IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 25, 2019, accepted January 3, 2020, date of publication January 10, 2020, date of current version January 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2965740

Color Image Encryption Algorithm Based on Dynamic Chaos and Matrix Convolution

XIANCHENG HU, LIANSUO WEI, WEI CHEN, QIQI CHEN, AND YUAN GUO
College of Computer and Control Engineering, Qiqihar University, Qiqihar 161006, China

Corresponding author: Liansuo Wei (wlsaaaaaa@163.com)

This work was supported in part by the National Science foundation of China under Grant 61571150 and Grant 61872204, in part by the Heilongjiang Provincial Natural Science Fund under Grant LH2019F037, in part by the Heilongjiang Provincial Education Department Surface Scientific Research Project under Grant 135109237 and Grant 135209235, and in part by the Heilongjiang Province Higher Education Teaching Reform Project under Grant SJGY20170386, Grant SJGY20180567, and Grant GBC1317212.

ABSTRACT This paper proposes a color image encryption algorithm based on a cloud model Fibonacci chaotic system, as well as a matrix convolution operation that can protect image content effectively and safely. The algorithm combines the cloud model with the generalized Fibonacci, creating a new complex chaotic system that realizes the dynamic random variation of chaotic sequences. The chaotic sequence is used to scramble the pixel coordinates of the mosaic images of the R, G, and B components of the color image. Then, the chaotic sequence value is used as a matrix convolution cloud algorithm that alternately updates the input value of the matrix convolution operation and the pixel value to obtain the permutation transformation of the original pixel value. Finally, the pixel values of the replacement and cloud model Fibonacci chaotic sequence and the pixel values of the front (rear) adjacent pixel points are subjected to a two-way exclusive XOR operation. Realizing the change of the arbitrary pixel value causes a chain transformation of the pixel values of all of the pixel points, and sequentially generates an encrypted image. Experiments show that the histogram of the encrypted image is smoother and adjacent pixels of the image have low correlation. In addition, this algorithm can resist attack experiments such as differential attack, select plaintext attack and noise attack and provides high encryption security, high anti-interference, and strong robustness. The dynamic chaotic system is used to realize the color image encryption of the dynamic key, and the encryption algorithm has higher security and the validity of the algorithm.

INDEX TERMS Cloud model, Fibonacci chaotic system, matrix convolution algorithm, color image, encryption.

I. INTRODUCTION

With the rapid development of information technology, the security of multimedia data such as images, videos, and audio has attracted widespread attention. Secure and efficient encryption of image information is the focus of much multimedia research. Due to the low entropy of digital images, as well as strong pixel correlation and high redundancy, traditional encryption methods typically cannot efficiently encrypt image information. The design of new cryptographic algorithms based on chaotic systems has become an attractive image encryption solution. Many new image encryption algorithms or methods have been proposed, such as image passwords based on chaotic systems [1]–[16].

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

A chaotic sequence is created by a chaotic map as a random sequence. Chaotic sequences are complex in structure and cannot be analyzed and predicted [7], [8], so they are widely used in image encryption. Many chaos-based encryption algorithms have been extensively studied and applied to the two steps of the common chaotic encryption scheme—scrambling and diffusion.

A common encryption method is to use chaotic sequences to scramble the plaintext image to change the pixel position. At the same time, since the pixel value of the pixel is changed, the original image information cannot be recognized. Finally, the pixel points are diffused to hide the information of the plaintext pixel points in more ciphertext pixels, and the image information processing in steps can improve the encryption security. In [9] a novel image encryption scheme using a 3-D Arnold cat map and the Fisher-Yates shuffling algorithm is

presented. A plain image is divided into various slices of equal size and then the 3-D representation of the image is shuffled by the 3-D chaotic map. In [10] a novel chaotic image encryption algorithm based on a content-sensitive dynamic function switching mechanism is presented. The proposed encryption algorithm is a symmetric cipher, operating on a 1D byte sequence. Three independent chaotic maps are used for scrambling plain image bytes in order to realize confusion and diffusion properties. In [11] a chaos-based image encryption algorithm is proposed. The replacement stage of the algorithm uses three maps—standard map, cat map, and Baker map—which provide higher security and high encryption speed, thereby enabling more practical applications. In [12] an effective chaos-based encryption algorithm specialized for images is proposed. A system of two independent chaotic functions with high sensitivity to initial states is used to sufficiently apply confusion and diffusion principles for images with any entropy.

In addition to dividing the encryption process into scrambling and diffusion processes, some researchers have also proposed encryption methods that improve the scrambling or diffusion methods. In [13] a new symmetric chaotic encryption algorithm based on bitmap permutation is proposed, which results in longer processing time and large computational complexity. In [14] a simple table look-up and swapping techniques are suggested for use in the diffusion phase, rather than the common use of a 1D chaotic map, wherein the real number arithmetic operation and the subsequent quantization step pose time-consuming limitations. In [15] a new two-way diffusion technique is proposed. In addition to the conventional diffusion process, the pixel values are from top to bottom and right to left in the second step. After modification, there are fewer rounds required, thereby resulting in shorter calculation times, and a higher level of security. In [16] a continuous diffusion technique is proposed, which adds complementary diffusion in addition to performing the conventional diffusion process. This approach has a sufficient level of safety in a small number of rounds, thus greatly reducing the time required. In [17] a new chaotic block image encryption algorithm based on dynamic random growth technology was proposed. In the diffusion process, an intermediate parameter is calculated according to the image block. The intermediate parameter is used as the initial parameter of chaotic map to generate random data stream, which can resist the chosen plaintext attack.

More recently, a new chaotic system encryption method combining a new algorithm with chaos has emerged. In [18] the suggested scheme consists of two iterative modules: first, a permutation module that is based on a nonlinear inter-pixel computing and swapping procedure (NICSP), and next, a diffusion phase that is governed by a snake-like mode, realized in the reverse order. In [19] the suggested scheme is composed of two phases—shuffling phase and masking phase. The encryption is block-based and is performed through the use of chaotic cat maps. Hybrid technology has then been added to ensure resistance to common. In [20] a new IE algorithm which can transform an original image into a visually meaningful cipher image is presented. The advantage of this method is that the generated visually meaningful cryptographic image does not attract the attention of the attacker, but it still has traces of texture features, which may increase the security risk to some extent. In [21] a hybrid image compression encryption algorithm based on a key-controlled measurement matrix is proposed. In [22] color space rotation is used to hide original color information. In [23], a new image encryption method based on matrix semi-tensor product theory is proposed. The hyperchaotic Lorenz system is used to generate chaotic sequences to scramble the matrix. Then the semi-tensor product method is used for diffusion, and finally an encrypted image is obtained. This method breaks the shackles of traditional matrix operations, not only has high security, but also improves encryption efficiency.

To improve security encryption, increase the randomness of chaos, and expand the key space, the combined chaotic system has many applications because it tends to overcome the disadvantages of 1D and higher order chaotic maps [24]–[30]. In [26] three 1D combined chaotic maps are proposed to achieve the tradeoff between computational cost and security. This system achieved better encryption results with reduced computational cost compared with higher order chaotic maps. In [27] a scheme is proposed that uses three chaotic maps—logistic, tent, and sine maps. The crossover unit extensively permutes the image pixels row-wise and column-wise based on the chaotic key streams generated from the Combined Logistic–Tent (CLT) system. The decomposed images are then mutated by XOR operation with quantized chaotic sequences from the Combined Logistic–Sine (CLS) system. In [28] the larger key space is achieved by combining complex Chen and complex Lorentz maps. In [29] a new encryption scheme based on chaotic mapping combined with Tinkerbell chaotic mapping is presented. In [30] a novel one-dimensional logistic-PWLCM (LP) modulation map that is derived from the logistic and PWLCM maps is presented.

The traditional single chaotic system is too simple, which directly affects the effect of scrambling and diffusion, and there are few methods for image pixel information replacement. Therefore, we hope to study a new kind of chaotic system, and generate a chaotic sequence with strong randomness by combining chaotic systems to improve the security of encryption, and replace traditional matrix operations with mathematical algorithms, and replace image information with new results.

To achieve more robust encryption, and to leverage the advantages derived from the research described above, a color image encryption method based on the cloud model Fibonacci dynamic chaotic system combined with a matrix convolution operation is proposed. Chaotic sequence with random transformation of random seeds of the cloud model, repeated operation produces uncorrelated multiple sets of dynamic chaotic sequences, and it increases the chaotic randomness, expands the key space, and increases the chaotic range. Using hybrid chaotic sequences to scramble image

pixel points spliced by RGB components, and then combining matrix convolution operations in convolutional neural networks [31], the matrix convolution operation is used to permute the pixel points, and enhance the anti-attack ability of the ciphertext. Finally, XOR operation diffuses pixels with the chaotic sequence and the pre-adjacent pixel values, which achieves the overall design of position scrambling, numerical transformation, and diffusion of the image encryption process. The algorithm provides protection from an anti-plaintext attack as well as provides high encryption security.

II. CLOUD MODEL FIBONACCI CHAOTIC SYSTEM

The purpose of chaotic system design is to generate chaotic sequences and provide secure and reliable support for image encryption [32], [33].

A. CLOUD MODEL

The cloud model generator is mostly based on a pseudo-random number generator [34], and the cloud model has three numerical features of expectation Ex, entropy En and hyper-entropy He, which are used to represent the model of uncertainty transformation [35].

$$\text{Expectation}: Ex = \overline{X} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{1}$$

$$\text{Sample varianc}: S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{X})^2 \tag{2}$$

$$\text{Entropy}: En = \sqrt{\frac{\pi}{2}} \times \frac{1}{n}\sum_{i=1}^{n}|x_i - Ex| \tag{3}$$

$$\text{Hyper-entrop}: He = \sqrt{S^2 - En^2} \tag{4}$$

The Ex in (1) reflects the position of cloud center of gravity in cloud drop group, the En in (3) reflects the relationship between data fuzziness and randomness, the He in (4) is the uncertainty measure of En, reflecting the dispersion and thickness of cloud. The normal random numbe y_i is generated by the variance He^2 and the expected valu En, the normal random number x_i, the cloud droplet is generated by the expected value Ex and the variance y_i^2, which has the characteristics of randomness and stability tendency.

$$y_i = R_N(En, He) \tag{5}$$

$$x_i = R_N(Ex, y_i) \tag{6}$$

where R_N is a normal random number, and the algorithm cloud model value is $Ex = 5000$, $En = 3$, $He = 0.1$.

