2022 4th International Conference on Natural Language Processing (ICNLP) | 978-1-6654-9544-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICNLP55136.2022.00030

An Image Encryption Algorithm Based on a New Fractional Order Chaotic Neural Network

Nanming Li
Xi'an University of Posts and Telecommunications
Xi'an, China
lnmailyr@163.com

Shucui Xie
Xi'an University of Posts and Telecommunications
Xi'an, China
xieshucui@163.com

Jianzhong Zhang
Shaanxi Normal University
Xi'an, China
1416655910@qq.com

Yangguang Lou
Xi'an University of Posts and Telecommunications
Xi'an, China
louyangg07@163.com

***Abstract*—An image encryption algorithm based on a new fractional order chaotic neural network (CNN) is proposed. Firstly, a three-dimensional continuous integral order CNN is obtained by numerical simulation on the chaotic neuron model. Then, the integral order CNN is extended to fractional order, and the fractional order CNN iteratively generates pseudo-random sequences for subsequent encryption. In addition, the initial value of the fractional order CNN is generated from the hash value (secure hash algorithm: SHA-256) of the plain image. Finally, we apply the encryption structure of forward diffusion, scrambling and backward diffusion to the encryption algorithm, and diffusion is performed at the bit level, scrambling is performed at the pixel level and bit level. Experimental results and security analysis show that the algorithm has better performance and can resist typical attacks.**

Keywords—fractional order chaotic system, neural networks, secure hash algorithm, image encryption

I. INTRODUCTION

As an information carrier, image is widely used in military, education, medical and other fields. However, due to the openness of the network, the security of image information is hard to be guaranteed. Therefore, many image encryption technologies have been proposed to solve this problem. A puzzling phenomenon is that image information has the characteristics of large data capacity and strong correlation between pixels, which makes the traditional encryption algorithms, such as advanced encryption standard (AES), data encryption standard (DES), and international data encryption algorithm (IDEA), not suitable for image encryption [1].

So far, image encryption technology based on chaos has been widely studied and applied [2-5]. Chaotic system is used for image encryption because of its similar characteristics with cryptography, such as initial value sensitivity, ergodicity and aperiodicity [6]. Meanwhile, some studies show that compared with the traditional integral order chaotic system, fractional order chaotic system can produce more accurate and complex dynamic behavior [7]. Therefore, it is significant to apply fractional order chaotic system to image encryption. Nowadays, many scholars devote themselves to the studies related to chaos. For instance, Cui et al. proposed a new chaotic neuron model and a new chaotic neural network (CNN) in [8], implementing the CNN with field programmable gate array (FPGA), and solving the traveling salesman problem by using the energy function of CNN. Cui et al. [9] proposed a fractional order three-dimensional quadratic two-wing hidden chaotic system and analyzed its chaotic characteristics. In addition, spectral entropy algorithm was used to measure the complexity of the chaotic systems and experimental results showed that fractional order chaotic systems are more complex than integral order chaotic systems. Chen et al. [10] proposed an image encryption algorithm based on fractional order discrete CNN and DNA sequence operation. Fractional order discrete CNN was used to generate pseudo-random sequence and DNA coding operation was used in the encryption process. Zhao et al. [11] proposed an efficient image encryption algorithm, which consisted of a multiple block substitution stage and a bidirectional-dynamic diffusion stage.

Based on the above analyses, we present an image encryption algorithm based on a new fractional order CNN. Firstly, we give a three-dimensional continuous integral order CNN and analyze its nonlinear dynamic characteristics, including phase diagram, lyapunov exponent spectrum, bifurcation diagram and 0-1 test. Secondly, the integral order CNN is extended to the fractional order, and the dynamic behavior of the fractional order CNN is studied by using the predictor-corrector method of Adams-Bashforth-Moulton type [12] and the Wolf's method. Finally, the fractional order CNN is applied to image encryption. The specific contributions are as follows: (1) a continuous integral order CNN is obtained, its dynamic characteristics are studied. On this basis, a fractional order CNN is proposed and applied to the encryption algorithm, (2) a new bidirectional bit level diffusion algorithm is proposed to make the image information fully hidden, and (3) an encryption structure of forward diffusion, scrambling, and backward diffusion is employed, which can effectively break the correlation between adjacent pixels and achieve better encryption performance.

The rest of the paper is organized as follows. In Sec. II, the CNN is proposed and its dynamic characteristics are analyzed. In Sec. III, the proposed encryption scheme is described. Simulation results and security analyses are provided in Sec. IV. Finally, the conclusion is given in Sec. V.

978-1-6654-9544-8/22/$31.00 ©2022 IEEE
DOI 10.1109/ICNLP55136.2022.00030

II. CHAOTIC NEURAL NETWORK

A. Integral Order Chaotic Neural Network

Based on the Hopfield neural network model, a new chaotic neuron model is proposed in [8], as shown in the following equation:

$$c_i \dot{x}_i = \sum_{j=1}^{n} s_{ij} x_j + \sum_{j=1}^{n} w_{ij} v_j + i_i \tag{1}$$

where x_i is the voltage on the capacitor c_i, s_{ij} is the conductance of membrane resistance of the outside and inside neurons, i_i is the nonlinear external input current, and w_{ij} is the synaptic weight of the connection strength between neurons, v_j is the activation function of neuron.

