An Image Encryption Algorithm Based on Hopfield Neural Network and Lorenz HyperChaotic System

Ye Tao, Wenhua Cui, Zhao Zhang, and Tianwei Shi

Abstract—**In this digital age, images have become one of the most important digital information. In order to ensure the security of images, various image encryption algorithms have come out one after another. Many excellent characteristics of chaotic maps can effectively enhance the stability of image encryption algorithms. Image encryption algorithms based on chaotic systems have become the focus of image encryption algorithm research. This paper proposes an image encryption algorithm combining neural network and chaotic map. A chaos matrix is generated by the Hopfield neural network model for image diffusion. The keys can be selected in the range of pixel values, and the keys space is larger. In this paper, the value of the keys are selected as three random pixel values of the image after the separation of the three primary colors of the plain. Because the pixel value of each image is different, the key is also different. Each image generates a different chaotic sequence, achieving "one key at a time". A chaos matrix is generated by Lorenz chaotic system for image scrambling. The diffusion and scrambling are carried out at the same time. Several indicators are analyzed in the experiment, and the experimental results show that the algorithm improves the key sensitivity and plain sensitivity, expands the key space, and can resist some common attacks, such as differential attack, individual diffusion attack, individual scrambling attack, etc.**

Index Terms—**image encryption, Hopfield neural network, chaotic system, scrambling, diffusion**

I. Introduction

IN the present digital world, information increasingly approaches real life, and the form of expression is becoming more and more diverse. Intuitive and simple multimedia is the key method that people use when exchanging information. In the field of information security, most information is expressed as an image. As an illustration of the similarity and viability of objective objects, images are important information vectors in our lives [1]. However, due to the improvement of decryption technology, some encryption algorithms proposed in the past have been cracked. Text encryption technology is not fully applicable to images because there is a large amount of data, high redundancy, and strong correlation between adjacent pixels [2].

The research of the image encryption algorithm is now a topic. It is important to find a more secure and effective image encryption algorithm. In the field of information security, the image encryption algorithm mainly includes image position conversion and gray level conversion. However, the above algorithms only uses the pixel level way to achieve encryption, is very easy to crack. Chaotic systems have advantages such as pseudorandom and initial value sensitivity and can effectively enhance the stability of the image encryption algorithm. Therefore, an image encryption algorithm based on chaotic systems is an important point in the study of image encryption algorithms [3].

In recent years, researchers have found that neural networks can be used to generate matrices in chaotic image encryption algorithms. This method can not only ensure the complexity of the encryption algorithm, but also fast parallel computing. Li et al. [4] focuses on two chaotic neural network modes, Aihara neural network and Inoue neural network. Because these two neural networks have the characteristics of nonlinear dynamics, which makes them extremely suitable for image encryption. At the same time, neural networks also have the advantage of nonlinear associative memory for information encryption. Wang Jin et al. [5] adopted the recursive training process. The improved Henon chaotic sequence is trained by three-layer BP neural network. Zhen et al. [6] integrated BP neural network with chaotic image encryption system.

In the research of chaotic image cipher algorithm, Liu et al. [7] propose that there is a constant problem to encrypt only the pseudorandom sequence generated from chaotic sequence. For example, the common chaotic image encryption technology mostly adopts pixel diffusion and scrambling based on one-dimensional or two-dimensional chaotic control system. The algorithm is vulnerable to differential attack, individual scrambling attack and individual diffusion attack respectively. Since Chaos Cryptography is low in security and is low in efficiency, neural network theory is applied to chaotic encryption.

Manuscript received April 11, 2022; revised Oct 12, 2022. This work was supported by National Natural Science Foundation of China (U1908218), the Project of Department of Education of Liaoning Province (2020FWDF01), and the Excellent Talent Training Project of University of Science and Technology Liaoning (2019RC05).

Ye Tao is a PhD. student in the School of Electronic and Information Engineering, and a lecturer of School of Computer and Software Engineering, University of Science and Technology Liaoning, Anshan, China. (e-mail: taibeijack@163.com).

Wenhua Cui is a Professor of School of Computer and Software Engineering, University of Science and Technology Liaoning, Anshan, China. (corresponding author to provide phone: +86-133-0422-4928; e-mail: taibeijack@126.com).

Zhao Zhang is an associate Professor of School of Computer and Software Engineering, University of Science and Technology Liaoning, Anshan, China. (e-mail: zhangzhao333@hotmail.com).

Tianwei Shi is an associate Professor of School of Computer and Software Engineering, University of Science and Technology Liaoning, Anshan, China. (e-mail: tianweiabbcc@163.com).

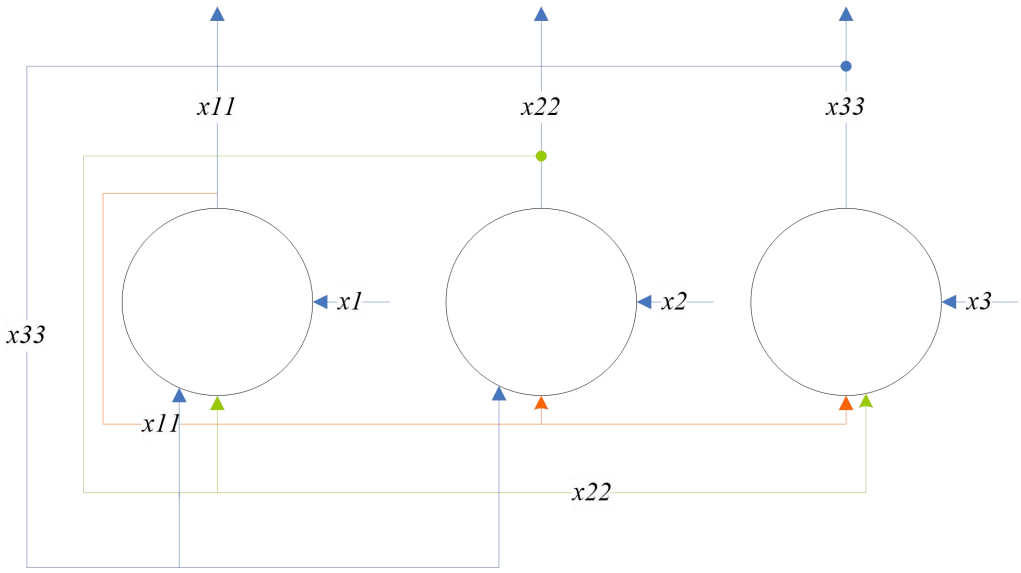


Fig.1. Hopfield neural network model

In 1982, Hopfield JJ [8] discovered a computer property that can be used in the construction of biological organisms or computers. It can also be expressed as a collective property of the system, with a large number of simple neurons. He proposed a model of the Hopfield system. This model is based on the field of neurobiology and applies to integrated circuits. Hopfield networks are characterized by simple connections between nodes. In this way, any neural network pattern of it can be called a "stable discharge pattern".