It can be seen from Fig.1 that the cloud droplet distribution does not have the characteristics of chaotic uniform distribution, but according to the characteristics of the cloud model, the cloud model data can be transformed through the transformation of random seeds. So we can combine the randomness of the cloud model with the chaotic sequence, which makes the chaotic sequence in the unpredictable state for a long time, and the sequence rules are difficult to find, which plays a very good role in the encryption application.

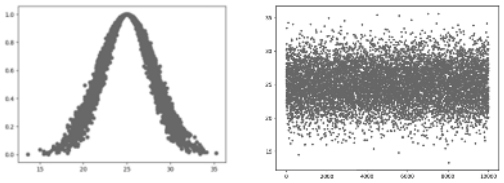


FIGURE 1. Cloud droplet distribution.

B. CONSTRUCTING CLOUD MODEL FIBONACCI CHAOTIC SYSTEM MODEL

The chaotic system uses Fibonacci to generate random numbers. The generated random numbers can overcome the correlation of the sequence itself. The Fibonacci sequence formula is shown in (7) [36]–[38]:

$$x_{i+1} = (x_i + x_{i-p}) \bmod M, \quad i = p, p+1, \cdots, M \in N \tag{7}$$

The Fibonacci sequence has the characteristics of being simple, fast and esy to implement, and the model adopts the generalized third-order Fibonacci function model:

$$F_j = (A_i F_{i-1} + B_i F_{i-2} + C_i F_{i-3}) \bmod M \tag{8}$$

where: A_i, B_i, and C_i represent random constants, which M are modules. Where F_j is cloud drop group.

The quantum logistic mapping produces a multi-dimensional sequence, which can dynamically replace the three parameters of the generalized third-order Fibonacci function [39]. The formula is:

$$\begin{cases} x_{n+1} = r(x_n - |x_n|^2) - ry_n \\ y_{n+1} = -y_n e^{-2\beta} + re^{-\beta} \\ \times[(2 - x_n - x_n^*)y_n - x_n z_n^* - x_n^* z_n] \\ z_{n+1} = -z_n e^{-2\beta} + re^{-\beta} \\ \times[2(1 - x_n^*)z_n - 2x_n y_n - x_n] \end{cases} \tag{9}$$

where: x_n, y_n, and z_n represent three sets of input values, β, r represent dissipation parameters and control parameters, x_n^* and z_n^* are conjugate complex numbers of x_n and z_n. This algorithm takes the initial value $x_0 = 0.3$, $y_0 = 0.06$, $z_0 = 0.2$, $r = 3.99$, $\beta = 6.2$ and then the sequence F_j is generated by a modulo operation.

Next, the construction of the cloud model Fibonacci chaotic system model is carried out. The model uses the quantum chaotic map as the random dynamic parameter to reduce the sequence correlation. Then it is coupled with the Logistic map after the generalized third-order Fibonacci function model operation. The cloud model Fibonacci chaotic system is obtained, as shown below:

$$X_{n+1} = A_{FQL} = (F(Q(\gamma, \beta))) + L(x_0, \mu) \bmod 1 \tag{10}$$

where $Q(\gamma, \beta)$ represents the quantum chaotic system, $F(Q(\gamma, \beta))$ is the chaotic sequence generated by (8), and $L(x_0, \mu)$ represents the Logistic chaotic system with an initial value x_0 and a parameter r. Finally, a new uncorrelated

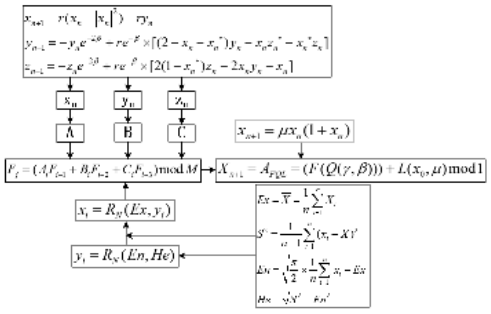


FIGURE 2. The schematic diagram of CFCS pseudo-random chaotic sequence generator.

chaotic sequence is generated by coupling with the Logistic map.

The cloud model Fibonacci chaotic system combines the quantum logistic, logistic, Fibonacci sequence and cloud model to make use of the simple, fast and easy-to-implement features of the Fibonacci sequence, as well as the chaos of multiple sets of mixed sequences and the normal distribution of the cloud model, constructing a new chaotic system, the system complexity is increased, and the time complexity is relatively increased.

The multi-group chaotic sequences generated by the Fibonacci chaotic system based on the cloud model are independent of each other, and multiple chaotic sequences are used in the process of color image encryption, such as the four chaotic sequences $F_1(i)$, $F_2(i)$, $F_3(i)$, $F_4(i)$ generated in the encryption process. However, the traditional chaotic system, such as logistic, is easy to deduce the rules and has low security.

The pseudo-random sequence generator designed according to the above steps has better pseudo-random characteristics, fast generation speed, independent non-repeating sequence, and compared with the traditional sequence generators such as Ten chaotic sequence generator, CNN chaotic sequence generator and Logistic chaotic sequence generator, it has higher security to generate even pseudo-random chaotic sequence. Fig.3(a) shows the sequence of pseudo-random chaos that produces uniform, Fig.3(b) presents the difference value graph of the two sequences and the difference value of interval variation shows that the algorithm has a strong sensitivity to the initial value of the key, Fig.3(c) is the histogram of the chaotic sequence, Fig.3(d) is a three-dimensional structure of the chaotic sequence.

C. STATISTICAL STOCHASTICITY ANALYSIS

Randomness is a crucial property both for Pseudo Random Number Generators and ciphers [40]. For cryptographic reasons, the output of these systems needs to satisfy randomness criteria measured by some statistical tests. Probability value (P-value) of each test should be greater than 0.1 for any bit sequence to be regarded as random. We used NIST SP800-22 to test the randomness of our key generator and our pre-encryption algorithm respectively. NIST randomness

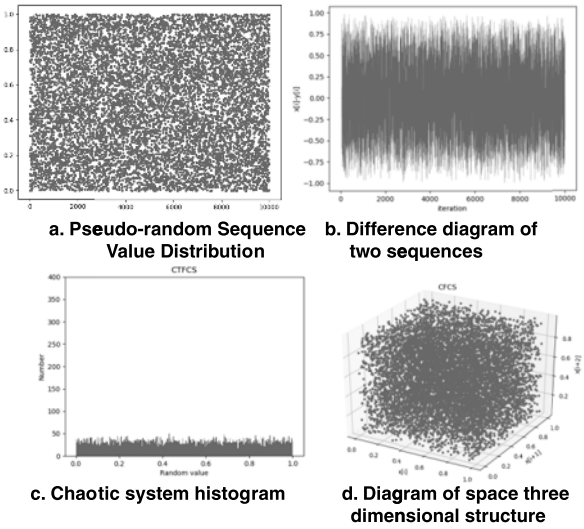


FIGURE 3. Chaotic sequence analysis.

TABLE 1. NIST statistical test results.

Test name	Average P-value	Result
Frequency	0.16103	Success
Block Frequency(m=12)	0.46484	Success
Cumulative Sums(Forward)	0.23687	Success
Cumulative Sums(Reverse)	0.20721	Success
Runs	0.66236	Success
Longest Run of Ones	0.31759	Success
Rank	0.50569	Success
Nonperiodic Template Matchings	0.49452	Success
Universal Statistical	0.67577	Success
Overlapping Template Matchings	0.39953	Success
Approximate Entropy	0.16166	Success
Random Excursions	0.47316	Success
Random Excursions Variant	0.43595	Success
Serial	0.49140	Success
Linear Complexity	0.56172	Success
FFT	0.27197	Success

test results for Peppers image of size 256 × 256 are given in Table 1.

It can be seen from the above table that the values of all 16 test items are greater than the significance level, and the values of 4 items (Runs test, Rank test, Universal Statistical test, Linear Complexity test) exceed 0.5, so the key stream sequence generated by the chaotic system proposed in this paper is random and has good security.

III. DESCRIPTION OF ENCRYPTION STEP ALGORITHM

A. SCRAMBLING ALGORITHM DESCRIPTION

In the image encryption system, scrambling can effectively disturb the original position of the pixel, destroy the original image information of the image, and map the chaotic sequence and the coordinates of the pixel to achieve scrambling of the pixel of the image. In order to ensure that the number of position coordinates corresponds, it is multiplied by the height and width of the plaintext image to expand,

and in order to scramble the security, the average value of the pixel points of the image is selected as the key, and the key is used as the initial value of the random seed and logistic map of the cloud model, respectively, to generate the cloud model Fibonacci chaotic sequence, and then the chaotic sequence is rounded to obtain an integer sequence between (*0, M*N*), and each element (*i=1, 2, ..., M*N*) in the integer sequence (x: line, y: column) is expressed in the form of coordinates. Finally, the chaotic coordinates are used to map the coordinates of the replacement pixels, and after adjusting the matrix, the scrambled is obtained, and the pixel position is scrambled.

The scrambling formula is as follows:

$$F_{11}(i) = \text{int}(F_1(i) \times height \times width) \tag{11}$$

$$\begin{cases} x_i = F_{11}(i)\%N \\ y_i = F_{11}(i)/N \\ Te = P_0[i,j] \\ P_0[i,j] = P_0[x_i, y_i] \\ P_0[x_i, y_i] = Te \end{cases} \tag{12}$$

where: x, y represent the row and column of each element in the chaotic sequence respectively, N represents the width of the image, and Te is the auxiliary variable, which is used to temporarily store the pixel value of the location that has been scrambled.