In this paper, we let c_i =1 and n =3, so the integral order CNN model is defined as:

$$\dot{x}_i = \sum_{j=1}^{3} s_{ij} x_j + \sum_{j=1}^{3} w_{ij} v_j + i_i \,, i = 1,2,3 \tag{2}$$

where $v_j = \tanh(x_j)$. Fig. 1 shows the connections between neurons in (2). Through numerical simulation, the connection weights w_{ij}, the conductance of membrane resistance s_{ij} and input current i_i in (2) are determined as follows:

$$(s_{ij})_{3\times 3} = \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 1 \\ 0 & -3 & -5 \end{bmatrix} \tag{3}$$

$$(w_{ij})_{3\times 3} = \begin{bmatrix} 2 & 1 & -9 \\ -9 & 2 & 4 \\ 1 & -9 & 2 \end{bmatrix} \tag{4}$$

$$(i_i)_{3\times 1} = \begin{bmatrix} 0 \\ c\sin(x_1) \\ 0 \end{bmatrix} \tag{5}$$

So (2) can be defined as follows:

$$\begin{cases} \dot{x}_1 = 2x_2 + 2\tanh(x_1) + \tanh(x_2) - 9\tanh(x_3) \\ \dot{x}_2 = x_3 - 9\tanh(x_1) + 2\tanh(x_2) + 4\tanh(x_3) + c\sin(x_1) \\ \dot{x}_3 = -3x_2 - 5x_3 + \tanh(x_1) - 9\tanh(x_2) + 2\tanh(x_3) \end{cases} \tag{6}$$

In this paper, we use the fourth-order Runge-Kutta method to solve the system (6). When the initial value of the system (6) is (0.1,0.1,0.1), the parameter c is 20 and the number of iterations is 800, the phase diagram of the system (6) is shown in Fig. 2, the lyapunov exponent (LE) obtained by Euler method and QR decomposition method is $LE1$=0.6583, $LE2$= 0.0094 and $LE3$= –4.9506. Fig. 3 shows the relationship betw-een the lyapunov exponent and the parameter c. Meanwhile, Fig. 4 is the x_2 -axis bifurcation diagram of the system (6). It can be seen from the bifurcation diagram that the system (6) enters into chaos by period doubling bifurcation.

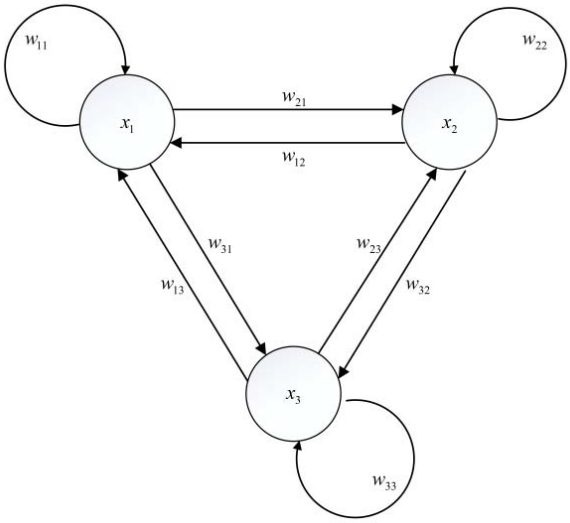


Fig. 1. The connections between neurons in (2).

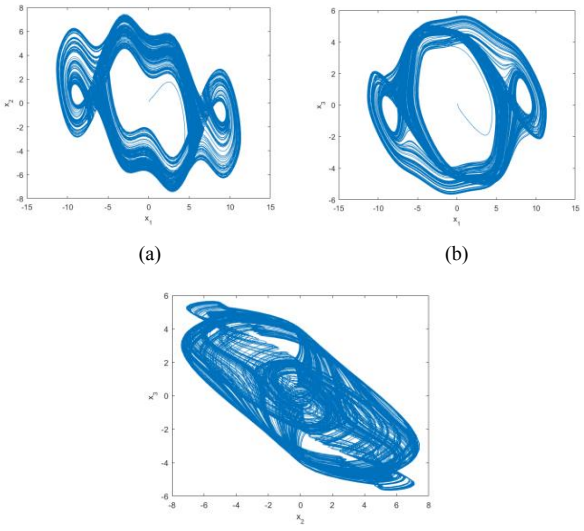


Fig. 2. The phase diagrams of system (6): (a) x_1 - x_2 plane, (b) x_1 - x_3 plane, and (c) x_2 - x_3 plane.

In addition, we use 0-1 test [13] to verify whether the system (6) is chaotic. When the parameter c of the system (6) is 20 and the initial value is (0.1,0.1,0.1), the trajectory of the (p,s) plane corresponds to Brownian motion, as shown in Fig. 5. The result of 0-1 test indicates that when parameter c is 20, the system (6) is in a chaotic state.

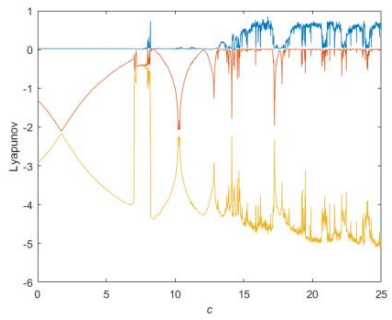


Fig. 3. Lyapunov exponent spectrum of the system (6).

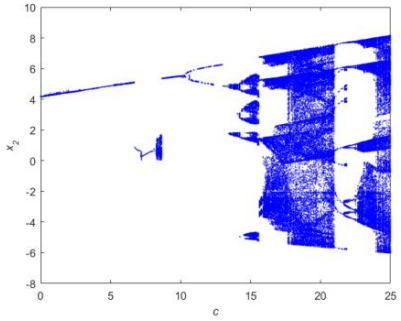


Fig. 4. The x_2 -axis bifurcation diagram of c.