Chaotic neural networks have been found by Madody et al. [9] to be able to provide large temporal and spatial complexity, nonlinear neurons, and associative memory networks by the encryption algorithm. This method can greatly improve the security encryption algorithm. Wang et al. [10] a chaotic image encryption method based on neural network model is proposed.This method improves encryption security. In [11], the diffusion matrix is generated using a random Hopfield chaotic neural network to make the encryption more secure. However, the algorithm of [11] is independent of the usual image, and scrambling and diffusion are independent. This vulnerability gives attackers the opportunity to use plain attack, independent scrambling attack and independent diffusion attack [12]. In this paper, we propose an fan-shaped cryptographic algorithm based on Hopfield neural network and Lorenz Chaos system.

II. Related Work

A. Hopfield neural network

Hopfield neural network (HNN) is an important artificial neural network. Memory and patterns can be stored in a brain like manner. Computer simulations show that HNN can display chaotic attractors and limit loops for different parameters. HNN has serial asynchronous and parallel synchronous processing capabilities. These capabilities can provide fast solutions to some special types of computing problems. [10-12] found that there is chaos in 3D HNNs. Their experiments show that chaos can occur in some simple three-dimensional HNN. In this paper, we adopt a 3-dimensional discrete Hopfield neural network model, and the generated chaotic sequences are used for image encryption. The Hopfield neural network model is shown in Fig.1. The output value of each node is the input value of another node.

The mathematical definition of parallel three-dimensional Hopfield neural network is shown in Equation (1) .

$$x = -x_i \sum_{i=1}^{3} \omega_{ij} v_i \tag{1}$$

Where, v is the hyperbolic tangent function, as shown in Equation (2) , and ω is the weight matrix. The weight matrix used in this paper is shown in Equation (3) .

$$v_i = tanh(x_i) = \frac{e^{x_i} - e^{-x_i}}{e^{x_i} + e^{-x_i}} \tag{2}$$

$$\omega = \begin{bmatrix} 2 & -1 & 0 \\ 1.72 & 1.73 & 1.11 \\ -2.6 & -2.4 & 0.55 \end{bmatrix} \tag{3}$$

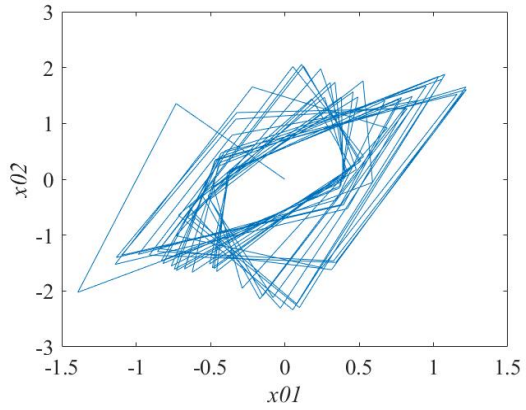
The weight matrix is placed in a neural network to obtain a final neural network model as shown in equation (4).

$$\begin{cases} \dot{x}_1 = 2f(x_1) - f(x_2) - x_1 \\ \dot{x}_2 = 1.72f(x_1) + 1.73f(x_2) + 1.11f(x_3) - x_2 \\ \dot{x}_3 = -2.6f(x_1) - 2.4f(x_2) + 0.55f(x_3) - x_3 \end{cases} \tag{4}$$

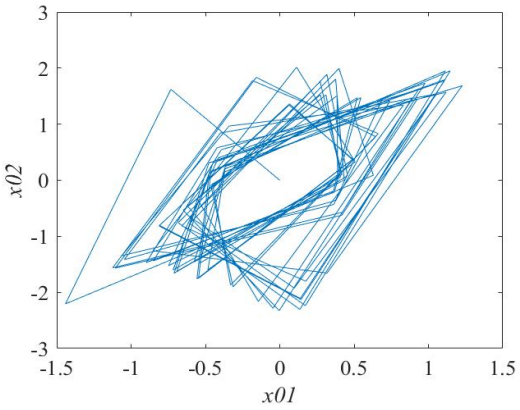
$$f(x_i) = \tanh(x_i), i = 1,2,3$$

After the network model is determined, the three primary color images are separated from the color plain, and each image selects a pixel value as the initial value to input into the neural network. After 100 iterations, the chaotic phenomenon presented by the neural network is shown in Fig. 2, and the pixel values randomly selected twice are (a), (b), (c), R=158, G = 256, B=064 and (d), (e), (f), R=160, G = 220, B = 54.

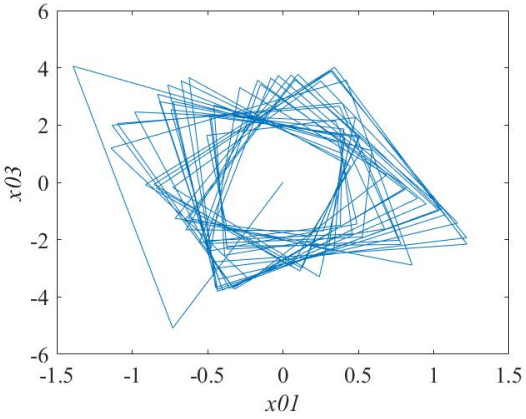
In Fig.2, the initial values of the Hopfield neural network are randomly selected twice. From these figures, it can be seen that due to the different input initial values, the obtained chaotic phenomenon graphs are also different. This situation is "one key at a time", and no matter how the initial value is chosen, the network model has chaos phenomenon.



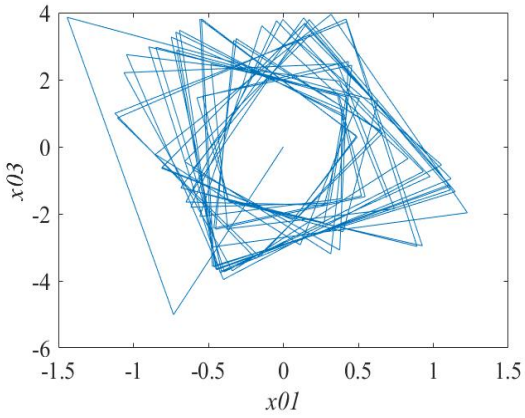
(a) Chaotic neural network x01, x02 plane



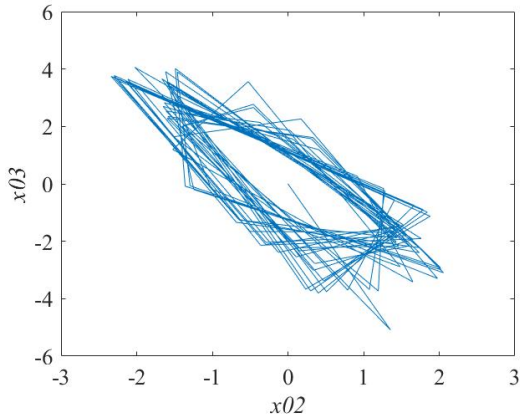
(d) Chaotic neural network x01, x02 plane



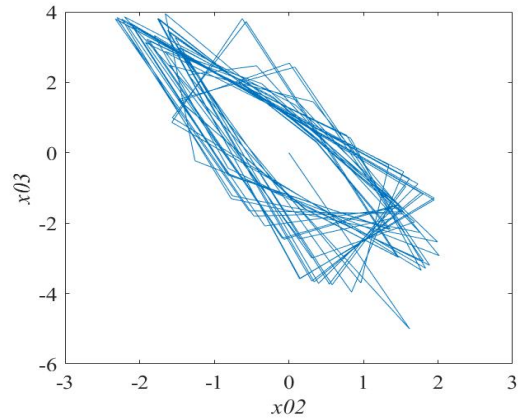
(b) Chaotic neural network x01, x03 plane