B. DESCRIPTION OF REPLACEMENT ALGORITHM

The scrambling changes the position of the pixel, and the pixel value of the original pixel does not change, and the replacement is to transform the original value of the pixel and replace the new data to conceal the real pixel value. Before the convolution operation, a column and a row of zero pixel points are respectively added to the rightmost and lowermost ends of the pixel matrix to avoid data loss when the convolution is replaced to the edge of the matrix.

The chaotic sequence $F_2(i)$ of Fibonacci chaotic system is extended by (13) to obtain $F_{22}(i)$, and put the chaotic sequence values into a 2×2 matrix to form a convolution kernel $filter(t)$, and then the matrix P_1 obtained in the second step is convoluted with the convolution kernel $filter(t)$, and finally multiplied by the random sequence $F_3(i)$ to obtain an image matrix P_2. The formula is as follows:

$$F_{22}(i) = \text{int}(F_2(i) \times 100), \quad i \in (0,1,2..., height \times width - 1) \tag{13}$$

$$P_2[i,j] = sum(\text{P}_1[i:i+2, j:j+2] \times filter(t)) - F_3(i) \tag{14}$$

where: $F_2(i)$ represents a chaotic sequence, $F_{33}(i)$ is a spread sequence, $F_3(i)$ represents another group of chaotic sequences, and Equation (1) is a convolution summation formula of the matrix P_1 and the convolution kernel.

The replacement process is to perform convolution operation on four pixel point values of 2×2 units in the pixel matrix and four chaotic sequence values in the convolution kernel, and at the same time, in the next operation, the 2×2 chaotic sequence value in the convolution kernel and the 2×2 unit in the pixel matrix are updated and replaced, and sequentially shifted to perform matrix convolution operation. The matrix size of the scheme is 2×2, which can not only change the original information, but also save the computational complexity. After the convolution replacement process is completed, the added row and column zero pixel values are removed. The matrix convolution operation flow is shown in Fig.5.

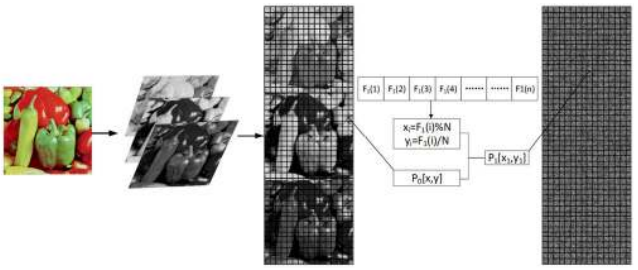


FIGURE 4. Scrambling process diagram.

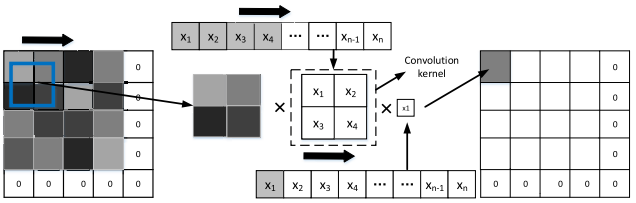


FIGURE 5. Convolutional replacement.

C. DESCRIPTION OF THE DIFFUSION ALGORITHM

In an image encryption system, diffusion means not to change the position of a pixel. By changing the gray value of the pixel, the information of any plain pixel is hidden in as many ciphertext pixels as possible, so that the pixel value information of any pixel affects the pixel values of other pixels as much as possible.

Random sequence extension:

$$F_{44}(i) = \text{int}(F_4(i) \times 10^8) \tag{15}$$

Forward diffusion:

$$\begin{cases} P_3[0] = P_2[0] \otimes F_{44}(0) \otimes \text{int}(image\_mean) \\ \quad i = 0 \\ P_3[i] = P_2[i] \otimes F_{44}(i) \otimes P_2[i-1] \\ \quad i \geq 1 \end{cases} \tag{16}$$

Back diffusion:

$$\begin{cases} P_3[\text{N} \times \text{M}] = P_2[\text{N} \times \text{M}] \otimes F_{44}(\text{N} \times \text{M}) \\ \otimes \text{int}(image\_mean) \\ P_3[i] = P_2[i] \otimes F_{44}(i) \otimes P_2[i+1] \quad i < \text{N} \times \text{M} \end{cases} \tag{17}$$

where: The value of 10^8 is derived from the range of the maximum value of the pixel after matrix convolution. *image_mean* represents the mean of the pixel values, $F_{44}(i)$ is the chaotic

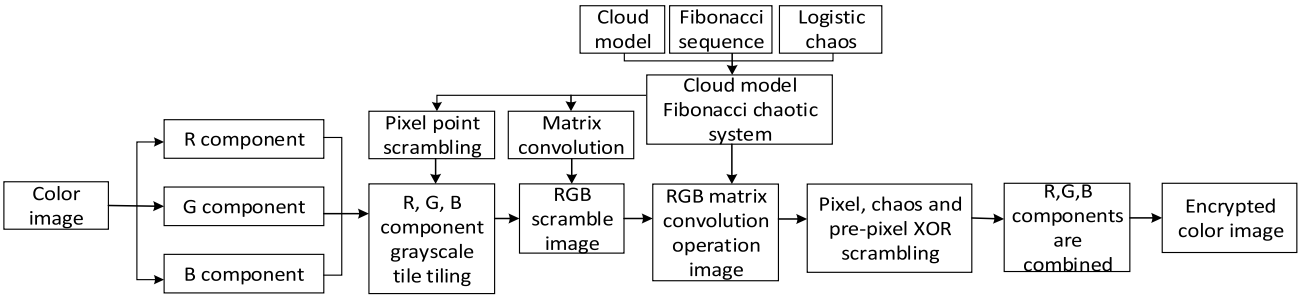


FIGURE 6. Encryption flow chart.

extension sequence, and the XOR process requires a pixel point, a chaotic sequence, and a front pixel point $P_2\,[i-1]$ or a back pixel point $P_2\,[i-1]$. When $i{=}0$ or $i{=}M^*N$, the initial front pixel value $P_2\,[i-1]$ or the last pixel value $P_2\,[M\times N]$ is defined as an average value.

IV. DESIGN OF ENCRYPTION PROCESS

A. ENCRYPTION ALGORITHM FLOW CHART

The algorithm firstly splices the RGB three channels of the original color image, and uses the chaotic sequence mapping to replace the pixel coordinates to achieve the purpose of scrambling the pixels, and then substitutes pixel values by matrix convolution, Finally, the XOR diffusion among the pixels is performed, and the encrypted image is obtained after the three channels are split and integrated. The encryption flow chart is shown in Fig.6, and the image encryption algorithm is shown in Algorithm 1.

Description of encryption algorithm:

B. ENCRYPTION ALGORITHM STEPS

Step1 Pretreatment: Color image conversion: The original color image $P_{m\times n\times 3}$ is a three-dimensional image, and the color image is decomposed into three-channel images of red, green and blue, and spliced into a two-dimensional gray rectangle image P_0.

Step2 Scrambling: The two-dimensional gray rectangle image P_0 is scrambled into P_1, that is, P_0 coordinates are mapped with the data coordinates converted by the chaotic sequence to generate an encrypted matrix image P_1.

The chaotic coordinate map is used to replace the coordinates of the pixel points to achieve pixel point scrambling.

Step3 Convolution Operation Replacement: The matrix image P_1 is subjected to a matrix convolution operation to generate an encrypted image P_2. Using chaotic sequences as internal data of the convolution kernel, and constantly changing after a calculation, the scrambled pixels are convoluted in units of 2×2, and then multiplied by a set of chaotic sequences to continuously generate new pixels to generate a replacement pixel matrix.

Step4 XOR Diffusion: P_2 and the cloud model Fibonacci chaotic sequence and the front (rear) adjacent pixel values are subjected to an exclusive OR operation in both directions, and an encrypted image P_3 is generated after the operation. First, extending the random sequence $F_4(i)$, and then the XOR operation of (16) is performed on the matrix P_2 and the previous adjacent pixel values, respectively, to diffuse the mutual influence among the pixel points. It takes two XOR operations (forward and backward) to get P_3.

Step5: Finally, the image is converted into a color ciphertext image in RGB mode.

The decryption process, in contrast to encryption, generates a cloud model Fibonacci random sequence based on the average of the pixels for decryption. The decryption steps are as follows:

Step1: Convert the encrypted image into RGB three-channel images and stitch them into a matrix by row.

Step2: Set the average of image pixels to the initial value to generate the cloud model Fibonacci chaotic random sequence.

Step3: XOR the encrypted matrix P_3 according to (15-16).

Step4: The matrix after the XOR operation is then deconvolution of (17). The formula is as follows:

$$\begin{aligned} &P_1[i,j]\\ &=P_2[i,j]+F_3(i)-P_2[i+1,j+1]\times filter[1,1]\\ &\quad -P_2[i+1,j]\times filter[1,0]-P_2[i,j+1]\times filter[0,1] \end{aligned} \tag{18}$$

Step5: Perform scrambling operation to obtain grayscale images.

The above image encryption algorithm combines the generalized third-order Fibonacci with the cloud model, which enhances the initial value sensitivity of the entire encryption system and increases the key space. The pixel average of the image is taken as the initial value of the chaotic sequence, and the average value is also closely combined with the XOR operation to improve the sensitivity of the plaintext. In the process of convolution operation, a small error will affect all the calculated values, and will gradually amplify the error, which realizes a good encryption effect. The above image encryption algorithm combines the generalized third-order Fibonacci with the cloud model, enhancing the initial value sensitivity of the entire encryption system and increasing the key space. The pixel average of the image is taken as the initial value of the chaotic sequence, and the average value

Algorithm 1 The Image Encryption Pseudo Code Algorithm

Input: Quantum loglistic variables $sum, x, y, z, r, belta$, loglistic variables x_0, x_1, x_2, h_0, cloud model variables Ex, En, He, N.