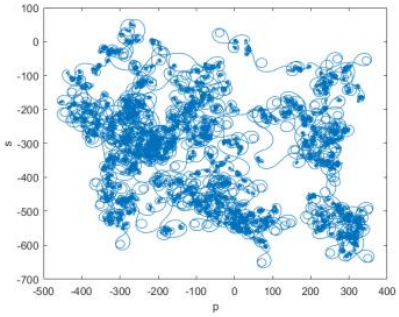


Fig. 5. (p,s) plane of x_3 sequence with $c = 20$.

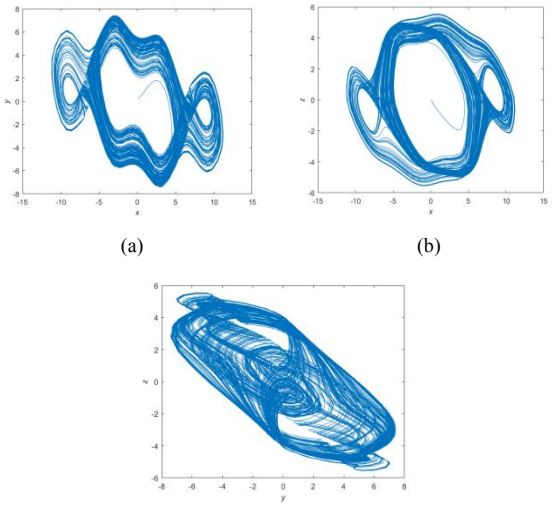
B. *Fractional Order Chaotic Neural Network*

In order to improve the randomness of sequences generated by chaotic systems, the integral order CNN is extended to fractional order, equation as follows:

$$\begin{cases} \dfrac{\mathrm{d}^{\alpha} x}{\mathrm{d}t^{\alpha}} = 2y + 2\tanh(x) + \tanh(y) - 9\tanh(z) \\ \dfrac{\mathrm{d}^{\beta} y}{\mathrm{d}t^{\beta}} = z - 9\tanh(x) + 2\tanh(y) + 4\tanh(z) + c\sin(x) \\ \dfrac{\mathrm{d}^{\gamma} z}{\mathrm{d}t^{\gamma}} = -3y - 5z + \tanh(x) - 9\tanh(y) + 2\tanh(z) \end{cases} \quad (7)$$

where α, β and γ are fractions, and different variables x, y and z can have different orders. For system (7), the theoretical analysis methods of different and the same orders are the same, so the fractional order system (7) is analyzed by $0 < \alpha = \beta = \gamma < 1$ in this paper.

We first adopt the predictor-corrector method of Adams-Bashforth-Moulton type to solve the system (7), and the phase diagrams of the system (7) are obtained as shown in Fig. 6, where the parameter $c = 20$, the initial value of the system is (0.1,0.1,0.1) and the order α, β and γ are 0.996. Then, we use the Wolf's method to calculate the lyapunov exponents of the system (7). Finally, we present the calculation results of lyapunov exponents of the system (7) varying with parameter c in Fig. 7, where the variation range of parameter c is 0 to 25, the step length $h = 0.01$, the order α, β and γ is 0.996 and the initial value of the system is (0.1,0.1,0.1). Combined with the phase diagram, we can judge that when the parameter c is 20 and the order α, β and γ are 0.996, the system (7) is in a chaotic state.



(a) (b)

(c)

Fig. 6. The phase diagrams of system (7): (a) x - y plane, (b) x - z plane, and (c) y - z plane in (2).

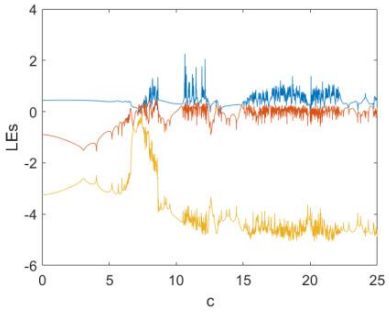


Fig. 7. Lyapunov exponent spectrum of the system (7).

The complexity of chaotic systems refers to adopting correlation algorithms to measuring the possibility that the sequences generated by chaotic system approach random sequences. The larger the complexity value is, the closer the sequence is to

random sequence. In this paper, we employ the spectral entropy algorithm [14] to study the complexity of the system (7) with the parameter $c = 20$. The research results are shown in Fig. 8, where $q = \alpha = \beta = \gamma$. The results show that the system (7) complexity values of $q = 0.996$ and $q = 1$ (the system (6)) are 0.6319 and 0.6198, respectively. It can be seen that the system (7) with $\alpha = \beta = \gamma$ =0.996 has higher complexity than the system (6). Therefore, we use system (7) for the subsequent encryption.

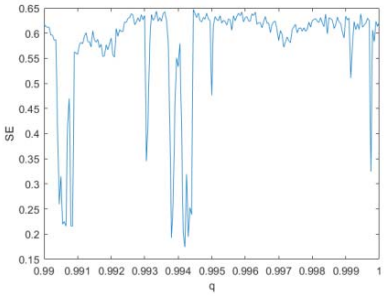


Fig. 8. Fractional order system complexity.

III. Encryption and Decryption Algorithms

The encryption algorithm in this paper is divided into four parts: generating random matrix, forward bit level diffusion, plaintext associative scrambling and backward bit level diffusion.