(e) Chaotic neural network x01, x03 plane



(c) Chaotic neural network x02, x03 plane



(f) Chaotic neural network x02, x03 plane

Fig. 2. Chaotic phenomenon diagram

B. Discretized hyperchaotic Lorenz systems

In the middle of the 20th century, Lorenz proposed the Lorenz equation based on the discovered chaotic attractor, also known as the Lorenz chaotic system [13]. The Lorenz chaotic system belongs to a high-dimensional nonlinear dynamic system, and its definition is shown in equation (5) . Where, $a = 10$, $b = 8/3$, $-1.52 \le r \le -0.06$, equation (5) is in hyperchaos state.

$$\begin{cases} \dot{x} = a(y - x) + w \\ \dot{y} = cx - y - xz \\ \dot{z} = xy - bz \\ w = -yz + rw \end{cases} \tag{5}$$

The discretization is made to the superchaotic Lorenz system, and the discretization uses the Runge Kutta method, which is applied in many fields. The discretized Lorenz chaotic system is shown in Equation (6).

$$
\begin{cases}
x_{i+1} = x_i + \frac{h}{6}(K_{11} + 2K_{12} + 2K_{13} + K_{14}) \\
K_{11} = a(y_i - x_i) + w_i \\
K_{12} = a\left[y_i - \left(x_i + \frac{h}{2}K_{11}\right)\right] + w_i \\
K_{13} = a\left[y_i - \left(x_i + \frac{h}{2}K_{12}\right)\right] + w_i \\
K_{14} = a\left[y_i - (x_i + hK_{13})\right] + w_i \\
y_{i+1} = y_i + \frac{h}{6}(K_{21} + 2K_{22} + 2K_{23} + K_{24}) \\
K_{21} = c\,x_{i+1} - y_i - x_{i+1}z_i \\
K_{22} = c\,x_{i+1} - \left(y_i + \frac{h}{2}K_{21}\right) - x_{i+1}z_i \\
K_{23} = c\,x_{i+1} - \left(y_i + \frac{h}{2}K_{22}\right) - x_{i+1}z_i \\
K_{24} = c\,x_{i+1} - (y_i + hK_{23}) - x_{i+1}z_i \\
z_{i+1} = z_i + \frac{h}{6}(K_{31} + 2K_{32} + 2K_{33} + K_{34}) \\
K_{31} = x_{i+1}y_{i+1} - b\,z_i \\
K_{32} = x_{i+1}y_{i+1} - b\left(z_i + \frac{h}{2}K_{31}\right) \\
K_{33} = x_{i+1}y_{i+1} - b\left(z_i + \frac{h}{2}K_{32}\right) \\
K_{34} = x_{i+1}y_{i+1} - b(z_i + hK_{33}) \\
w_{i+1} = w_i + \frac{h}{6}(K_{41} + 2K_{42} + 2K_{43} + K_{44}) \\
K_{41} = -y_{i+1}z_{i+1} - r\,w_i \\
K_{42} = -y_{i+1}z_{i+1} - r\left(w_i + \frac{h}{2}K_{41}\right) \\
K_{43} = -y_{i+1}z_{i+1} - r\left(w_i + \frac{h}{2}K_{42}\right) \\
K_{44} = -y_{i+1}z_{i+1} - r(w_i + hK_{43})
\end{cases}
\tag{6}
$$

In Equation (6), h is the step size, n is the total number of states, t is the number of filtering states, x_0, y_0, z_0 and w_0 is the initial value, and their value ranges are (-40,40),(-40,40),(1,81) and (-250,250) . When these parameters are within the value range, the hyperchaotic Lorenz system still has chaotic characteristics after discretization.

C. Image three basic color principle

In theory, any color can be made by mixing the three basic colors in various ratios. Common light is equivalent represented by mixing three basic colors of Red (R), Green (G) and Blue (B) in different ratios. The choice of three primary colors is not unique, can consider another three color light as three basic colors. However, the three colors must be independent of each other, and no one color can be composed of two other colors at the same time. Because the human eye is most sensitive to the three colors of light R, G, and B, the color area produced by the fusion of the three colors of R, G, and B is also the largest, and people usually choose it as the three basic colors. The three primary colors are shown in Fig.3.

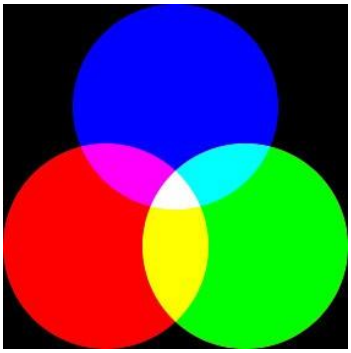


Fig. 3. Three basic colors

In this paper, the R, G, and B colors of a color image are separated to generate three grayscale images, and then each grayscale image is encrypted separately, and finally the three cipher are fused.

III. Image Encryption and Decryption Algorithm

A. Image encryption algorithm

When the Hopfield neural network has fixed weights, the initial value can arbitrarily take the pixel value in the plain. The output sequence will show chaotic characteristics, so as to achieve the effect of "one key at a time". In this paper, the chaotic matrix generated by neural network is used for forward and backward spreading of image encryption, and the diffusion adopts the algorithm of plain association. The chaos matrix generated by the hyperchaotic Lorenz system is used for scrambling, and the scrambling adopts a single-line non-repetition algorithm. Diffusion and scrambling are performed simultaneously. The specific encryption process is shown in Fig. 4.

The algorithm separates three images according to the three primary colors of the 256 * 256 plain. Three pixels are randomly selected from the three separated images. Input three pixels to a 3D Hopfield neural network. The formula of the neural network iteratively generates three 256 * 256 chaotic sequences h1, h2, h3. Use these three chaotic sequences to diffuse the three pictures R, G, and B respectively. At the same time, the chaos matrix generated by the hyperchaotic Lorenz system is used to perform single-line non-repetitive scrambling.

The first round of image encryption is shown in Fig. 5.

Step 1: Generate chaotic matrix $h1$ through Equation (4) for diffusion. Chaotic matrix L is generated by Equation (6) for scrambling. $x1 = 3$, $x2 = 6$, $x3 = 7$.

Step 2: The R image (1, 1) pixels are diffused by equation (7).

$$r(1,1) = (R(1,1) + h1(1,1) + r1 + r2) \bmod 256 \tag{7}$$

where, $r1$=91, $r2$=105, r is the encrypted matrix.

