fast image encryption algorithm based on parallel computing system,” Inf. Sci., vol. 486, pp. 340–358, Jun. 2019.

[2] W. Xingyuan, F. Le, W. Shihing, C. Zhang, and Z. Yingqian. “Spatiotemporal chaos in coupled logistic map lattice with dynamic coupling coefficient and its application in image encryption.” IEEE Access, vol. 6, pp. 39705–39724, 2018.

[3] X.-Y. Wang, S.-X. Gu, and Y.-Q. Zhang. “Novel image encryption algorithm based on cycle shift and chaotic system,” Opt. Lasers Eng., vol. 68, pp. 126–134, May 2015.

[4] X. Wang, L. Liu, and Y. Zhang. “A novel chaotic block image encryption algorithm based on dynamic random growth technique,” Opt. Lasers Eng., vol. 66, pp. 10–18, Mar. 2015.

[5] X. Wang and S. Gao, “Image encryption algorithm based on the matrix semi-tensor product with a compound secret key produced by a Boolean network,” Inf. Sci., vol. 539, pp. 195–214, Oct. 2020.

[6] X.-J. Tong, P. Chen, and M. Zhang. “A joint image lossless compression and encryption method based on chaotic map,” Multimedia Tools Appl., vol. 76, no. 12, pp. 13995–14020, Jun. 2017.

[7] H. Zhu, C. Zhao, and X. Zhang. “A novel image encryption–compression scheme using hyper-chaos and Chinese remainder theorem,” Signal Process., Image Commun., vol. 28, no. 6, pp. 670–680, Jul. 2013.

[8] C. Zhu and K. Sun. “Cryptanalyzing and improving a novel color image encryption algorithm using RT-enhanced chaotic tent maps,” IEEE Access, vol. 6, pp. 18759–18770, 2018.

[9] J. S. Khan, J. Ahmad, and M. A. Khan. “TD-ERCS map-based confusion and diffusion of autocorrelated data,” Nonlinear Dyn., vol. 87, no. 1, pp. 1–15, 2016.

[10] J. Sun, G. Han, Z. Zeng, and Y. Wang. “Memristor-based neural network circuit of full-function Pavlov associative memory with time delay and variable learning rate,” IEEE Trans. Cybern., vol. 50, no. 7, pp. 2935–2945, Jul. 2020.

[11] Z. Hua, Y. Zhang, and Y. Zhou. “Two-dimensional modular chaotification system for improving chaos complexity,” IEEE Trans. Signal Process., vol. 68, pp. 1937–1949, 2020.

[12] J. S. Khan and J. Ahmad. “Chaos based efficient selective image encryption,” Multidimensional Syst. Signal Process., vol. 30, no. 2, pp. 943–961, 2019.

[13] S. F. Abbasi, J. Ahmad, J. S. Khan, M. A. Khan, and S. A. Sheikh. “Visual meaningful encryption scheme using intertwinning logistic map,” in Intelligent Computing, vol. 857, 2019, pp. 764–773.

[14] A. Alfalou, C. Brossseau, and N. Abdallah. “Assessing the performance of a method of simultaneous compression and encryption of multiple images and its resistance against various attacks,” Opt. Express, vol. 21, no. 7, pp. 8025–8043, 2013.

[15] J. Sun, X. Zhao, J. Fang, and Y. Wang. “Autonomous memristor chaotic systems of infinite chaotic attractors and circuitry realization,” Nonlinear Dyn., vol. 94, pp. 2878–2879, Aug. 2018.

[16] Z. Hua and Y. Zhou. “Exponential chaotic model for generating robust chaos,” IEEE Trans. Syst., Man, Cybern. Syst., early access, Aug. 28, 2019, doi: 10.1109/TSMCB.2019.2932616.

[17] Y. Kong and H.-J. Lu. “Time-varying neural networks for dynamical systems modeling with application to image compression,” Int. J. Satur. Appl., vol. 10, no. 10, pp. 323–334, Dec. 2016.

[18] K. Jafari, R. H. H. Gholami, and I. Moon. “Lossless and lossy compression of quantitative phase images of red blood cells obtained by digital holographic imaging,” Appl. Opt., vol. 55, no. 36, pp. 10409–10416, 2016.

[19] J. Sun, Y. Wu, G. Cui, and Y. Wang. “Finite-time real combination synchronization of three complex-variable chaotic systems with unknown parameters via sliding mode control,” Nonlinear Dyn., vol. 88, no. 3, pp. 1677–1690, May 2017.

[20] C. Samson and V. K. S. Satriyo. “An RGB image encryption supported by wavelet-based lossless compression,” Int. J. Adv. Comput. Sci. Appl., vol. 9, no. 3, pp. 39–50, Jan. 2014.
A. A. Abd El-Latif, L. Li, N. Wang, Q. Han, and X. Niu, “A new approach to chaotic image encryption based on quantum chaotic system,” Inf. Sci., vol. 507, pp. 16–36, Jan. 2020.