A. Generating Random Matrix

The three pseudo-random sequences generated by the system (7) with $\alpha = \beta = \gamma$ =0.996 and parameter $c = 20$ are spliced to obtain a new sequence. Then the random matrix is obtained by processing the new sequence. Suppose P represents the plain image to be encrypted, its size is $M \times N$, let $L = \max(M, N)$. The specific steps of generating random matrix are shown as follows:

Step 1: Perform the SHA-256 function on the plain image P to obtain a 256-bit key K, and convert K into 32 decimal numbers $k_1, k_2, \cdots, k_{32}$ with every 8 bits as a group.

Step 2: The initial values x_0, y_0 and z_0 of system (7) are calculated as follows:

$$\begin{cases} x_0 = \mathrm{mod}(k_1 \oplus k_{17} + \sum_{i=1}^{11} k_{3i-1}, 256) \Big/ 2^8 \\ y_0 = \mathrm{mod}(k_2 \oplus k_{18} + \sum_{i=1}^{10} k_{3i}, 256) \Big/ 2^8 \\ z_0 = \mathrm{mod}(k_3 \oplus k_{19} + \sum_{i=0}^{10} k_{3i+1}, 256) \Big/ 2^8 \end{cases} \tag{8}$$

where $\oplus$ represents XOR operation.

Step 3: Iterate the system (7) $\lceil M \times N \div 3 \rceil + 2000$ times, remove the first 2000 data of x, y and z to skip the transition state, where $\lceil b \rceil$ represents the nearest integer great-er than or equal to b. Then, the sequence D consists of the remaining x, y and z data (x followed by y and y follow-ed by z). Finally, the sequence Z is the first $M \times N$ data of the sequence D.

Step 4: Matrices Q, S, R, W, E and T of size $M \times N$ are generated by (9):

$$\begin{cases} Q(i,j) = \mathrm{mod}(\mathrm{floor}(\ Z((i-1)\times N + j) \times 10^{14}), 256) \\ S(i,j) = \mathrm{mod}(\mathrm{floor}(\ Z((i-1)\times N + j) \times 10^{13}), 256) \\ R(i,j) = \mathrm{mod}(\mathrm{floor}(\ Z((i-1)\times N + j) \times 10^{12}), L) \\ W(i,j) = \mathrm{mod}(\mathrm{floor}(\ Z((i-1)\times N + j) \times 10^{11}), L) \\ E(i,j) = \mathrm{mod}(\mathrm{floor}(\ Z((i-1)\times N + j) \times 10^{10}), 2) \\ T(i,j) = \mathrm{mod}(\mathrm{floor}(\ Z((i-1)\times N + j) \times 10^{9}), 2) \end{cases} \tag{9}$$

B. Forward Bit Level Diffusion

In the process of forward bit level diffusion, we first XOR the lowest bit plane of the plain image with the lowest bit plane of the random matrix, and then diffuse one by one from the lowest bit plane to the highest bit plane. The specific diffusion process is as follows: Firstly, plain image P and random matrix Q are decomposed into 8 bit planes: P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8 and Q_1, Q_2, Q_3, Q_4, Q_5, Q_6, Q_7, Q_8. Then, the 8 bit planes of the plain image are diffused through (10), and finally merge A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8 into A.

$$\begin{cases} A_1 = (P_1 \oplus Q_1) \oplus E \\ A_2 = (P_2 \oplus Q_2) \oplus A_1 \\ A_3 = (P_3 \oplus Q_3) \oplus A_2 \\ A_4 = (P_4 \oplus Q_4) \oplus A_3 \\ A_5 = (P_5 \oplus Q_5) \oplus A_4 \\ A_6 = (P_6 \oplus Q_6) \oplus A_5 \\ A_7 = (P_7 \oplus Q_7) \oplus A_6 \\ A_8 = (P_8 \oplus Q_8) \oplus A_7 \end{cases} \tag{10}$$

C. Plaintext Associative Scrambling

Firstly, we calculate the coordinate (m, n) corresponding to $A(i, j)$ by (11). Secondly, if $m \neq i$ and $n \neq j$, $A(i, j)$ will be cyclic shifted to the left by d bit, d is calculated by (12), and then exchange the positions of $A(i, j)$ and $A(m, n)$. Otherwise, $A(i, j)$ will not be shifted, and the positions of $A(i, j)$ and $A(m, n)$ remain unchanged.

$$\begin{cases} m = \mathrm{mod}((A(i,1) + \cdots + A(i, j-1) + A(i, j+1) + \cdots + A(i, N) + R(i, j)), M) \\ n = \mathrm{mod}((A(1, j) + \cdots + A(i-1, j) + A(i+1, j) + \cdots + A(M, j) + W(i, j)), N) \end{cases} \tag{11}$$

$$d = \mathrm{mod}(\ A(m, n), 8) \tag{12}$$

In Fig. 9, we give an example to facilitate understanding the scrambling operation.

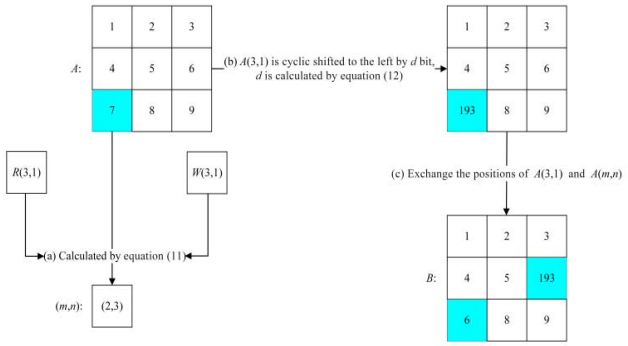


Fig. 9. Scrambling operation of a pixel.