Step 3: The pixels in the first row of the R image are diffused by equation (8), and then the positions of pixels $r(1, j)$ and $r(1, N-j+1)$ in this row are exchanged. The pixel value in the chaos matrix L is the serial number of the

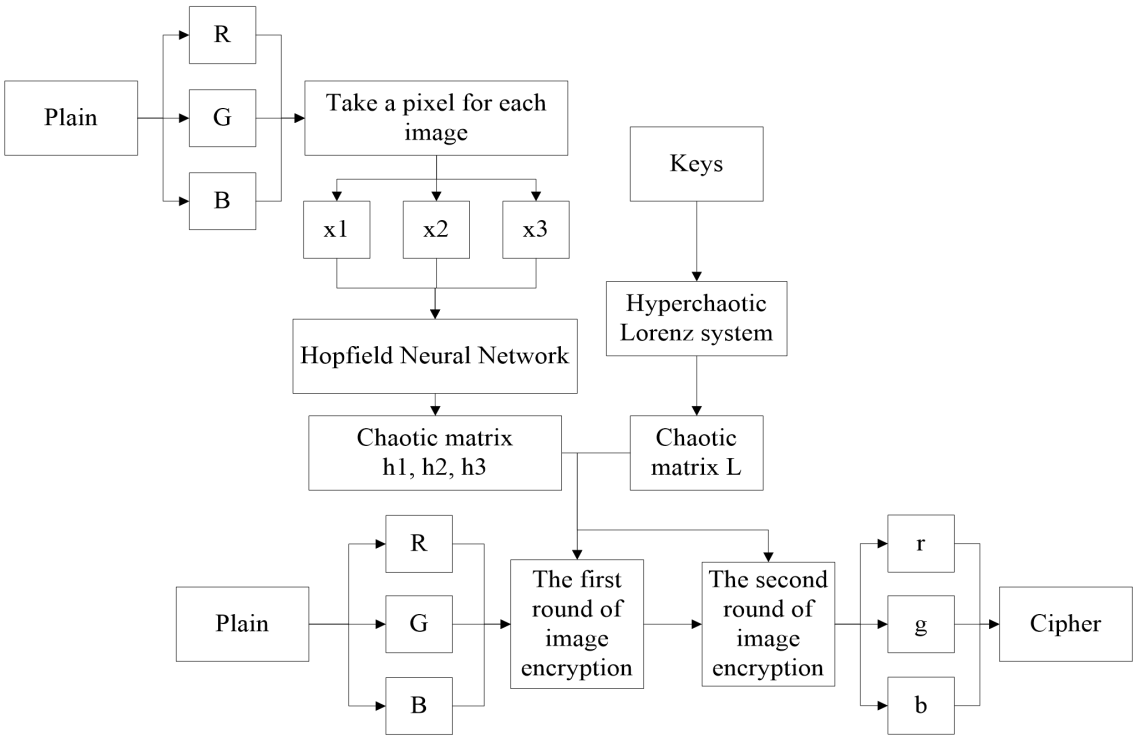


Fig. 4. Image encryption algorithm flow chart

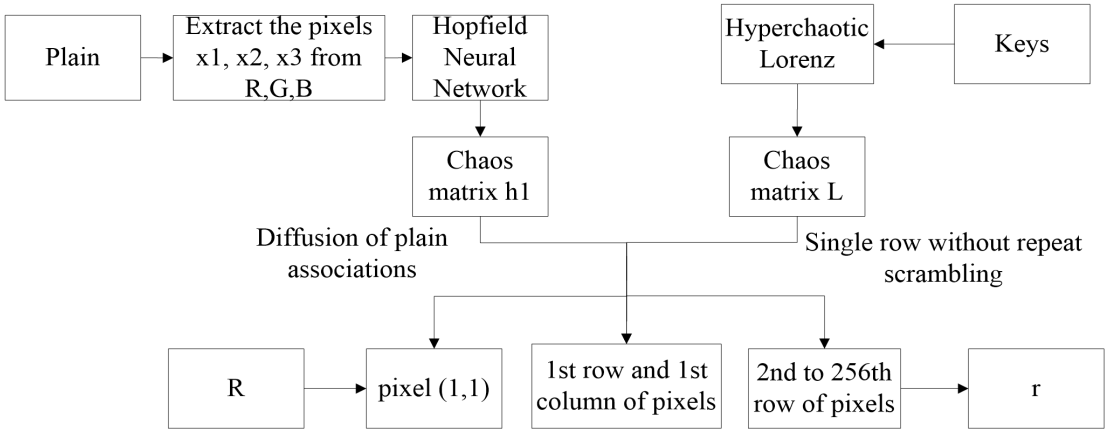


Fig. 5. The first round of image encryption flow chart

pixel in the R image, and each pixel is only scrambled once.

$$r(1,j)=(R(1,j)+r(1,j-1)+h1(1,j))\ \ mod\ \ 256 \tag{8}$$

Step 4: The pixels in the first column of the *R* image are diffused by equation (9), and then the positions of pixels $r(i,1)$ and $r(M-i+1,1)$ in this column are exchanged. The pixel value in the chaos matrix *L* is the serial number of the pixel in the R image, and each pixel is only scrambled once.

$$r(i,1)=(R(i,1)+r(i-1,1)+h1(i,1))\ mod\ 256 \tag{9}$$

Step 5: From the second row to the last row of the *R* image, the pixels of each row are diffused from left to right using Equation (10), then the positions of pixels $r(i,j)$ and $r(i,N-j+1)$ in this row are exchanged., and each pixel is scrambled only once.

Each pixel is the sum of its left pixel, upper pixel, upper left pixel, pixel at the same position in the plain, and pixel at the same position in the chaotic matrix. This ensures that each pixel value affects its right, lower, and lower right pixel values.

$$r(i,j)=(R(i,j)+r(i-1,j)+r(i,j-1)+r(i-1,j-1)+h1(i,j))\ mod\ 256 \tag{10}$$

The second round image encryption algorithm is the reverse process to the first round algorithm. First, diffuse the cipher pixel *r*(256, 256). Then, the pixels in the 256th row and the 256th column are diffused by the plain association, and the single row and single column are scrambled without repetition. Finally, the other pixels in each row are diffused and scrambled from right to left to generate a cipher.

In the same way, the encryption process of the above algorithm is performed on the *G* and *B* images of the plain, and the ciphers *g* and *b* are generated. Integrate the *r*, *g*, *b* ciphers to generate the final cipher.