Code of mixing degree sequence

Initializes: $x_0 = 0.3$, $y_0 = 0.06$, $z_0 = 0.2$, $r = 3.99$, $Ex = 5000$, $En = 3$, $He = 0.1$;

for $i = 1$ to N do

$F_j = (_{A_iF_{i-1}} + B_iF_{i-2} + C_iF_{i-3}) \bmod M$,

$X_{n+1} = A_{FQL} = (F(Q(\gamma, \beta))) + L(x_0, \mu) \bmod 1$,

end

Scrambling code

for j **in** range(height):

for i **in** range(weight):

$x_i = F_{11}(i)\%N$; $y_i = F_{11}(i)/N$; $T_e = P_0[i, j]$; $P_0[i, j] = P_0[x_i, y_i]$; $P_0[_{x_i,y_i}] = T_e$;

end

end

Convolution code, image_new_test for scrambled images

for i in range(len(image_new_test)):

if $i == 0$:

$P_2[0, j] = sum(P_1[0 : 2, j : j + 2] \times filter(t)) \times F_3(0))$

else:

$P_2[i, j] = \text{sum}(P_1[i : i + 2, j : j + 2] \times filter(t)) - F_3(i)$,

Generating Encrypted Image P_2 by Matrix Convolution.

end

Exclusive or operation code

for i **in** range (len(image_new_test) $-1, -1, -1$):

if $i ==$ *len* (image_new_test) -1:

$P_3[0] = P_2[0] \oplus F_{44}(0) \oplus$ int (image_mean)

else:

$P_3[i] = P_2[i] \oplus F_{44}(i) \oplus P_2[i - 1]$, Encrypted Image P_3 Generated by Two XOR Operations in Positive and Reverse Directions.

end

Output: P3[i], Encrypted image.

is also closely combined with the XOR operation to improve the sensitivity of the plaintext. In the process of convolution operation, a small error will affect all the calculated values, and will gradually amplify the error, which realizes a good encryption effect.

C. ENCRYPTION PROCESS

The encryption process image selects a color image Peppers with a pixel size of 256 × 256 to respectively display the images after scrambling, replacement and diffusion in the image encryption process, and it is difficult to distinguish the encrypted images by visual analysis. The image encryption flowchart is shown in Fig. 7.

TABLE 2. Summarizes the result of χ^2 test for various cipher images.

Image	$\chi^2_{0.01}(255) = 310.4574$, $\chi^2_{0.05}(255) = 293.2478$, $\chi^2_{0.1}(255) = 284.3359$	
	Variance	χ^2
peppers (256×256)	1.958×10^5	287.0703
Lena(256×256)	2.627×10^5	269.6766
Black (512×512)	2.674×10^8	277.4233
White (512×512)	2.6739×10^8	259.9667

V. ANALYSIS OF ENCRYPTION SIMULATION TEST RESULTS

A. EXPERIMENTAL RESULT

The Python 3.6 platform is used to encrypt the color image. The parameters of the chaotic system are $x_0 = 0.3$, $y_0 = 0.06$, $z_0 = 0.2$, $r = 3.99$, $\beta = 6.2$, the cloud model takes values $Ex = 5000$, $En = 3$, $He = 0.1$, and the average pixel value of each image is taken as the key. In order to verify the encryption effect of different color images, four images of Lena, peppers, black and white were selected for encryption.

Fig.8(a)(c)(e) and (g) are original images. After the encryption process, the original image information cannot be recognized in the ciphertext image, and the useful image information is hidden, which means that the proposed encryption algorithm has a good encryption effect.

B. HISTOGRAM ANALYSIS

The RGB component histogram distribution before and after encryption is shown in Fig.9. The encrypted histogram distribution is obviously different from the plaintext histogram, and the encrypted RGB component histogram is distributed smoothly, and the distribution law before encryption cannot be identified, which shows a better encryption performance. Fig.9 shows the three channel histogram of 256 × 256 color image.

The performance of the proposed chaotic image encryption system is analyzed. First of all, histogram of encrypted images are considered. The chi-square test is used to justify the uniformity of their histograms. For M × N images, the unilateral hypothesis test χ^2 can be expressed as:

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

where $g_i = g = \frac{MN}{256}$, $i = 0, 1, \cdots, 255$, given the significance level α, and $P\left\{\chi_2 \geq \chi^2_\alpha(n-1)\right\} = \alpha$, generally, when $\alpha = 0.01, 0.05, 0.1$, $\chi^2_{0.05}(255) = 293.2478$, $\chi^2_{0.01}(255) = 310.4574$, $\chi^2_{0.1}(255) = 284.3359$.

From Table 2, it can be seen that the χ^2 of plaintext image is significantly greater than 0.01, while the χ^2 of ciphertext image is less than 0.01, for example, the x value of Lena ciphertext image is lower than 0.01, which passes the χ^2-test. And the variance of ciphertext image has a great change

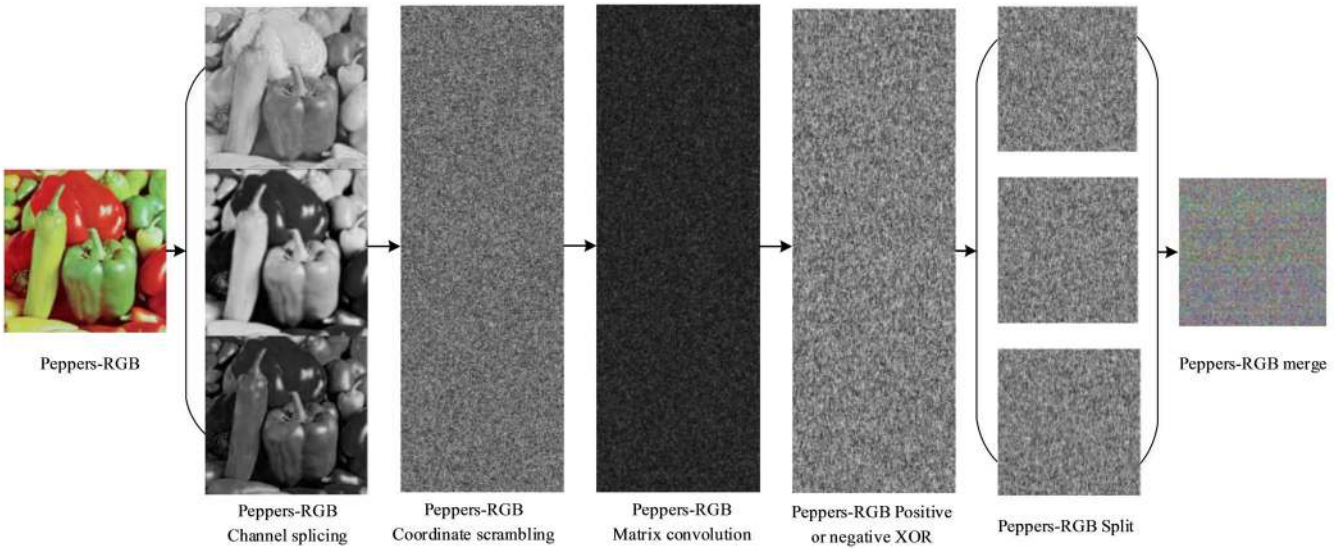


FIGURE 7. Color image encryption.

TABLE 3. NIST statistical test results.

Test name	Peppers Cipher image		Lena Cipher image	
	P-value	Result	P-value	Result
Frequency	0.38681	Success	0.34695	Success
Block Frequency (m=128)	0.21812	Success	0.23571	Success
Cumulative Sums (Forward)	0.33322	Success	0.28499	Success
Cumulative Sums (Reverse)	0.34821	Success	0.27915	Success
Runs	0.33507	Success	0.39517	Success
Longest Run of Ones	0.37925	Success	0.32759	Success
Rank	0.52788	Success	0.50828	Success
Nonperiodic Template Matchings	0.20845	Success	0.28082	Success
Universal Statistical	0.37577	Success	0.40524	Success
Overlapping Template Matchings	0.51387	Success	0.60113	Success
Approximate Entropy	0.42166	Success	0.43152	Success
Random Excursions	0.38511	Success	0.44716	Success
Random Excursions Variant	0.34315	Success	0.29822	Success
Serial	0.51239	Success	0.49214	Success
Linear Complexity	0.48513	Success	0.50894	Success
FFT	0.41736	Success	0.37197	Success

compared with that of plaintext image. It can be concluded that the image generated by encryption algorithm is random enough and has high security level. The uniformity of histogram can be evaluated by the test of χ^2 of cipher image.

At the same time, the randomness of ciphertext image is tested, NIST statistical test results for cipher images of Lena and Peppers gray-scale image are presented in Table 3. If the p-value $>$ a (in SP 800-22 test a=0.01)the sequence passed the test. A good random/pseudo-random sequence should pass all the tests.

As can be seen from the above table, the values of all 16 test items of Lena and Peppers ciphertext images are larger than the standard level, and the values of 3 items exceed 0.5, and the rest of the data are far greater than 0.01, so the secret generated by this image encryption algorithm The key stream sequence is random, and the encryption algorithm algorithm has good security.

C. KEY SENSITIVITY ANALYSIS

Key sensitivity is an important detection step in the security analysis of encryption algorithms. In this paper, the chaotic initial key deviation is 10^{-16} and then decrypted. The decrypted image cannot be restored to the original image. The original image can be restored when the deviation is 10^{-17}, which proves that the algorithm has strong sensitivity. The deviation decryption is performed on the encrypted Lena and Peppers color images as shown in Fig.9.

In order to better distinguish the ciphertext image and the error decryption image, data analysis is carried out through the standard measurement of mean square error (MSE), as shown in (20):

$$MSE = \frac{I}{M \times N} \sum_{j=1}^{M} \sum_{i=1}^{N} (a_{ij} - b_{ij})^2 \tag{20}$$

where: The parameters a_{ij} and b_{ij} represent the gray values of plaintext image and ciphertext image respectively, the larger the MSE value, the more secure the encryption is. It can be found from Table 4 that MSE data of ciphertext image and plaintext image, error decryption image and plaintext image can be distinguished from error decryption image by MSE between images.

