A new multi-wing hyperchaotic system and its application in image encryption

Pengfei Ding*, Weiwei Hu, Penghui Geng, Juan Zhang, Jingge Zhu

School of Electronics and Engineering, Xi'an University of Posts and Telecommunications, Xi'an, Shaanxi, 710121, People's Republic of China
*Author to whom any correspondence should be addressed
E-mail: dpf@xupt.edu.cn

Abstract

In this study, a novel image encryption scheme based on a multi-wing hyperchaotic system is constructed and its performance is comprehensively evaluated. By introducing a new segmented linear function, a multi-wing hyperchaotic system with high complexity and stochasticity is constructed for image encryption. The algorithm utilizes bit plane decomposition, Zigzag disambiguation, forward diffusion and backward diffusion to encrypt the image efficiently, while combining with the key generation mechanism to achieve a high degree of randomness and security. The experimental results show that the encrypted images with uniformly distributed histograms, information entropy close to the theoretical maximum, and correlation of neighboring pixels close to zero, demonstrating excellent resistance to statistical attacks. Moreover, the image encryption algorithm has outstanding performance in terms of security indexes: the pixel change rate (NPCR) is close to 99.61%, and the uniform change intensity (UACI) is close to 33.43%, both of which are close to the theoretical optimal value, indicating that the algorithm has a strong resistance to differential attacks. In the robustness test, even if the ciphertext image suffers from noise attack or cropping attack, the decrypted image can still effectively recover the original information body, showing good anti-interference performance. Compared with the existing algorithms. The combined advantages of security, robustness and efficiency make it valuable in real-time encryption scenarios, and this research has injected new design inspirations for the development of image encryption techniques guided by chaos theory.

Keywords: multi-wing hyperchaotic chaotic system, image encryption, forward diffusion, backward diffusion, bit plane decomposition.

1 Introduction

Chaos theory is iconic in the discipline of nonlinear dynamics, in-depth analysis of the characteristics of the chaotic system can provide theoretical guidance for the study of nonlinear phenomena in real life. Chaotic system refers to the existence of uncertain random-like chaotic phenomena within a definite system, chaotic systems exhibit strong sensitivity characteristics to changes in initial values and control parameters, a small disturbance of the system parameters will cause a great change of the phase trajectory of the chaotic system, and it has the characteristics of nonlinearity, unpredictability, and stochastic-like etc[1]. As scholars' research on chaotic systems becomes more and more in-depth, it is found that the characteristics of chaotic

systems are applicable to many fields, such as confidential communication[2], image encryption[3], medicine[4], complex neural networks[5].

Chaotic systems received extensive attention from scholars in various fields, but most of the current work is mainly concerned with low-dimensional chaotic systems, and less research focus on hyperchaotic systems and multi-wing chaotic systems, which are more complex in dynamical behaviors[6]. Hyperchaotic systems are chaotic systems with multiple positive Lyapunov exponents, can generate more complex chaotic attractors than ordinary chaotic systems, its dynamical behaviors evolve in multiple directions, thus hyperchaotic systems have higher complexity as well as greater stochasticity and unpredictability[7]. Constructing hyperchaotic systems is usually achieved by adding parametric perturbations, memristors or periodic forcing signals to the original chaotic system[8]. Compared with classical chaotic systems, multi-wing chaotic systems present a more complex spatial topology, which makes the chaotic sequences generated by multi-winged chaotic systems possess a stronger randomness[9]. In the wave of technological development, the traditional low-dimensional chaotic model can no longer meet the practical needs, and the hyperchaotic system with multi-wing structure has become a hot spot of research in the field.

Nowadays, the world is connected by the Internet, and the network has been integrated into people's daily life, especially with the rapid development and wide application of big data technology. There are uncountable amounts of information being generated and transmitted across the network almost every moment, and the corresponding information security issues are also getting more and more attention. As an important data carrier with rich information, there is a risk of unauthorized users attacking and illegally accessing the image to leak the content when it is stored and transmitted, so securing images has always been a hot topic in the field of information security. Image encryption is an important method to help images effectively resist various attacks and avoid data leakage, using keys and encryption algorithms to convert images into informative noise-like or texture-like images, thus preventing illegal visitors from obtaining useful information from them, and only authorized users with the correct key can decrypt and recover the original image. However, the traditional text and data encryption schemes are not applicable to image encryption, because the image itself has a large data capacity, strong correlation, high redundancy[10, 11]. In order to solve this problem, based on the chaos theory proposed by Lorenz[12] and the mathematical definition of chaos given by Li and Yorke[13], researchers gradually found that chaotic systems perform well in image encryption[14-16]. And along with the development and application of chaotic systems, chaos-based image encryption techniques have gradually become popular, attracting extensive attention from researchers and exhibiting excellent performance in image encryption[17-19].