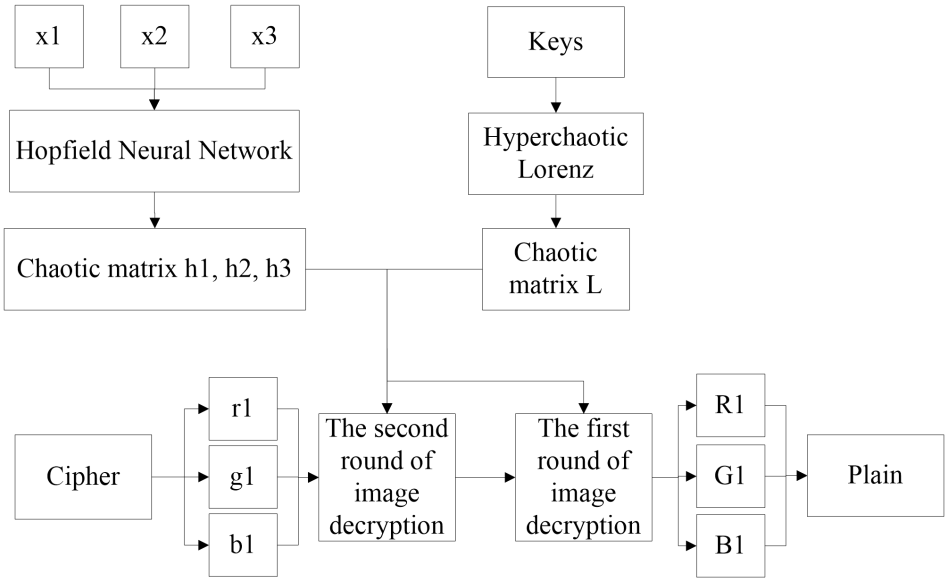


Fig. 6 Image decryption algorithm flow char

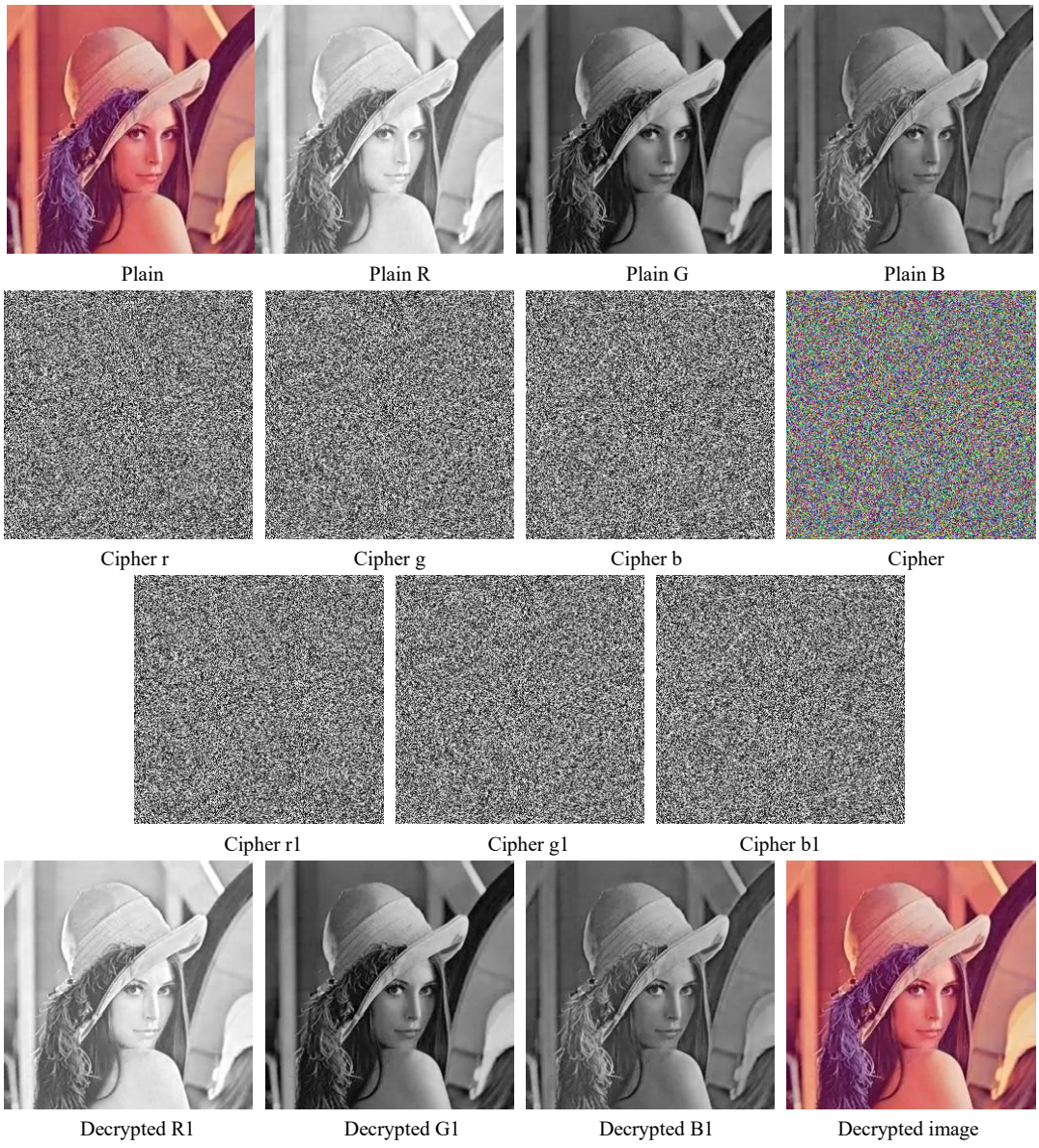


Fig. 7. Experimental results

B. Image decryption algorithm

The image decoding algorithm is an inverse calculation of the above two round cipher algorithm. First, the cipher C is separated into three primary color images r1, g1, and b1. Then decrypt the three images into R1, G1, B1 respectively and integrate them back into the original plain. Experiments show that the algorithm is correct, and the plain can be restored, and the specific decryption process is shown in Fig. 6.

IV. Experimental Results and Safety Analysis

A. The experimental results

The algorithm proposed in this paper is simulated in MatLab R2018b. The operating system is Windows10, the CPU is Intel Core i5-1135G7, the memory is 16GB, and the test image is a 256*256 color image. The experimental results are shown in Fig. 7.

Fig. 7 includes the three primary color images R, G, B which are normally separated, the encrypted images r, g, b, the synthesized cipher image cipher, the separated images r1, g1, b1, and decoded images R1, G1, B1 and the combined decoded image. The experimental results are completely correct.

B. Safety analysis

(1) Encryption and decryption time

This paper carries out 100 times of encryption and decoding for 256 images of 256 * 256, calculates average cipher time and decoding time, and compares it with other references in Table 1.

TABLE 1
ENCRYPTION AND DECRYPTION TIME

Algorithm	Encryption time (unit:s)	Decryption time (unit:s)
Proposed	0.7460	0.7541
Ref.[9]	1.1950	1.1366
Ref.[10]	0.5409	0.5687
Ref.[11]	0.7824	0.7409
Ref.[12]	0.8645	0.8784

In the image encryption algorithm proposed in this paper, in order to use a different key each time, each image randomly selects pixels in the red, green, and blue images separated from the plain as the key of the Hopfield neural network. Each time the input key is different, the obtained chaotic matrix is also different, so that each picture has a different chaotic matrix for encryption and decryption. Therefore, it has a certain impact on the encryption and decryption time, but the average encryption and decryption time is less than one second, and the security of the encryption system is greatly improved, which proves that the algorithm in this paper has strong practicability.

(2) Key space

The key space is the set of all valid keys in a cryptosystem. In the image encryption algorithm in this paper, the key of the diffusion algorithm is all combinations of any three numbers between 1 and 256. The key space is $4.7*10^{21}$ .The keys $\{h,t,a,b,c,r,x_0,y_0,z_0,w_0\}$ are used in the scramble. $x_0 \in (-40,40)$, $y_0 \in (-40,40)$, $y_0 \in (1,81)$, $y_0 \in (-250,250)$. The steps of x_0, y_0, z_0 are 10^{-13} , the step of w_0 is 10^{-12} , h=0.002, t=800, a=,10 b= 8 / 3 , c=28, r=-1. The total key space is approximately 2.56 *. For a normal image encryption system, the key space should be at least resistant to violent attacks. Therefore, the point of the paper is about . This proved that the algorithm can effectively resist the brute force attack. Table 2 shows the key space comparison results of the algorithm and other references.