D. CORRELATION ANALYSIS

The positional relationship of image pixels is divided into horizontally adjacent, vertically adjacent and diagonally adjacent. The strong correlation among image pixels threatens

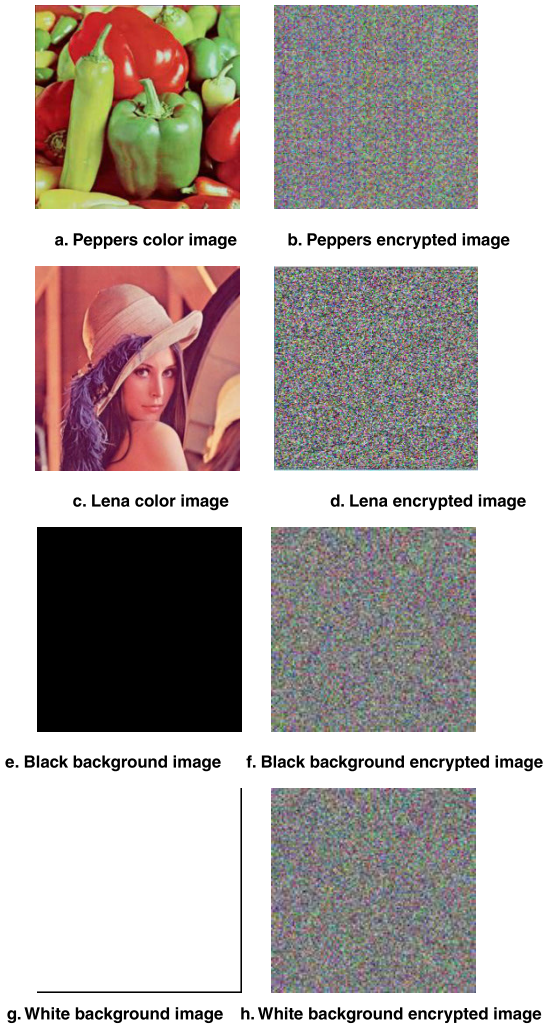


FIGURE 8. Encrypted image.

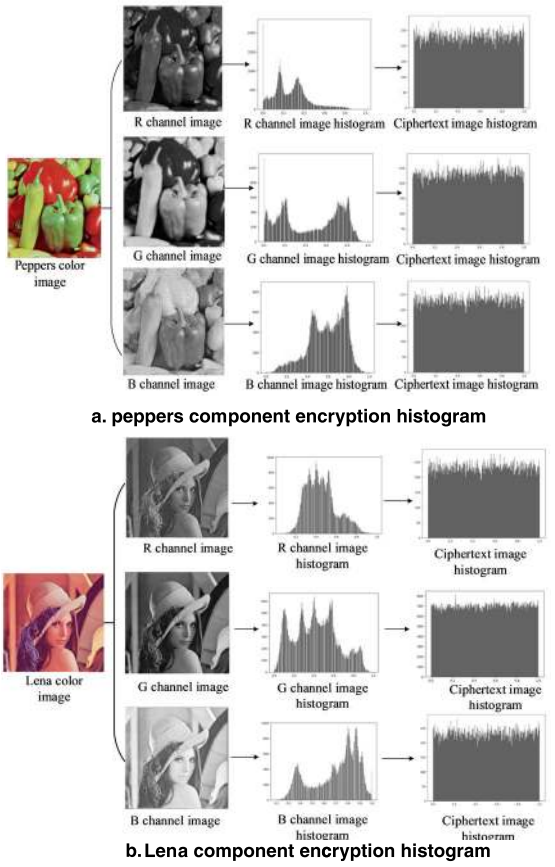


FIGURE 9. Histogram.

TABLE 4. Mean square error analysis.

Image	MSE
Cipher-image of peppers - Plaintext image	8295
Error-image of peppers -Plaintext image	8154
Cipher-image of peppers-Error-image of peppers	≈10000
Cipher-image of Lena -Plaintext image	7506
Error-image of Lena-Plaintext image	7465
Cipher-image of Lena-Error-image of Lena	≈10000

the security of the image information, and the lower the correlation, the higher the disruption degree of scrambling. Fig.11(a)-(f) show the distribution of adjacent pixels, convolution image, and ciphertext image in the horizontal direction of the Lena and pepper color plaintext image, respectively. Table 1 is a correlation analysis table of Lena image pixels. After the image encryption process, it can be seen from the data in table 1 that the image of the plaintext image component has a high correlation, and the correlation coefficients are close to 1, and the correlation coefficient of the ciphertext image approaches 0. The encryption algorithm in this paper destroys the statistical properties of the original image, and the correlation analysis formula is as follows:

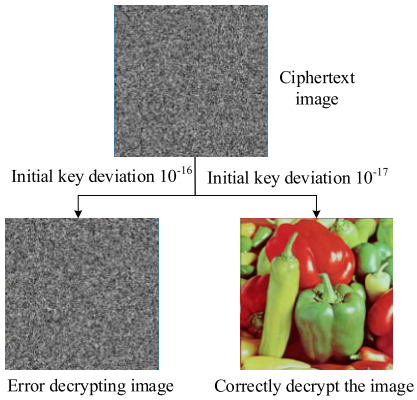
$$r(x,y)=\frac{\sum_{i=1}^{N}(x_i-\frac{1}{N}\sum_{i=1}^{N}x_i)(yi-\frac{1}{N}\sum_{i=1}^{N}yi)}{\sqrt{\sum_{i=1}^{N}(x_i-\frac{1}{N}\sum_{i=1}^{N}x_i)^2\times\sum_{i=1}^{N}(y_i-\frac{1}{N}\sum_{i=1}^{N}y_i)^2}} \tag{21}$$

where: x and y are the values of adjacent pixel points, N is the number of pixel points, and the correlation between the original image and the encrypted image is shown in Table 5.

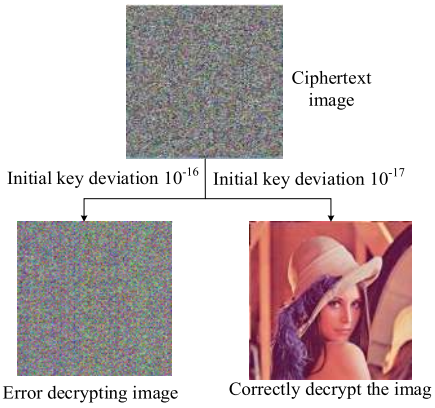
Table 5 shows the correlation coefficients of the original test image and the encrypted image, and carries out experimental analysis on the pepper and Lena images respectively. For example, the average correlation results of the calculated pepper images in the horizontal, vertical and diagonal directions are 0.9773, 0.9705, 0.9525, respectively, and the average correlation results of the corresponding encrypted image are −0.0010, 0.0016 and 0.0031. From the correlation coefficient results and figures obtained, it can be seen that the plain images are adjacent. The height relationship between pixels, which effectively reduces the pixels of the corresponding cipher image using the proposed cipher algorithm, reflects

TABLE 5. Correlation coefficient of test image.

Image	Correlation		
	Horizontal	Vertically	Diagonal
Plain-image of peppers	0.9773	0.9705	0.9525
Cipher-image of peppers	-0.0010	0.0016	0.0031
Plain-image of Lena	0.9891	0.9813	0.9047
Cipher-image of Lena	0.0012	0.0034	0.0017
Plain-image of Black background	1.0000	1.0000	1.0000
Cipher-image of Black background	-0.0038	0.0034	0.0042
Plain-image of White background	1.0000	1.0000	1.0000
Cipher-image of White background	0.0048	0.0022	0.0043



a. Peppers color image decryption



b. Lena color image decryption

FIGURE 10. Key Sensitivity Analysis.

the effectiveness of this method to hide the spatial redundancy in the pixels of the cipher image.

E. INFORMATION ENTROPY

Information entropy is used to measure the distribution of pixel values in an image. The more uniform the distribution of pixel values are, the larger the information entropy is, and the calculation formula of information entropy is as

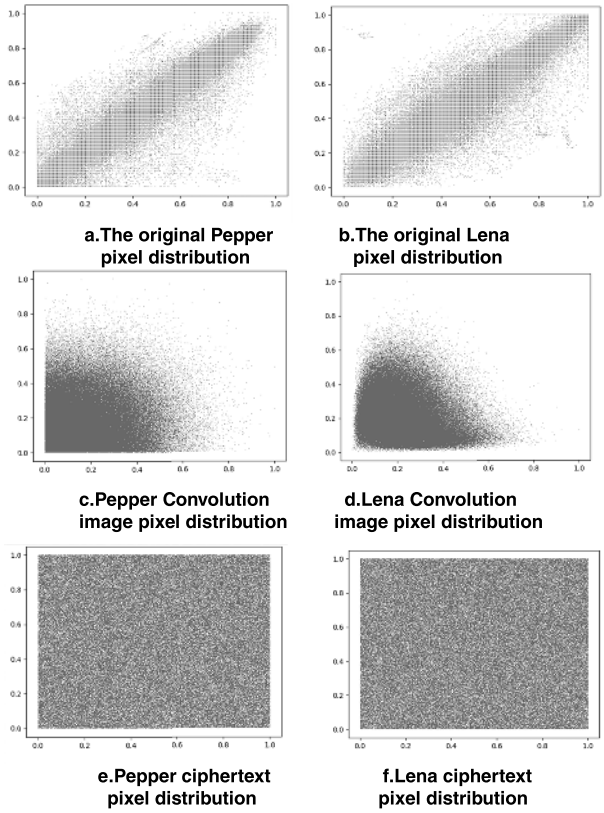


FIGURE 11. Horizontal adjacent pixel distribution.

follows:

$$H(m) = \sum_{i=0}^{255} p(m_i) \times \log_b 1/p(m_i) \tag{22}$$

where m_i represents the value of the pixel, $p(m_i)$ represents the probability of occurrence of the pixel, and Table 6 shows the entropy values of several different algorithms for 256 × 256 image. The results show that the cipher image generated by the proposed encryption method has a high entropy value, and the entropy image of the cipher is close to the ideal value of 8, which means that the probability of accidental information disclosure is very small.