Meanwhile, the scrambling order in this paper is as follows:

Step 1: Scramble the $N-1$ elements (i.e. $A(M,1)$, $A(M,2)$, $\cdots$, $A(M,N-1)$) in the M-th row of the image to obtain $B(M,1\ to\ N-1)$.

Step 2: Scramble the $M-1$ elements (i.e. $A(1,N)$, $A(2,N)$, $\cdots$, $A(M-1,N)$) in the N-th column of the image to obtain $B(1\ to\ M-1,N)$.

Step 3: Scramble the $N-1$ elements (i.e. $A(1,1)$, $A(1,2)$, $\cdots$, $A(1,N-1)$) in the first row of the image to obtain $B(1,1\ to\ N-1)$.

Step 4: Perform the operation of Step 3 from the second row to the $M-1$ row of the image successively to obtain $B(2\ to\ M-1,1\ to\ N-1)$.

Step 5: Scramble the element $A(M,N)$ to obtain $B(M,N)$.

D. Backward Bit Level Diffusion

In the process of backward bit level diffusion, we first XOR the highest bit plane of the scrambled image with the highest bit plane of the random matrix, and then diffuse them one by one from the highest bit plane to the lowest bit plane. The specific diffusion process is as follows: Firstly, decompose matrix B and random matrix S into 8 bit planes: B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8 and S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8. Then diffuse the 8 bit planes of B through (13). Finally, merge C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8 into C.

At this point, the whole encryption process is completed, and the obtained C is the cipher image. In Fig. 10, we give the encryption flow chart of the proposed algorithm.

E. Decryption Algorithm

The decryption process is the inverse operation of the encryption process, and the flow chart of the decryption algorithm is shown in Fig. 11.

IV. Simulation Results and Security Analyses

In this section, we first give the results of encryption and decryption, and then evaluate the security performance of the proposed algorithm by analyzing the following indicators: key space, histogram, correlation analysis of adjacent pixels, key sensitivity, differential attack, information entropy, and comparison with other algorithms.

$$\begin{aligned} C_8 &= (B_8 \oplus S_8) \oplus T \\ C_7 &= (B_7 \oplus S_7) \oplus C_8 \\ C_6 &= (B_6 \oplus S_6) \oplus C_7 \\ C_5 &= (B_5 \oplus S_5) \oplus C_6 \\ C_4 &= (B_4 \oplus S_4) \oplus C_5 \\ C_3 &= (B_3 \oplus S_3) \oplus C_4 \\ C_2 &= (B_2 \oplus S_2) \oplus C_3 \\ C_1 &= (B_1 \oplus S_1) \oplus C_2 \end{aligned} \tag{13}$$

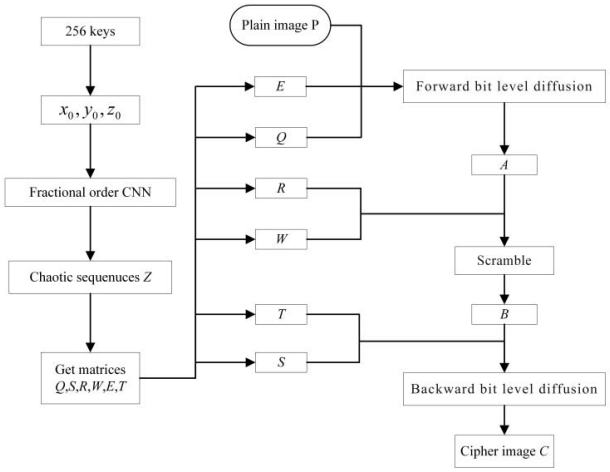


Fig. 10. The encryption flow chart of the proposed algorithm.

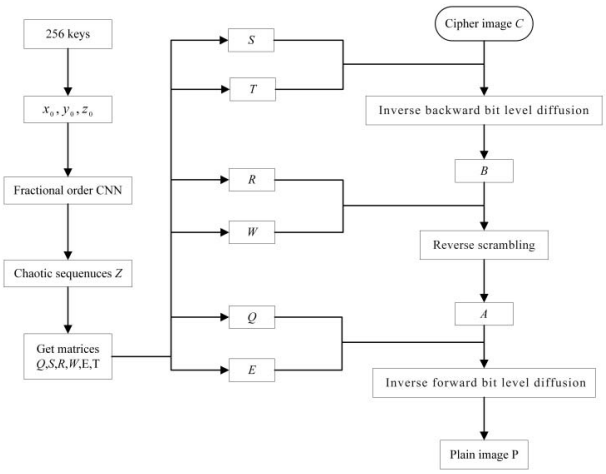


Fig. 11. The flow chart of the decryption algorithm.

A. Encryption and Decryption Results

Fig. 12 shows the encryption and decryption results of three plain images. From the results, we can see that the encrypted image is like noise, and there is a huge visual difference with the plain image, while the image decrypted by the correct key is visually identical to the plain image.