TABLE 2
KEYS SPACE

Algorithm	Proposed	Ref. [9]	Ref. [10]	Ref. [11]	Ref. [12]
Key space	10^{81}	10^{50}	10^{58}	10^{38}	10^{41}

(3) Cipher statistics

The image encryption algorithm divides the usual image into three basic color images and encrypts the cipher into the noise mode using the encryption algorithm. Here, the plain three basic color R, G, B images and the encrypted cipher r, g, b images are compared with histograms, their correlation characteristics are analyzed, and the statistical characteristics of the cipher images are evaluated. The histogram of plain and cipher is shown in Fig. 8.

As can be seen from Fig. 8, the histograms of the three images of the three basic colors of the plain are all ups and downs, while the images of the three primary colors of the cipher tend to be flat. Normal and cryptographic correlation numbers are calculated to prove that there is little correlation between adjacent ciphers. If the correlation coefficient is 0, there is no correlation between adjacent pixels. Take any N pairs of adjacent pixels from the image and record their gray value as (σ_i, τ_i), i = 1, 2, 3, ... The formula of correlation coefficient of vector $\sigma = \{\sigma_i\}, \tau = \{\tau_i\}$ is shown in Equation (11).

$$\begin{cases} r_{xy} = \dfrac{\mathrm{cov}(\sigma,\tau)}{\sqrt{D(\sigma)}\sqrt{D(\tau)}} \\ D(\sigma) = \dfrac{1}{N}\sum_{i=1}^{N}(\sigma_i - E(\sigma))^2 \\ E(\sigma) = \dfrac{1}{N}\sum_{i=1}^{N}\sigma_i \\ cov(\sigma,\tau) = \dfrac{1}{N}\sum_{i=1}^{N}(x_i - E(\sigma))(y_i - E(\tau)) \end{cases} \tag{11}$$

Set the coordinate of σ_i as (x_i, y_i). if the coordinate of τ_i is (x_{i+1}, y_i), then the correlation coefficient in the horizontal direction is calculated. If the coordinate of τ_i is (x_i, y_{i+1}), then the correlation coefficient in the vertical direction is calculated; If the coordinate of τ_i is (x_{i+1}, y_{i+1}), calculate the coefficient in the diagonal direction. The test results are shown in Fig. 9.