A.-V. Diaconu, “Circular inter–intra pixels bit-level permutation and chaos-based image encryption,” Inf. Sci., vol. 355–356, pp. 314–327, Aug. 2016.

Z. Y. Hua, Y. Zhou, C.-M. Pun, and C. L. P. Chen, “2D sine logistic modulation map for image encryption,” Inf. Sci., vol. 297, pp. 80–94, Mar. 2015.

N. Sharma, I. Saini, A. Yadav, and P. Singh, “Phase-image encryption based on 3D-Lorenz chaotic system and double random phase encoding,” 3D Res., vol. 8, no. 4, p. 39, Dec. 2017.

Y. Zhang, D. Xiao, W. Wen, and K.-W. Wong, “On the security of symmetric ciphers based on DNA coding,” Inf. Sci., vol. 289, pp. 254–261, Dec. 2014.

Z. L. Zhu, W. Zhang, K.-W. Wong, and H. Yu, “A chaos-based symmetric image encryption scheme using a bit-level permutation,” Inf. Sci., vol. 181, no. 6, pp. 1171–1186, 2011.

X. Wang, Y. Hou, S. Wang, and R. Li, “A new image encryption algorithm based on CML and DNA sequence,” IEEE Access, vol. 6, pp. 62272–62285, 2018.

Y. Wu, H. Kan, and J. Kurths, “A new color image encryption scheme based on DNA sequences and multiple improved 1D chaotic maps,” Appl. Soft Comput., vol. 37, pp. 24–39, Dec. 2015.

X. Chai, X. Fu, Z. Gan, Y. Lu, and Y. Chen, “A color image cryptosystem based on dynamic DNA encryption and chaos,” Signal Process., vol. 155, pp. 44–62, Feb. 2019.

J. Wu, X. Liao, and B. Yang, “Color image encryption based on chaotic systems and elliptic curve ElGamal scheme,” Signal Process., vol. 141, pp. 109–124, Dec. 2017.

M. Alawida, A. Samsudin, J. S. Teh, and R. S. Alkhawaldeh, “A new hybrid digital chaotic system with applications in image encryption,” Signal Process., vol. 160, pp. 45–58, Jul. 2019.

P. Rakheja, R. Vig, and P. Singh, “An asymmetric watermarking scheme based on random decomposition in hybrid multi-resolution wavelet domain using 3D Lorenz chaotic system,” Optik, vol. 198, Dec. 2019.

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, Sep. 2019.

F. P. An and J. E. Liu, “Image encryption algorithm based on adaptive wavelet chaos,” J. Sensors, vol. 2019, pp. 1–12, Dec. 2019.

C. Cao, K. Sun, and W. Liu, “A novel bit-level image encryption algorithm based on 2D-LCM hyperchaotic map,” Signal Process., vol. 143, pp. 122–133, Feb. 2018.

W. Cao, Y. Mao, and Y. Zhou, “Designing a 2D infinite collapse map for image encryption,” Signal Process., vol. 171, Jun. 2020, Art. no. 107457.

S. Tedmori and N. Al-Najdawi, “Image cryptographic algorithm based on the Haar wavelet transform,” Inf. Sci., vol. 269, pp. 21–34, Jun. 2014.

A. Ajish, “Wavelet based advanced encryption standard algorithm for image encryption,” Int. J. Eng. Res. Gen. Sci., vol. 3, no. 1, pp. 943–950, 2015.

Z. Liu, J. Dai, X. Sun, and S. Liu, “Triple image encryption scheme in fractional Fourier transform domains,” Opt. Commun., vol. 282, no. 4, pp. 518–522, Feb. 2009.

F. Ozkaynak, “Brief review on application of nonlinear dynamics in image encryption,” Nonlinear Dyn., vol. 92, no. 2, pp. 305–313, Apr. 2018.

Y.-Q. Zhang and X.-Y. Wang, “A new image encryption algorithm based on non-adjacent coupled map lattices,” Appl. Soft Comput., vol. 26, pp. 10–20, Jan. 2015.

Y. Wu, Y. Zhou, J. P. Noonan, and S. Agaian, “Design of image cipher using Latin squares,” Inf. Sci., vol. 264, pp. 317–339, Apr. 2014.

E. N. Lorenz, “Deterministic nonperiodic flow,” J. Atmos. Sci., vol. 20, no. 2, pp. 130–141, 1963.

A. A. Abd El-Latif, L. Li, N. Wang, Q. Han, and X. Niu, “A new approach to chaotic image encryption based on quantum chaotic system, exploiting color spaces,” Signal Process., vol. 93, no. 11, pp. 2986–3000, Nov. 2013.

YONGJIN XIAN was born in Liaoning, China, in 1989. He received the B.S. and B.E. degrees in software engineering from Dalian Jiaotong University, China, in 2012, and the M.S. degree in mathematics from Dalian Jiaotong University, China, in 2018. He is currently pursuing the Ph.D. degree with the Computer Application Technology, Dalian Maritime University, China. His main research interests include chaotic cryptography, information security, and systems biology.