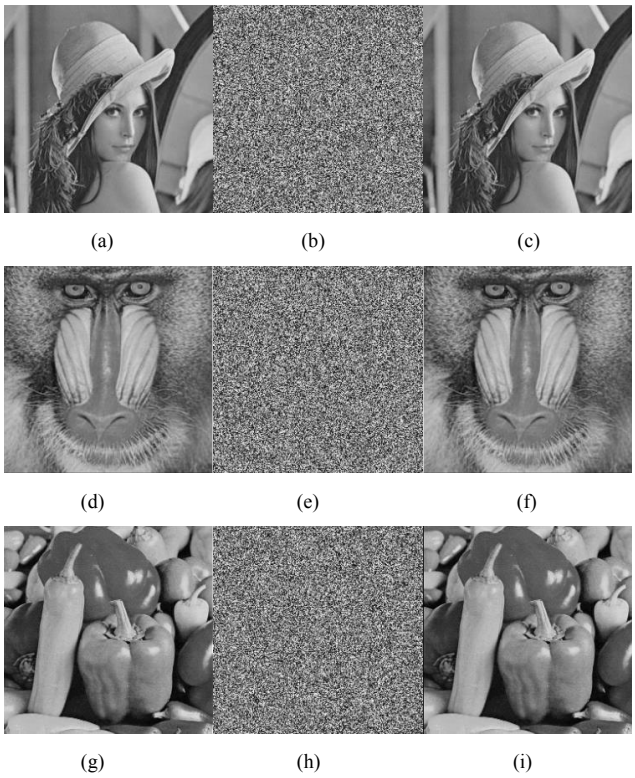


Fig. 12. Encrypted and decrypted results: (a) Lena, (b) cipher image of Lena, (c) decrypted image of Lena, (d) Baboon, (e) cipher image of Baboon, (f) decrypted image of Baboon, (g) Pepper, (h) cipher image of Pepper, and (i) decrypted image of Pepper orders.

B. Key Space Analysis

In order to resist violent attacks effectively, the key space of image encryption algorithm must be large enough. Refer-ence [15] indicated that the size of the key space should not be smaller than 2^{100}. The key of the proposed algorithm is composed of 256-bit binary hash values, and its key space size is 2^{256}, which is greater than 2^{100}. Thus, the key space is large enough to resist violent attacks.

C. Histogram Analysis

Fig. 13 shows the histogram of the image before and after encryption, and it can be seen that the pixel values of the cipher image are uniformly distributed. Therefore, the proposed algorithm is robust against statistical analysis attacks. In addition, chi-square is introduced to quantitatively analyze the uniformity of histograms. It is defined as follows:

$$\chi^2 = \sum_{i=0}^{255} \frac{(f_i - g)^2}{g} \tag{14}$$

where $g = M \times N / 256$, and f_i is the occurrence frequency of the pixels whose value is i. When the significance level is 0.01, the chi-square value should be less than 284.3359. It can be seen from Table I that the chi-square values of the cipher image are all less than 284.3359, so the cipher images encrypt-ed by the proposed algorithm have the better randomness. As a result, the proposed algorithm can make statistical attacks invalid.

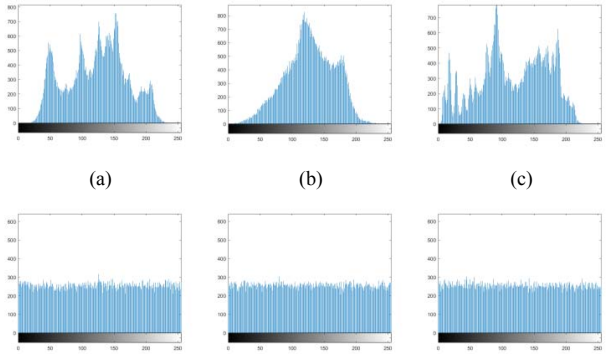


Fig. 13. Histograms of the images:(a) Lena, (b) Baboon, (c) Pepper, (d) cipher image of Lena, (e) cipher image of Baboon, and (f) cipher image of Pepper.

TABLE I. CHI-SQUARE FOR PLAIN AND CIPHER IMAGES

Image	χ^2			Decision
	Plain	Cipher	$\chi^2_{0.01}(255)$	
Lena	4.1101×10^4	266.3828	284.3359	Pass
Baboon	5.8449×10^4	205.6797	284.3359	Pass
Pepper	3.3126×10^4	243.0625	284.3359	Pass

D. Correlation Analysis

The lower the correlation between adjacent elements in the cipher image is, the more it can withstand statistical attacks. Here, we randomly selected 2000 pairs of adjacent pixels in the plain image and cipher image to calculate the correlation coefficient r_{xy} by (15).

$$\begin{cases} r_{xy} = \dfrac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \\ D(x) = \dfrac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2 \\ cov(x,y) = \dfrac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \\ E(x) = \dfrac{1}{N}\sum_{i=1}^{N} x_i \end{cases} \tag{15}$$

where x_i and y_i are the gray values of the pixels and N is the number of selected pairs of pixels. The calculated results are shown in Table II. It can be seen from the results that the corre-

lation of adjacent pixels in the plain image is close to 1, and the correlation of adjacent pixels in the cipher image is close to 0. And from Fig. 14, we find that the pixel values of the cipher image are uniformly distributed, which is quite different from the distribution of the pixel values of the plain image.

In Table III, the correlation coefficients of adjacent pixels are compared with other algorithms, and the results show that the proposed algorithm has some merits compared with [16-19].