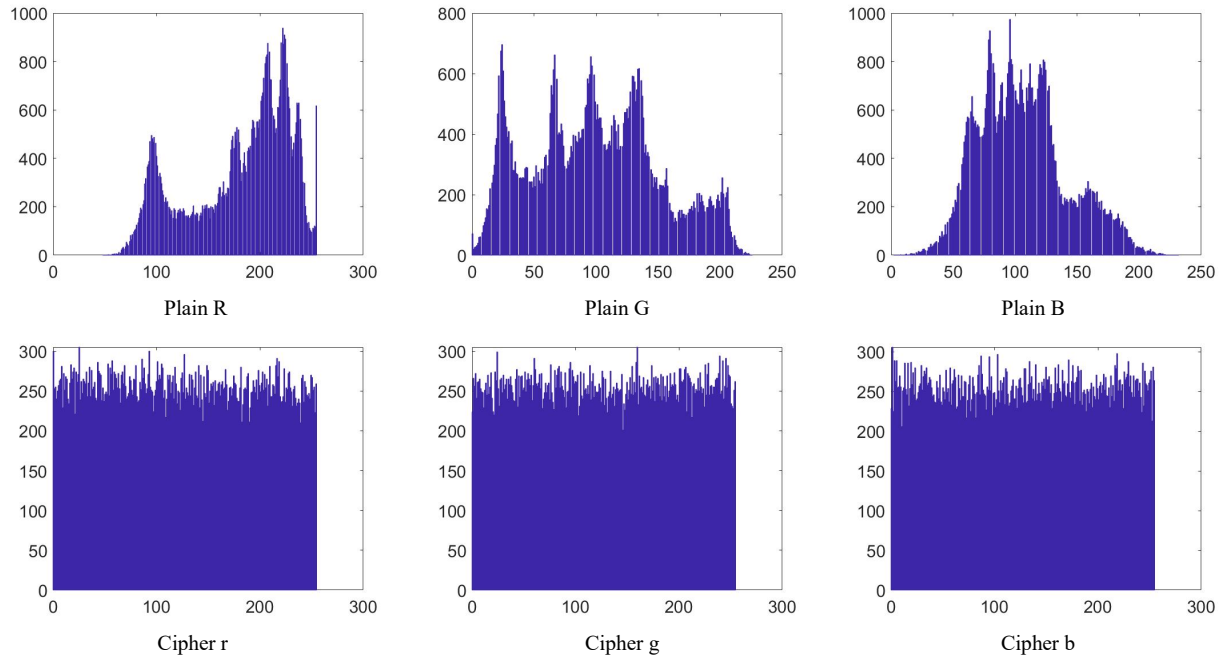


Fig. 8. The histogram of plain and cipher

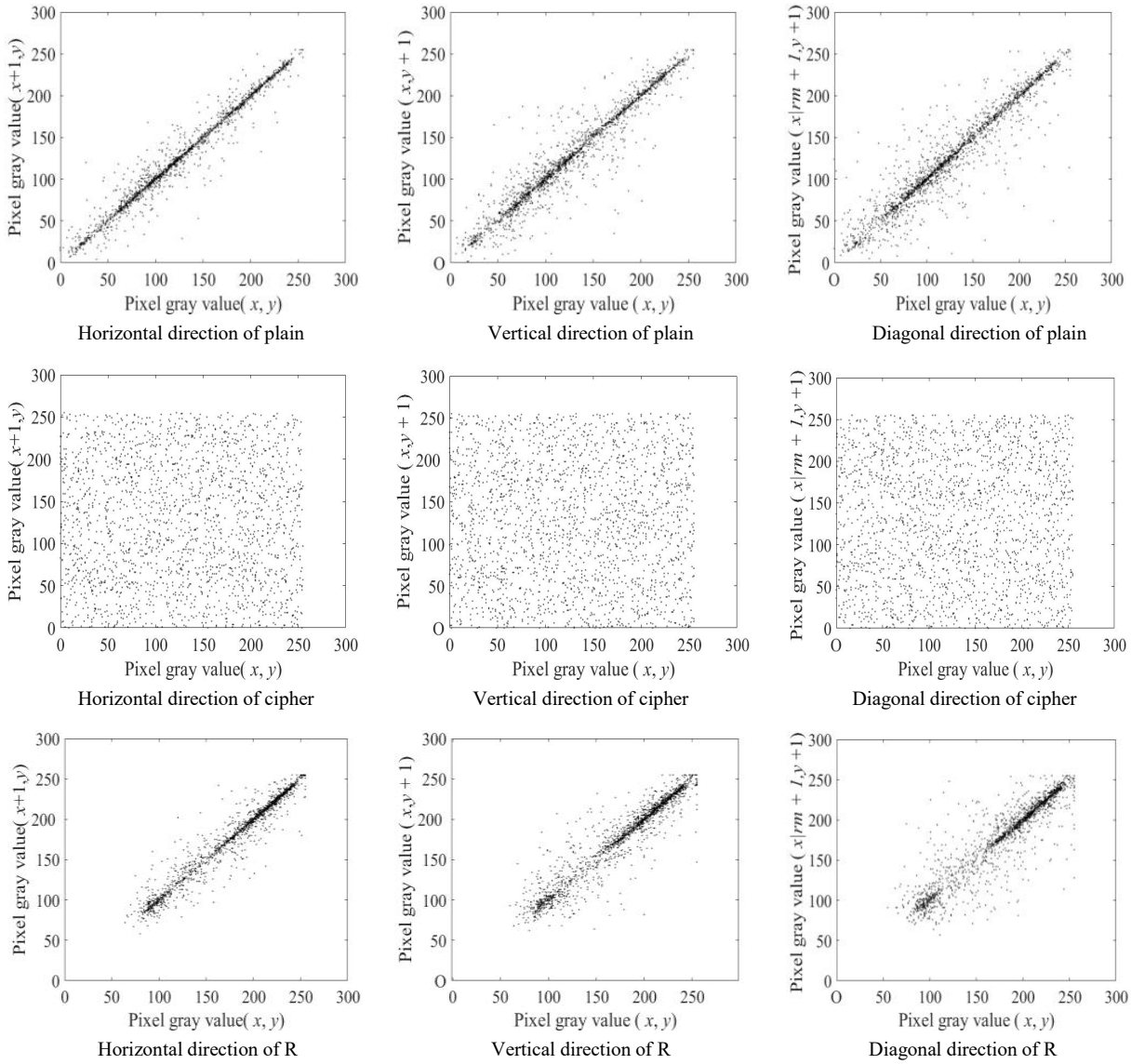


Fig. 9. Correlation distribution diagram of adjacent pixels (1)

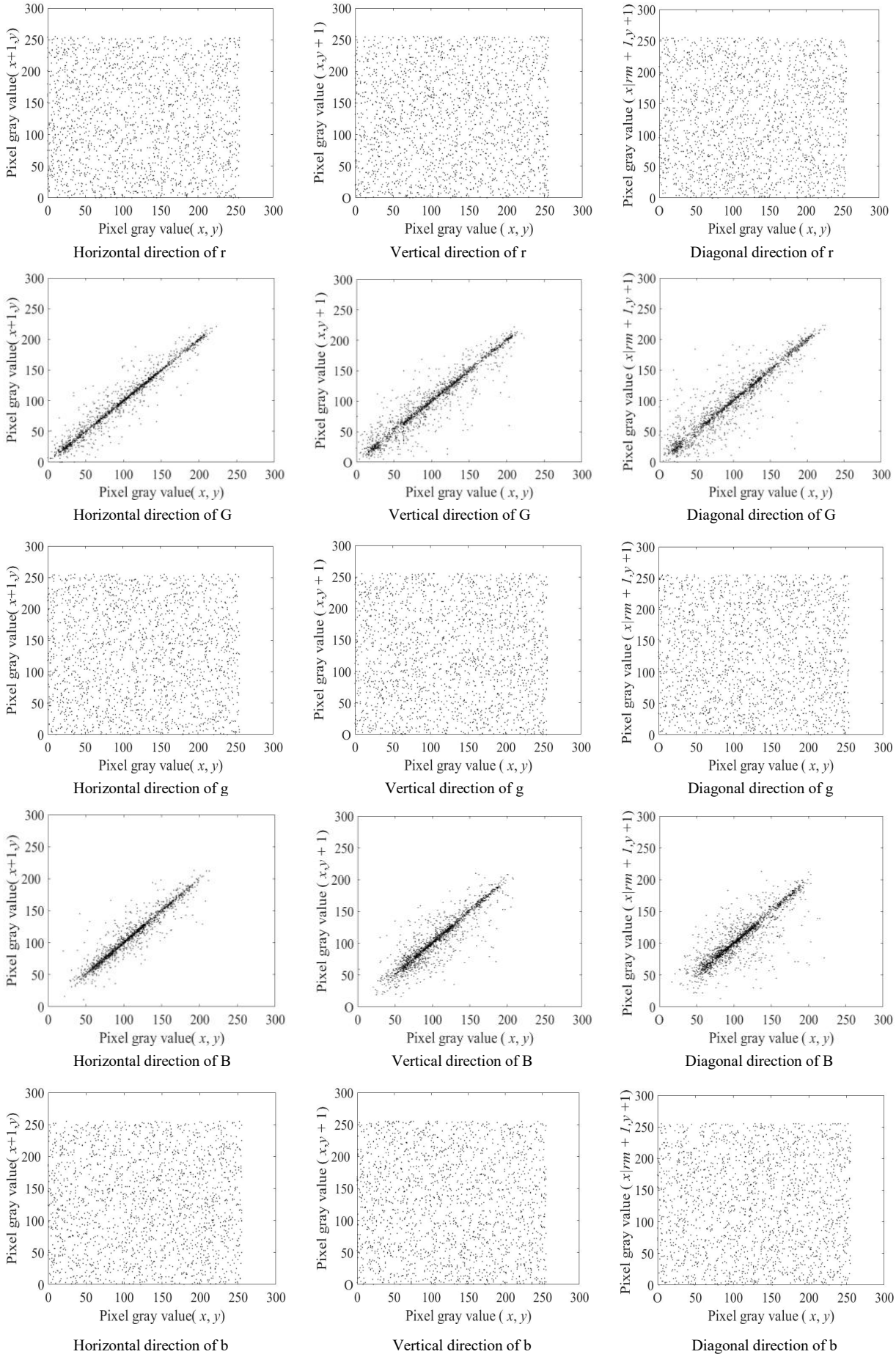


Fig. 9. Correlation distribution diagram of adjacent pixels (2)

From the Fig. 9, the pixels in the three directions of the plains are inclined to straight lines to prove strong correlation. It is proved that the distribution of pixels in the three directions of the cipher is uniform, which is weak in correlation and can be effectively resistant to statistical attacks. Table 3 shows the results of calculation of correlation coefficients between adjacent pixels in the horizontal, vertical and diagonal directions of the color image.

TABLE 3
THE CORRELATION COEFFICIENT

Algorithm	Horizontal direction	Vertical direction	Diagonal direction
Proposed（plain）	0.9770	0.9650	0.9391
Proposed（cipher）	0.0006	0.0009	0.0085
Ref.[9]（cipher）	0.0046	0.0011	0.0031
Ref.[10]（cipher）	0.0056	-0.0876	0.0085
Ref.[11]（cipher）	-0.0131	0.0142	-0.0091
Ref.[12]（cipher）	–0.0088	0.0014	−0.0015

Table 3 shows that the correlation between adjacent pixels of the plains is higher than 0.9000, and the correlation is quite strong. The correlation coefficient between adjacent pixels in the horizontal, vertical, and diagonal directions is close to zero, which has little correlation.This phenomenon reflects the safety and effectiveness of the proposed algorithm. Compared with other references, the correlation coefficient meets the requirements.

(3) Key sensitivity

The key sensitivity is a slight change in key parameters and initial values, and the sensitivity of the algorithm is analyzed by comparing the difference between encrypted images. There are two main indicators for measuring differences between two images: NPCR and UACI. In this paper, two images of the same size are represented as P1 and P2, and the image size is M*N.