F. LOCAL SHANNON ENTROPY MEASURE

In [46], a new statistical test of image randomness based on local Shannon entropy measure is proposed, which is an extension of traditional Shannon entropy. Conventionally, in the image encryption community, the usage of Shannon entropy for image randomness is to compute (23) for a sample image S. Global entropy alone can be misleading. Because the local Shannon entropy measures image randomness by computing the sample mean of Shannon entropy over a number of non-overlapping and randomly selected image blocks, It can overcome the weakness of inaccuracy, inconsistency and low efficiency of Shannon entropy.

$$\overline{\mathrm{H}_{k,T_B}}(S) = \sum_{i=1}^{k} \frac{H(S_i)}{k} \tag{23}$$

TABLE 6. Analysis results of information entropy.

Method	Information entropy	
	Lena (256 ×256)	Peppers(256 ×256)
Ref.[5]	7.9973	7.9975
Ref.[6]	7.9973	7.9984
Ref.[9]	7.9943	7.9962
Ref.[27]	7.9971	7.9973
Ref.[41]	7.9872	7.9745
Ref.[42]	7.9970	7.9971
Ref.[43]	7.9900	7.9894
Ref.[44]	7.9974	7.9973
Ref.[45]	7.9969	7.9969
The method	7.9941	7.9940

TABLE 7. Local Shannon Entropy tests for ciphertext images.

Image	Plain	Encrypted	Theoretical block entropy(60,1936)	
			α=0.01	α=0.001
			$h^{l*}_{left} = 7.90209$	$h^{l*}_{left} = 7.90209$
			$h^{l*}_{left} = 7.90284$	$h^{l*}_{left} = 7.90284$
Lena	6.89881	7.90232	Pass	Pass
Pepper	6.39232	7.90221	Pass	Pass
White	0.0000	7.90254	Pass	Pass
Black	0.0000	7.90210	Pass	Pass

Consequently, the (k, T_B)-local Shannon entropy $\overline{H}(S)$ are used as the measure for describing the randomness over the entire test image S.

Table 7 shows the local entropy measurement data selected from the fixed (k, T_B) parameter set, $k = 60$, $T_B = 1936$, and the local Shannon entropy test was performed on four test images. The real randomness of the image can be tested more accurately by using local entropy, and the image with poor encryption can be projected by local entropy. In the block entropy test, 60 non overlapping blocks of 44×44 size are randomly selected from each ciphertext image, and the average entropy is calculated by (23). It can be seen from the table that the entropy value of the test image is with in the range of standard test lines a = 0.01 and a = 0.001. The entropy shows the effectiveness and robustness of the algorithm. Therefore, the image encrypted by our method has good local randomness. Therefore, the image encrypted by our method has good local randomness.

G. DIFFERENTIAL ATTACK

Differential attack is a serious threat to the security of image information transmission. The number of changing pixel rate (NPCR) and the unified averaged changed intensity (UACI) are two most common quantities used to evaluate the strength of image encryption algorithms/ciphers with respect to differential attacks. Conventionally, a high NPCR/UACI score is usually interpreted as a high resistance to differential attacks.

The rate of change of NPCR and UACI is a measure of anti-differential attack, which is used to illustrate the anti-differential attack performance of the encryption algorithm. The calculation formula is as follows:

$$N_{NPCR} = \frac{\sum_{i,j} D(i,j)}{M \times N} \times 100\% \tag{24}$$

$$U_{UACI} = \frac{1}{M \times N} \times \sum_{i,j} \frac{|HD(i,j) - HD1(i,j)|}{255} \times 100\% \tag{25}$$

where M and N represent the length and width of the image, and the ideal expected values for the rate of change of the pixel and the averaged intensity of the pixel are 99.609% and 33.464%. Wu et al. Pointed out in [47] that the ideal values of NPCR and UACI are related to the size of pictures, and the ideal values of pictures with different sizes are also different. In addition, the statistical hypothesis model of NPCR and UACI was established in the analysis, and the NPCR hypothesis model at the significant level is shown in (26).

$$\begin{cases} H_0 : NPCR = \mu_N \\ H_1 : NPCR < \mu_N \end{cases} \tag{26}$$

In (26), μ_N is the ideal value of NPCR, $\mu_N = F/(F+1)$, where F is the maximum expected value of pixel point, in this paper, $F = 255$. When NPCR $< N^*$, reject the hypothesis of H_0, otherwise, accept the hypothesis of H_0. N^* is the threshold value of NPCR, and its definition is shown in (27).

$$N^* = \left(F - \Phi^{-1}(\alpha) \sqrt{\frac{F}{MN}} \right) \Big/ (F+1) \tag{27}$$

where $\Phi^{-1}(\bullet)$ is the reciprocal of the cumulative density function of the standard normal distribution. The hypothesis test of UACI at α significant level is shown in (28).

$$\begin{cases} H_0 : UACI = \mu_u \\ H_1 : UACI \neq \mu_u \end{cases} \tag{28}$$

In the above formula, μ_u is the ideal UACI value. When $UACI \notin (u^{*-}_\alpha, u^{*+}_\alpha)$, the assumption of H_0 is rejected. The definitions of thresholds u^{*-}_α and u^{*-}_α are shown in (29).

$$\begin{cases} u^{*-}_\alpha = \dfrac{F+2}{3F+3} - \Phi^{-1}(\alpha/2) \sqrt{\dfrac{(F+2)(F^2+2F+3)}{18(F+1)^2 MNF}} \\ u^{*+}_\alpha = \dfrac{F+2}{3F+3} + \Phi^{-1}(\alpha/2) \sqrt{\dfrac{(F+2)(F^2+2F+3)}{18(F+1)^2 MNF}} \end{cases} \tag{29}$$

In the experiment, randomly select a pixel point for Lena and pepper color images of different sizes, modify the pixel value of the image, generate a new plaintext, and finally test the encrypted image with NPCR and UACI. The test results are shown in Table 8 and Table 9. In this paper, images of different sizes are selected as test cases, including three

TABLE 8. Numerical results for NPCR randomness test.

Tested Image Size M-by-N		Theoretically NPCR Critical Value		
		256-by-256	$N^*_{0.05}=99.5693\%$,	$N^*_{0.01}=99.5527\%$, $N^*_{0.001}=99.5341\%$
		512-by-512	$N^*_{0.05}=99.5893\%$,	$N^*_{0.01}$ 99.5810%, $N^*_{0.001}=99.5717\%$
		1024-by-1024	$N^*_{0.05}=99.5994\%$,	$N^*_{0.01}=99.5952\%$, $N^*_{0.001}=99.5906\%$
Image	Reported Value（s）	NPCR Test Results		
		0.05-level	0.01-level	0.001-level
Lena 256-by-256	0.996236	Pass	Pass	Pass
Peppers 256-by-256	0.996164	Pass	Pass	Pass
Lena 512-by-512	0.996073	Pass	Pass	Pass
Peppers 512-by-512	0.996110	Pass	Pass	Pass
Lena 1024-by-1024	0.996019	Pass	Pass	Pass
Peppers 1024-by-1024	0.996155	Pass	Pass	Pass

TABLE 9. Numerical results for UACI randomness test.

Tested Image Size M-by-N		Theoretically UACI Critical Value		
		256-by-256	$C^{*-}_{0.05}=33.2824\%$,	$C^{*-}_{0.05}=33.2255\%$, $C^{*-}_{0.05}=33.1594\%$
			$C^{*+}_{0.05}=33.6447\%$,	$C^{*+}_{0.05}=33.7016\%$, $C^{*+}_{0.05}=33.7677\%$
		512-by-512	$C^{*-}_{0.05}=33.2824\%$,	$C^{*-}_{0.05}=33.2255\%$, $C^{*-}_{0.05}=33.1594\%$
			$C^{*+}_{0.05}=33.6447\%$,	$C^{*+}_{0.05}=33.7016\%$, $C^{*+}_{0.05}=33.7677\%$
		1024-by-1024	$C^{*-}_{0.05}=33.2824\%$,	$C^{*-}_{0.05}=33.2255\%$, $C^{*-}_{0.05}=33.1594\%$
			$C^{*+}_{0.05}=33.6447\%$,	$C^{*+}_{0.05}=33.7016\%$, $C^{*+}_{0.05}=33.7677\%$
Image	Reported Value（s）	UACI Test Results		
		0.05-level	0.01-level	0.001-level
Lena 256-by-256	0.333619	Pass	Pass	Pass
Peppers 256-by-256	0.333688	Pass	Pass	Pass
Lena 512-by-512	0.333813	Pass	Pass	Pass
Peppers 512-by-512	0.333804	Pass	Pass	Pass
Lena 1024-by-1024	0.333801	Pass	Pass	Pass
Peppers 1024-by-1024	0.333824	Pass	Pass	Pass

images with the size of 256 × 256, three images with the size of 512 × 512 and three images with the size of 1024 × 1024. From the observation of Table 8 and Table 9, we can see that the test results in this paper meet the test standards. And the test pass rate of the encryption algorithm is close to 1, which shows that the encryption algorithm can effectively resist differential attack.

H. NOISE ATTACK

In reality, the transmission of information is susceptible to various interferences and attacks, so image encryption algorithms are required to have strong robustness. In order to test the anti-noise attack of the encryption algorithm in this chapter, as shown in Fig.12, Gaussian noise of different intensity added to the encrypted ciphertext image. After adding 0.2 intensity Gaussian noise, the decrypted image can visually identify the main information of the image, and after increasing the noise intensity to 0.3, the image is blurred, but the basic outline of the original image can still be identified. Therefore, the encryption algorithm can resist noise attacks and has the ability to resist noise interference.

Noise is added to the encrypted ciphertext image so that the internal ciphertext pixel value portion is replaced. In the decryption process, the inverse of the diffusion is first performed, and the changed noise point is transmitted in the diffusion. In the inverse of the permutation, the value of the data is converted into new data for transmission. Without replacement pixels, the data can be restored, but the noise masks the original real data. The data shows that within a certain range of noise, the error message cannot cover the main information, and the noise attacks of size 0.2 and 0.3 cannot cover the image information.