TABLE II. THE CORRELATION COEFFICIENT OF THE ADJACENT PIXELS

Image	Direction	Plain	Cipher
Lena	Horizontal	0.9683	0.0010
	Vertical	0.9509	–0.0005
	Diagonal	0.9099	0.0058
Baboon	Horizontal	0.8887	0.0039
	Vertical	0.8978	0.0030
	Diagonal	0.7987	–0.0124
Pepper	Horizontal	0.9385	0.0079
	Vertical	0.9415	–0.0122
	Diagonal	0.8908	–0.0006

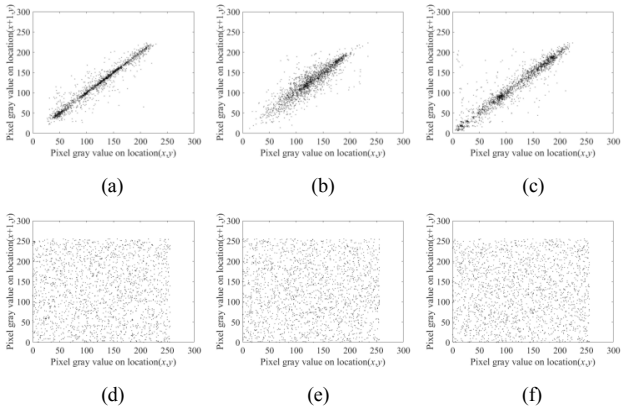


Fig. 14. Correlation scatterplots of image in horizontal direction: (a) Lena, (b) Baboon, (c) Pepper, (d) cipher image of Lena, (e) cipher image of Baboon, and (f) cipher image of Pepper.

TABLE III. COMPARISON ON CORRELATION COEFFICIENTS FOR LENA

Algorithm	Horizontal	Vertical	Diagonal
Proposed	0.0010	–0.0005	0.0058
Ref. [16]	–0.0024	–0.0086	0.0402
Ref. [17]	0.0019	0.0038	0.0019
Ref. [18]	0.0078	–0.0078	0.0013
Ref. [19]	0.0083	–0.0021	–0.0025

E. Key Sensitivity Analysis

Key sensitivity is an important index to evaluate the effectiveness of an encryption algorithm. In this analysis, we first use SHA-256 function to generate a key K, and use this key to encrypt the plain image to obtain the cipher image, then randomly change one bit of the key to obtain a new key K_1, and use the new key K_1 to encrypt the same image to obtain the new cipher image. Finally, the difference between the two cipher images is analyzed. Here, we use the number of pixels change rate (NPCR) and unified average changing intensity (UACI) to analyze the difference between two cipher images. Reference [20] indicated that the ideal values of the NPCR and UACI for two random noise images with 256 gray levels are 99.61% and 33.46%, respectively. NPCR and UACI are defin-ed as follows:

$$\begin{cases} NPCR = \dfrac{1}{M \times N} \displaystyle\sum_{i=1}^{M} \sum_{j=1}^{N} D(i,j) \times 100\% \\ UACI = \dfrac{1}{M \times N} \displaystyle\sum_{i=1}^{M} \sum_{j=1}^{N} \dfrac{|C_1(i,j) - C_2(i,j)|}{255} \times 100\% \end{cases} \tag{16}$$

where

$$D(i,j) = \begin{cases} 1, C_1(i,j) \neq C_2(i,j) \\ 0, C_1(i,j) = C_2(i,j) \end{cases} \tag{17}$$

C_1 and C_2 represent two different cipher images, $M \times N$ represents the size of image.

We randomly changed K for 24 times, and calculated the NPCR and UACI values of the ciphertext images respectively. Their average values are shown in Table IV. From the results, we can see that the values of NPCR and UACI are close to the ideal value, so the proposed algorithm is key sensitive.

TABLE IV. THE AVERAGE VALUES OF NPCR AND UACI BETWEEN CIPHER IMAGES

Image	NPCR(%)		UACI(%)	
	Calculated value	Ideal value	Calculated value	Ideal value
Lena	99.61	99.61	33.50	33.46
Baboon	99.61	99.61	33.44	33.46
Pepper	99.61	99.61	33.49	33.46

F. Differential Attack Analysis

A good image encryption algorithm should be able to make small differences in the plaintext to cause huge changes in the ciphertext. To test the performance of the proposed algorithm against differential attacks, we randomly select a pixel in the plain image P_1 and slightly modify its value through (18) to obtain an image P_2, and then the proposed algorithm is used to encrypt the images P_1 and P_2 respectively to obtain cipher images C_1 and C_2. Finally, NPCR and UACI are used to measure the difference between image C_1 and image C_2. Ten pixels were randomly selected to calculate the NPCR and UACI values of the ciphertext images respectively. Their average values are shown in Table V. The results show that the values of NPCR and UACI are close to the ideal value, so the proposed algorithm can resist differential attack effectively.

$$\text{value} = \text{mod}(\text{value} + 1, 256) \tag{18}$$

TABLE V. NPCR AND UACI VALUES OF CIPHER IMAGES

Image	NPCR(%)	UACI(%)
Lena	99.61	33.44
Baboon	99.61	33.46
Pepper	99.61	33.45

G. Information Entropy Analysis

In order to evaluate the encryption effect of the proposed algorithm, we introduce information entropy to analyze the unpredictability and randomness of the image information before and after encryption. The information entropy of the information source s can be computed by (19):

$$H(s) = \sum_{i=0}^{L} p(s_i) \log \frac{1}{p(s_i)} \tag{19}$$

where L is the grayscale grade of the image, and $p(s_i)$ is the probability of the grayscale value s_i.

For a truly random image with a gray level of 256, the theoretical value of its information entropy is 8. Table VI illustrates the entropy values of the plain and cipher images. We can see that the information entropy of the image encrypted by the proposed algorithm is close to the theoretical value. Meanwhile, Table VII gives the comparison result with other algorithms for Lena. It can be seen that the entropy value of the proposed algorithm is not lower than that of [16-19].