The NPCR is a value which compares the corresponding positions of two images, and records the ratio of different pixels in all pixels. The theoretical expectation of NPCR is 99.6094%. The calculation method is shown in equation (12).

$$NPCR(P1,P2)=\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}|\,Sign(P1(i,j)-P2(i,j))\,|\times 100\% \quad (12)$$

The UACI compares the pixel values at the corresponding positions of the two images, records their differences, and then calculates the average value of the pixel difference and the maximum difference ratio at all corresponding positions. The theoretical value of UACI is 33.4635%. The calculation method is shown in equation (13).

$$UACI(P1,P2)=\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{|\,P1(i,j)-P2(i,j)}{255-0}|\times 100\% \quad (13)$$

The initial values of the key are changed during image encryption. The initial value of the diffusion key is changed 1000 times by 1 each time, and an initial value of the scrambled key is changed 1000 times by each time. As shown in Table 4, the keys sensitivity of the analysis algorithm is compared with the results of other references.

TABLE 4
KEYS SENSITIVITY

Algorithm	NPCR（%）	UACI(%)
Proposed	99.6061	33.4657
Ref.[9]	99.6155	29.4367
Ref.[10]	99.6249	33.4756
Ref.[11]	99.6160	33.4559
Ref.[12]	99.6037	33.4523
Theoretical value	99.6094	33.4635

Table 4 shows that the NPCR and UACI values of this algorithm are very close to theoretical values and are closer to other algorithms. It is proved that the "one key at a time" of the algorithm has better effect.

(5) Plain sensitivity

The sensitivity of plain is to slightly change the value of a pixel in plain before encryption, and compare the difference of cipher images before and after encryption. The simple sensitivity is tested by calculating the values of NPCR and UACI. In this paper, the random pixel value of color plain image is slightly changed, and the change value is 1. The experiment was conducted for 1000 times, and the average values of NPCR and UACI were calculated. The comparison data between the experimental results and other references are shown in Table 5.

TABLE 5
PLAIN SENSITIVITY

Algorithm	NPCR（%）	UACI(%)
Proposed	99.6097	33.4663
Ref.[9]	99.6101	33.4801
Ref.[10]	99.7926	33.4386
Ref.[11]	99.6198	33.4547
Ref.[12]	99.6027	33.4819
Theoretical value	99.6094	33.4635

Since the initial value of the Hopfield neural network of this algorithm is the pixel value of the plains, the pixel value after the change may be an initial value. Even if the initial value is not the initial value, the diffusion algorithm of this paper has a characteristic of correlation. Therefore, the sensitivity is usually close to the theoretical value and can be resisted by the currently selected normal attack and known plain attack.

(6) Information entropy

Information entropy reflects uncertainty of image information. The larger the entropy, the larger the uncertainty and the less visible information. The information entropy is calculated as shown in equation 14.

$$H=-\sum_{i=0}^{L}p(i)\log_2 p(i) \quad (14)$$

In Equation 14, L is the gray level of the image and $L=256$, and $p(i)$ is the probability of the occurrence of gray value i.

Table 6 shows the information entropy results of the algorithm proposed in the paper and other references.

TABLE 6
INFORMATION ENTROPY RESULTS

Algorithm	Information entropy
Proposed (plain)	7.9875
Proposed (cipher)	7.9974
Ref.[9] (cipher)	7.9976
Ref.[10] (cipher)	7.8992
Ref.[11] (cipher)	7.9991
Ref.[12] (cipher)	7.9889
Theoretical value	8

For 256 * 256 images, the data in the table shows that the entropy of information of each cipher is close to the theoretical value. The information entropy of the plain is remarkably different from the theoretical value, and it proves that this algorithm has better information entropy.

V. CONCLUSION

This paper proposes an image encryption algorithm combining Hopfileld neural network and Lorenz hyperchaotic system. Divide the plain into three primary colors. Take one pixel from each of the three primary colors as the initial value and input it into Hopfield neural network. This method can achieve the effect of "one key at a time". Three chaotic matrices generated by Hopfiled neural network iteration are respectively diffused to plaintext. In the process of diffusion, a plaintext correlation algorithm is incorporated to resist differential cryptanalysis. The discrete Lorenz hyperchaotic system generator is used to generate matrices, and then these matrices are used to scramble plaintext. The two processes of diffusion and scrambling are carried out at the same time, which can resist the single diffusion attack and the single scrambling attack. By analyzing various indexes, it is proved that the algorithm expands the key space and improves the key sensitivity and plaintext sensitivity. Encryption time, correlation coefficient, information entropy and other indicators meet the encryption requirements. These indexes prove that the algorithm has high security and practicability.

REFERENCES

[1] Y. Tao, W. Cui, J. Zhao, W. Zhang, Z. Zhang, “A Snake Encryption Algorithm for Image with Multiple Chaos Fusion,” Engineering Letters, vol. 30, no. 3, pp. 1034-1043, 2022.

[2] Y. Tao, W. Cui, Z. Zhang, “Spatiotemporal chaos in multiple dynamically coupled map lattices and its application in a novel image encryption algorithm,” Journal of Information Security and Applications. vol. 55, no. 1, pp. 2214-2126, 2020.

[3] Y. Zhang , X. Wang, “A symmetric image encryption algorithm based on mixed linea-nonlinear coupled map lattice,” Information Sciences. vol.273, pp. 329-351, 2014.

[4] Li Kun, Liu Jing. “Research on Image Encryption Based on Chaos and Neural Network,” Information and Computer (Theoretical Edition), pp. 60-61, May, 2019.

[5] Wang Jin, Qiu Rong, Wang Xiang. “Research on Image Encryption Based on Chaos and Neural Network”, Internet of Things Technology. pp. 79-81, Aug 4, 2018.

[6] Zhen Yi, Ma Boyuan, Zhang Yi, et al. “Image compression based on chaotic diagonal neural network”, Journal of Hebei Normal University (Natural Science Edition), China. pp. 387-392, May, 2015.

[7] Liu Jiasheng. “Research on image encryption technology based on chaos”, Anhui University, Hefei, China, 2007.

[8] Hopfield J J, Feinstein D I, Palmer R G. “Unlearning has a stabilizing effect in collective memories”, Nature, 1983.

[9] G. Maddodi, A. Awad, D. Awad, M. Awad, B. Lee. “A new image encryption algorithm based on heterogeneous chaotic neural network generator and DNA encoding”, Multimedia Tools & Applications, 2018.

[10] X. Wang, L. Yang, R. Liu, A. Kadir. “A chaotic image encryption algorithm based on perceptron model”, Nonlinear Dynamics, vol. 62, pp. 615-621, 2010.

[11] X. Wang, Z. Li, “A color image encryption algorithm based on Hopfield chaotic neural network,” Optics and Lasers in Engineering, vol. 115, pp. 107–118, 2019.

[12] G. Tu, X. Liao, X. Tao, “Cryptanalysis of a color image encryption algorithm based on chaos,” Optik - International Journal for Light and Electron Optics, vol. 124, pp. 5411-5415, 2013.

[13] Wang X , Wang M . “A hyperchaos generated from Lorenz system”, Physica A Statistical Mechanics & Its Applications, vol. 387, pp. 3751-3758, 2008.