I. CROPPING ATTACK

The experiment verifies the anti-shearing ability of the encryption algorithm by performing region clipping on the encrypted image. As shown in Fig. 13(a), the ciphertext image is cropped by 1/4 density, and the decrypted image is as

TABLE 10. The execution-time performance test.

	Proposed	Ref.[5]	Ref.[6]	Ref.[9]	Ref.[10]	Ref.[23]	Ref.[27]	Ref.[42]	Ref.[44]	Ref.[45]
PC speed	2.4	2.5	2.5	2.4	2.4	3.0	2.4	3.0	2.4	2.5
Peppers	1.13	0.654	0.46	0.091	0.022	-	0.416	2.48	1.627	2.46
Lena	3.45	0.613	1.26	0.354	0.085	0.174	1.632	7.53	4.98	8.86

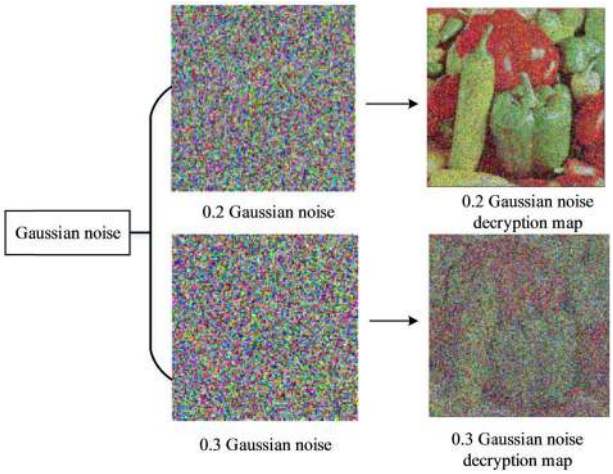


FIGURE 12. Noise attack decryption map.

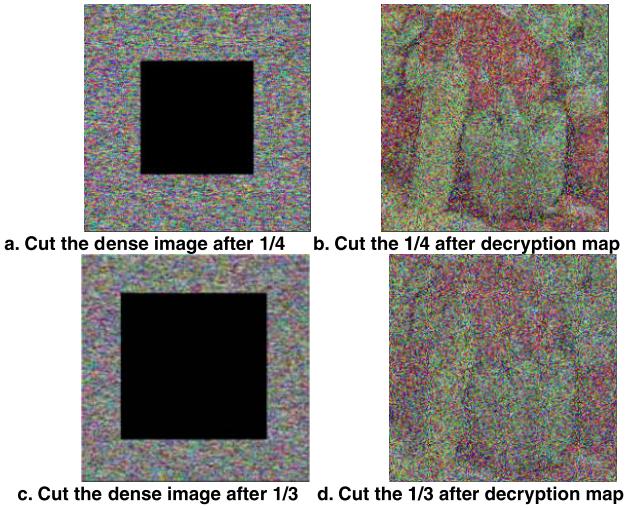


FIGURE 13. Cut attack decryption map.

shown in Fig. 13(b). The decrypted image has a lot of noise, but does not affect the overall contour of the image; after the ciphertext clipping of the 1/3 area is performed, the outline of the decrypted image is blurred, but the image information is still recognizable. Therefore, when the ciphertext image encounters clipping interference in the transmission, the proposed algorithm has better security and can effectively resist the clipping attack.

J. CHOSEN PLAINTEXT ATTACKS

When analyzing the cryptosystem, plaintext attack analysis is indispensable. Typical attacks are ciphertext only, Known plaintext, Chosen ciphertext and Chosen plaintext. Obviously, the chosen plaintext attack is the strongest attack. If the cryptosystem can resist plaintext attack, it can resist other types of attacks [48]. Selecting a plaintext attack means that the attacker uses a known encryption algorithm to derive the intermediate ciphertext through the corresponding ciphertext. The algorithm chooses to use the selected plaintext attack to test the security of the system, and adds 1 to the pixel value of the first pixel of the color image Pepper to obtain a new plaintext image, and then select the plaintext $I = \{0, 0, 0, 0\}$ whose pixel value is all 0, set the cloud model Fibonacci chaotic sequence $F_2 = \{1, 2, 3, 4\}$, chaotic sequence $F_3 = \{5\}$; The scrambling operation is invalid for the plaintext with the pixel value of 0. The scrambled ciphertext is still $Z = \{0, 0, 0, 0\}$, and then do the convolution operation to get 5 is so that the pixel value is not 0. Through the pixel demo data, the algorithm can effectively resist the choice of plaintext attacks.

K. KEY SPACE ANALYSIS

Key space analysis is an important detection method, which is a necessary test for the feasibility of using random sequence generator to generate encryption key. The key space of the encryption algorithm should be large enough to ensure the security of the detection target. The experimental data shows that the data of the algorithm is accurate to 11 decimal places, and the key space is $10^11 \times 6 \approx 2^{219}$, which is much larger than 2^{218}, which increases the ability to resist key attacks.

L. SPEED ANALYSIS AND COMPARISON

Execution-time is also an important factor, with respect to security level. Encryption and decryption durations of the proposed algorithm are analysed for images with different sizes and compared with previous studies presented in [5], [6], [9], [10], [23], [27], [42], [44], [45]. As shown in Table 10.

The proposed encryption and decryption algorithm is simulated in Python 3.6 environment in Windows Intel(R) Core 2 Duo CPU 3GHz processor with 4 GB RAM. The encryption time of 256 × 256 gray image is more than 1 second. For example, the encryption time of Pepper image (256 × 256) is 1.3sec. Compared with the encryption speed of [5], [9], [10], [23], the encryption speed of this algorithm is much slower, especially that of [10]. The main reason is that in order to improve the encryption security of this encryption method, the image is scrambled with complex chaos, and the image pixel value is added with matrix convolution, which

greatly increases the encryption speed. Compared with literature [6], [27], the data are close, and compared with literature [42], [44], [45], the time efficiency is greater than them. To sum up, in order to ensure a more secure encryption effect, the time consumption is large, and the encryption efficiency is slightly lower than that of some encryption methods.

VI. CONCLUSION

In this paper, a color image encryption algorithm combining cloud model Fibonacci dynamic chaotic system and matrix convolution algorithm is proposed. The generalized third-order Fibonacci and cloud model are combined to improve the complexity of chaotic system and generate dynamic chaotic series. Chaotic sequences are applied to the scrambling and permutation of image encryption to improve the security of ciphertext. Secondly, the encryption algorithm encrypts the color image in two dimensions, reducing the computational strength and space requirements of the algorithm. At the same time, a convolution operation is added to the algorithm to replace the pixel values, reducing the RGB correlation and improving the complexity of the plaintext and ciphertext relationship. Finally, the diffusion module adopts a positive and negative two-direction XOR operation to ensure the comprehensiveness of the diffusion. The experimental results show that the encrypted ciphertext image successfully hides the image information, and the image information distribution law cannot be recognized. It can effectively resist test attacks such as interference attacks and plaintext attacks. A variety of analyses and tests, such as statistical analysis, key-sensitivity and key-space analysis, plain-image sensitivity analysis and speed test have been conducted to demonstrate the security and the validity of the proposed algorithm. It has the characteristics of high encryption security and will have high use value in image encryption.

REFERENCES

[1] A. Belazi, A. A. A. El-Latif, and S. Belghith, "A novel image encryption scheme based on substitution-permutation network and chaos," *Signal Process.*, vol. 128, pp. 155–170, Nov. 2016.

[2] X. Kang, Z. Han, A. Yu, and P. Duan, "Double random scrambling encoding in the RPMPFrHT domain," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017.

[3] J. Lang, "Color image encryption based on color blend and chaos permutation in the reality-preserving multiple-parameter fractional Fourier transform domain," *Opt. Commun.*, vol. 338, pp. 181–192, Mar. 2015.

[4] C. Pak and L. Huang, "A new color image encryption using combination of the 1D chaotic map," *Signal Process.*, vol. 138, pp. 129–137, Sep. 2017.

[5] L. Xu, X. Gou, Z. Li, and J. Li, "A novel chaotic image encryption algorithm using block scrambling and dynamic index based diffusion," *Opt. Lasers Eng.*, vol. 91, pp. 41–52, Apr. 2017.

[6] X. Chai, Z. Gan, and M. Zhang, "A fast chaos-based image encryption scheme with a novel plain image-related swapping block permutation and block diffusion," *Multimedia Tools Appl.*, vol. 76, no. 14, pp. 15561–15585, Jul. 2017.

[7] G. Zhang and Q. Liu, "A novel image encryption method based on total shuffling scheme," *Opt. Commun.*, vol. 284, no. 12, pp. 2775–2780, Jun. 2011.

[8] A. K. Acharya, "Image encryption using a new chaos based encryption algorithm," in *Proc. Int. Conf. Commun., Comput. Secur. (ICCCS)*, 2011.

[9] F. Musanna and S. Kumar, "A novel fractional order chaos-based image encryption using Fisher Yates algorithm and 3-D cat map," *Multimedia Tools Appl.*, vol. 78, no. 11, pp. 14867–14895, Jun. 2019.

[10] E. Yavuz, "A novel chaotic image encryption algorithm based on content-sensitive dynamic function switching scheme," *Opt. Laser Technol.*, vol. 114, pp. 224–239, Jun. 2019, doi: 10.1016/j.optlastec.2019.01.043.

[11] Y. Zhang, Y. Wang, and X. Shen, "A chaos-based image encryption algorithm using alternate structure," *Sci. China Ser. F, Inf. Sci.*, vol. 50, no. 3, pp. 334–341, Jun. 2007.

[12] E. Yavuz, R. Yazıcı, M. C. Kasapba ı, and E. Yamaç, "A chaos-based image encryption algorithm with simple logical functions," *Comput. Elect. Eng.*, vol. 54, pp. 471–483, Aug. 2016.