TABLE VI. INFORMATION ENTROPY FOR PLAIN AND CIPHER IMAGES

Image	Plain	Cipher
Lena	7.4416	7.9971
Baboon	7.2636	7.9977
Pepper	7.5553	7.9973

TABLE VII. INFORMATION ENTROPY COMPARISON OF CIPHER IMAGE

Image	Proposed	Ref. [16]	Ref. [17]	Ref. [18]	Ref. [19]
Lena	7.9971	7.9897	7.9971	7.9956	7.9971

V. CONCLUSION

This paper proposes an image encryption algorithm based on a new fractional order CNN. Firstly, through a series of dynamic analyses, the dynamic behavior of the proposed CNN is revealed, and its chaotic characteristics are verified. Second-ly, an encryption construction of forward diffusion, scrambling, and backward diffusion is adopted. This encryption construct-ion can fully hide plaintext information. Meanwhile, the encry-ption algorithm includes bit level and pixel level operations, which has better security performance. In addition, the SHA-256 hash value of the plain image is used to obtain the initial values of the fractional order CNN. Thus the proposed algorit-hm is highly sensitive to plain image. Finally, the performance of the proposed algorithm resist attacks are verified by key sensitivity analysis, differential attack analysis, information entropy analysis and etc.

REFERENCES

[1] Hongxiang Zhao, Shucui Xie, Jianzhong Zhang, Tong Wu, "A dynamic block image encryption using variable-length secret key and modified Henon map," Optik, 230, 166307(1-22), 2021.

[2] Xiangjun Wu, Haibin Kan, Jürgen Kurths, "A new color image encrypt-ion scheme based on DNA sequences and multiple improved 1D chaotic maps," Applied Soft Computing, 37:24-39, 2015.

[3] Yueping Li, Chunhua, Wang, Hua Chen, "A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permu-tation," Optics & Lasers in Engineering, 90:238-246, 2017.

[4] Saiyma Fatima Raza, Vishal Satpute, "A novel bit permutation-based image encryption algorithm," Nonlinear Dynamics, 95(2):859-873, 2019.

[5] Ali Mansouri, Xingyuan Wang, "A novel one-dimensional chaotic map generator and its application in a new index representation-based image encryption scheme," Information Sciences, 563:91-110, 2021.

[6] Tong Wu, Shucui Xie, Jianzhong Zhang, Hongxiang Zhao, "Color image encryption algorithm based on the position index and chaos theory," Journal of Electronic Imaging, 28(5), 053008(1-11), 2019.

[7] Lanlan Huang, Ju H. Park, Guocheng Wu, Zhiwen Mo, "Variable-order fractional discrete-time recurrent neural networks," Journal of Computa-tional and Applied Mathematics, 370, 112633, 2019.

[8] Li Cui, Chaoyang Chen, Jie Jin, Fei Yu, "Dynamic analysis and FPGA implementation of new chaotic neural network and optimization of traveling salesman problem," Complexity, vol. 2021, 5521192, 2021.

[9] Li Cui, Ming Lu, Qingli Ou, Hao Duan, Wenhui Luo, "Analysis and circuit implementation of fractional order multi-wing hidden attractors," Chaos, Solitons & Fractals, 138, 109894, 2020.

[10] Y. Liping Chen, Hao Yin, Liguo Yuan, António M. Lopes, et al., "A novel color image encryption algorithm based on a fractional-order discrete chaotic neural network and DNA sequence operations," Fronti-ers of Information Technology & Electronic Engineering, 21(6):866-879, 2020.

[11] Hongxiang Zhao, Shucui Xie, Jianzhong Zhang, Tong Wu, "Efficient image encryption using two-dimensional enhanced hyperchaotic Henon map," Journal of Electronic Imaging, 29(2), 023007(1-27), 2020.

[12] Diethelm K., Ford N. J., Freed A. D., "A predictor-corrector approach for the numerical solution of fractional differential equations," Nonlinear Dynamics, 29:3-22, 2002.

[13] Georg A. Gottwald, Ian Melbourne, "Testing for chaos in deterministic systems with noise," Physica D: Nonlinear Phenomena, 212(1-2):100-110, 2005.

[14] Malihe Sabeti, Serajeddin Katebi, Reza Boostani, "Entropy and comple-xity measures for EEG signal classification of schizophrenic and control participants," Artificial Intelligence in Medicine, 47(3):263-274, 2009.

[15] Gonzalo Álvarez, Shujun Li, "Some basic cryptographic requirements for chaos-based cryptosystems," International Journal of Bifurcation and Chaos, 16(8):2129- 2151, 2006.

[16] Xiangjun Wu, Haibin Kan, Jürgen Kurths, "A new color image encryption scheme based on DNA sequences and multiple improved 1d chaotic maps," Applied Soft Computing, 37:24-39, 2015.

[17] Xingyuan Wang, Lintao Liu, Yingqian Zhang, "A novel chaotic block image encryption algorithm based on dynamic random growth techni-que," Optics and Lasers in Engineering, 66:10-18, 2015.

[18] M. Jun Mou, Feifei Yang, Ran Chu, Yinghong Cao, "Image compress-ion and encryption algorithm based on hyper-chaotic map," Mobile Networks and Applications, 26:1849-1861, 2019.

[19] Xingyuan Wang, Shengnan Chen, Yingqian Zhang, "A chaotic image encryption algorithm based on random dynamic mixing," Optics & Laser Technology, 138, 106837, 2021.

[20] Jing Chong, Shucui Xie, Jianzhong Zhang, and Dingqin Liu, "Block color image encryption algorithm based on elementary cellular automata and DNA sequence operations," Journal of Electronic Imaging, 30(4), 043025(1-23), 2021.