[13] M. François, T. Grosges, D. Barchiesi, and R. Erra, "Image encryption algorithm based on a chaotic iterative process," *Appl. Math.*, vol. 3, no. 12, pp. 1910–1920, 2012.

[14] K.-W. Wong, B. S.-H. Kwok, and C.-H. Yuen, "An efficient diffusion approach for chaos-based image encryption," *Chaos, Solitons Fractals*, vol. 41, no. 5, pp. 2652–2663, Sep. 2009.

[15] X. Zhang and Z. Zhao, "Chaos-based image encryption with total shuffling and bidirectional diffusion," *Nonlinear Dyn.*, vol. 75, nos. 1–2, pp. 319–330, Jan. 2014.

[16] J.-X. Chen, Z.-L. Zhu, and H. Yu, "A fast chaos-based symmetric image cryptosystem with an improved diffusion scheme," *Optik-Int. J. Light Electron Opt.*, vol. 125, no. 11, pp. 2472–2478, Jun. 2014.

[17] 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.

[18] J.-X. Chen, Z.-L. Zhu, C. Fu, L.-B. Zhang, and Y. Zhang, "An image encryption scheme using nonlinear inter-pixel computing and swapping based permutation approach," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 23, nos. 1–3, pp. 294–310, Jun. 2015.

[19] A. Kanso and M. Ghebleh, "An efficient and robust image encryption scheme for medical applications," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 24, nos. 1–3, pp. 98–116, 2015.

[20] L. Bao and Y. Zhou, "Image encryption: Generating visually meaningful encrypted images," *Inf. Sci.*, vol. 324, pp. 197–207, Dec. 2015.

[21] N. Zhou, A. Zhang, F. Zheng, and L. Gong, "Novel image compression–encryption hybrid algorithm based on key-controlled measurement matrix in compressive sensing," *Opt. Laser Technol.*, vol. 62, pp. 152–160, Oct. 2014.

[22] N. Zhou, Y. Wang, L. Gong, X. Chen, and Y. Yang, "Novel color image encryption algorithm based on the reality preserving fractional Mellin transform," *Opt. Laser Technol.*, vol. 44, no. 7, pp. 2270–2281, Oct. 2012.

[23] X. Wang and S. Gao, "Application of matrix semi-tensor product in chaotic image encryption," *J. Franklin Inst.*, vol. 356, no. 18, pp. 11638–11667, Dec. 2019.

[24] A. G. Radwan, S. H. A. El-Haleem, and S. K. Abd-El-Hafiz, "Symmetric encryption algorithms using chaotic and non-chaotic generators: A review," *J. Adv. Res.*, vol. 7, no. 2, pp. 193–208, 2016.

[25] Y. Zhou, L. Bao, and C. L. P. Chen, "A new 1D chaotic system for image encryption," *Signal Process.*, vol. 97, pp. 172–182, Apr. 2014.

[26] H.-I. Hsiao and J. Lee, "Fingerprint image cryptography based on multiple chaotic systems," *Signal Process.*, vol. 113, pp. 169–181, Aug. 2015.

[27] D. Ravichandran, P. Praveenkumar, J. B. B. Rayappan, and R. Amirtharajan, "Chaos based crossover and mutation for securing DICOM image," *Comput. Biol. Med.*, vol. 72, pp. 170–184, May 2016.

[28] A. N. Pisarchik and M. Zanin, "Image encryption with chaotically coupled chaotic maps," *Phys. D, Nonlinear Phenom.*, vol. 237, no. 20, pp. 2638–2648, 2008.

[29] R. Enayatifar, H. J. Sadaei, A. H. Abdullah, M. Lee, and I. F. Isnin, "A novel chaotic based image encryption using a hybrid model of deoxyribonucleic acid and cellular automata," *Opt. Lasers Eng.*, vol. 71, pp. 33–41, Aug. 2015.

[30] X. Wang, Y. Wang, S. Wang, Y. Zhang, and X. Wu, "A novel pseudo-random coupled LP spatiotemporal chaos and its application in image encryption," *Chin. Phys. B*, vol. 27, no. 11, Nov. 2018, Art. no. 110502.

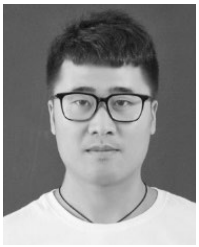
[31] S. M. Daza, F. Vega, L. Matos, C. T. Moreno, M. L. Diaz, and Y. M. Daza, "Image encryption based on convolution operation in the gyrator transform domain," in *Proc. 38th Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2012.

[32] W. Chang, B. Fang, X. Yun, S. Wang, and X. Yu, "A pseudo-random number generator based on LZSS," in *Proc. Data Compress. Conf.*, 2010.

[33] Y. Wang, Z. Liu, J. Ma, and H. He, "A pseudorandom number generator based on piecewise logistic map," *Nonlinear Dyn.*, vol. 83, no. 4, pp. 2373–2391, Mar. 2016.

[34] D. Li, C. Liu, and W. Gan, "A new cognitive model: Cloud model," *Int. J. Intell. Syst.*, vol. 24, no. 3, pp. 357–375, 2010.

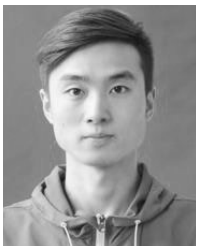
[35] B. Cao, D. Li, K. Qin, G. Chen, Y. Liu, and P. Han, "An uncertain control framework of cloud model," in *Proc. 5th Int. Conf. Rough Set Knowl. Technol. (RSKT)*, Beijing, China, Oct. 2010.
[36] Ö. Deveci, E. Karaduman, C. M. Campbell, "The Fibonacci–Circulant sequences and their applications," *Iranian J. Sci. Technol. Trans. A, Sci.*, vol. 41, no. 2, pp. 1033–1038, Dec. 2017.
[37] W. M. Abd-Elhameed and Y. H. Youssri, "Spectral tau algorithm for certain coupled system of fractional differential equations via generalized Fibonacci polynomial sequence," *Iranian J. Sci. Technol. Trans. A, Sci.*, vol. 43, no. 2, pp. 543–554, 2017.
[38] S. Shen and J. Cen, "On the bounds for the norms of R-circulant matrices with the Fibonacci and Lucas numbers," *Appl. Math. Comput.*, vol. 216, no. 10, pp. 2891–2897, Jul. 2010.
[39] S. Shen, H. U. Yan, and J. Cen, "On the spectral norms of the permutation factor circulant matrices with the k-fibonacci and k-lucas numbers," *Bull. Sci. Technol.*, vol. 82, nos. 602–605, pp. 361–364, 2011.
[40] J. K. M. S. U. Zaman and R. Ghosh, "Review on fifteen statistical tests proposed by NIST," *J. Theor. Phys. Cryptogr.*, vol. 1, pp. 18–31, Nov. 2012.
[41] M. Kar, M. K. Mandal, D. Nandi, A. Kumar, and S. Banik, "Bit-plane encrypted image cryptosystem using chaotic, quadratic, and cubic maps," *IETE Tech. Rev.*, vol. 33, no. 6, pp. 651–661, Nov. 2016.
[42] X.-Y. Wang, Y.-Q. Zhang, and X.-M. Bao, "A novel chaotic image encryption scheme using DNA sequence operations," *Opt. Lasers Eng.*, vol. 73, pp. 53–61, Oct. 2015.
[43] H. Liu and A. Kadir, "Asymmetric color image encryption scheme using 2D discrete-time map," *Signal Process.*, vol. 113, pp. 104–112, Aug. 2015.
[44] L. Xu, Z. Li, J. Li, and W. Hua, "A novel bit-level image encryption algorithm based on chaotic maps," *Opt. Lasers Eng.*, vol. 78, pp. 17–25, Mar. 2016.
[45] S. Amina and F. K. Mohamed, "An efficient and secure chaotic cipher algorithm for image content preservation," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 60, pp. 12–32, Jul. 2018.
[46] Y. Wu, Y. Zhou, G. Saveriades, S. Agaian, J. P. Noonan, and P. Natarajan, "Local Shannon entropy measure with statistical tests for image randomness," *Inf. Sci.*, vol. 222, pp. 323–342, Feb. 2013.
[47] Y. Wu, J. P. Noonan, and S. S. Againian, "NPCR and UACI randomness tests for image encryption," *J. Sel. Areas Telecommun.*, vol. 1, no. 2, pp. 31–38, 2011.
[48] X. Wang, L. Teng, and X. Qin, "A novel colour image encryption algorithm based on chaos," *Signal Process.*, vol. 92, no. 4, pp. 1101–1108, Apr. 2012.



XIANCHENG HU received the bachelor's degree in network engineering from Linyi University, in 2015. He is currently pursuing the master's degree with Qiqihar University. His current research interests include image encryption and image processing.

LIANSUO WEI received the B.S. degree in mathematics and the M.S. degree in computer engineering and application electrical engineering from Qiqihar University, Qiqihar, China, in 2003 and 2010, respectively. He is currently a Professor of computer science and technology with Qiqihar University. His current research interests include artificial intelligence and pattern recognition, sensor technology, underwater sensor networks, and information processing and simulation.

WEI CHEN received the bachelor's degree in network engineering from Qiongzhou University, in 2017. He is currently pursuing the master's degree with Qiqihar University. His current research interests include neural networks and image processing.



QIQI CHEN received the bachelor's degree in software engineering from the Wuchang Institute of Technology, in 2017. He is currently pursuing the master's degree with Qiqihar University. His current research interests include neural networks and image processing.

YUAN GUO received the B.S. degree in automation from Qiqihar University, Qiqihar, China, in 1997, and the M.S. and Ph.D. degrees in electrical engineering from Yanshan University, Qinhuangdao, China, in 2004 and 2008, respectively. She was a Visiting Scholar with Johns Hopkins University, Baltimore, MD, USA, from 2012 to 2013. She is currently a Professor of computer science and technology with Qiqihar University. Her current research interests include photoelectric detection, optical image encryption, sensor technology, and image processing.