There are two existing image encryption steps using chaos. In the first one, the pixel is used as the smallest unit of operation, and the purpose of encrypting the image is achieved by performing a chaotic diffusion operation on the pixel. The other is that a pixel can be further divided into several bits and the encryption process is

accomplished by shifting and exchanging each bit. According to the literature[20], the distribution of information in the high and low bits plane in the bit plane structure is extremely uneven, with the former occupying more than 94% of the pixel information. In literature[21], Ding et al. encrypted only the high four bits of each pixel of the image and did not operate on the low four bits of the pixel, this algorithm reduces the execution time as it is equivalent to encrypting only half of the pixels. In a conventional disorder operation, two pixel positions are typically exchanged. Extending this idea further to the bit-level level, that is, swapping a certain bit in one pixel with a certain bit in another pixel. By doing so, the information between the two pixels is interchanged, and at the same time, their original values are changed. So, just one bit-level operation combines the effect of disruption and diffusion.

The paper is organized as follows: first, the theoretical model of the proposed multi-winged hyperchaotic system is described in Section 2, then the dynamical behavior of the multi-wing hyperchaotic system is analyzed in Section 3, and standard test methods such as the 0-1 test, the initial sensitivity test, the complexity test, and the NIST test are carried out, and then, in Section 4, by integrating the bit-plane decomposition, the Zigzag disruption, and the bidirectional diffusion algorithm, the A novel image encryption scheme based on this hyperchaotic system is proposed. The proposed encryption algorithm is analyzed for performance and compared with other encryption schemes in Section 5, and finally Section 6 systematically summarizes the research results of the whole paper.

2 Model of the new hyperchaotic system

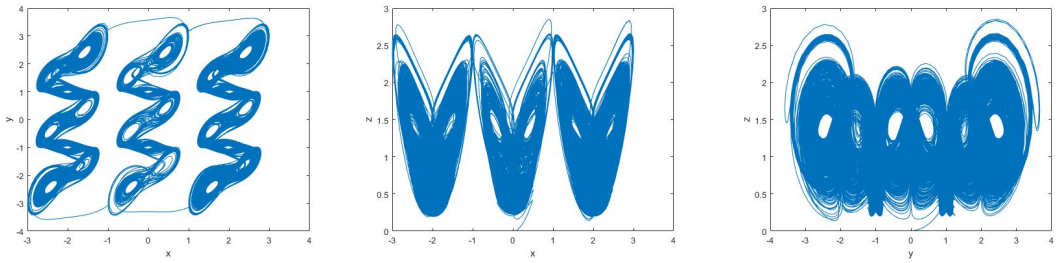
A new multi-wing hyperchaotic system is obtained by adding a first-order differential equation with respect to w to the Lorentz system, and designing a new segmented linear function $g(x)$ and $f(y)$ using the sign function. A new multi-wing hyperchaotic system is obtained by utilizing the segmented linear function $g(x)$ instead of the state variable x, and utilizing the segmented linear function $f(y)$ instead of the state variable y. The equations of the multi-wing hyperchaotic system are given as equation (1) and equation (2).

$$\begin{cases} a(f(y)-g(x))=0 \\ cg(x)-f(y)-kzg(x)+w=0 \\ kf(y)g(x)-bz=0 \\ -hg(x)=0 \end{cases} \tag{1}$$

$$\begin{cases} \mathrm{g}(x)=x-\mathrm{sgn}(x-1)-\mathrm{sgn}(x+1) \\ f(y)=y-\mathrm{sgn}(y-1)-\mathrm{sgn}(y+1) \end{cases} \tag{2}$$

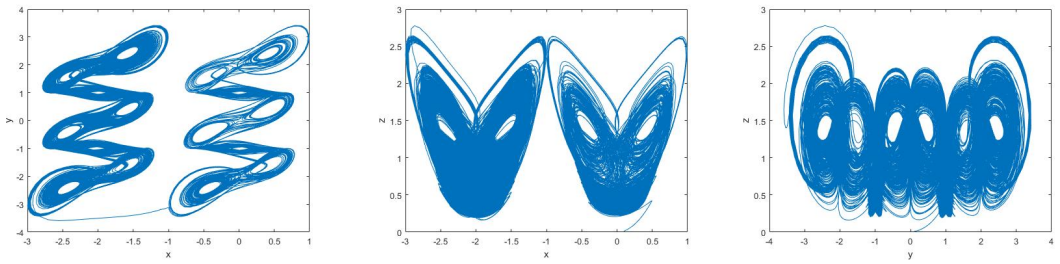
In equation (1), a, b, c, k, and h are the parameters of the system, and x, y, z, and w are the state variables of the system. When a=10, b=8/3, c=28, k=19.9, h=4.7, and the initial values are [0.1,0.1,0.01,0.01], the four Lyapunov exponents of the system are LE1=1.4214, LE2=0.6162, LE3=0.6726, LE4=−2.8770. The system has three Lyapunov exponents greater than zero, and one Lyapunov exponent less than zero, so at this time the system is a hyperchaotic system. And at this time the system can produce 3×3 grid multi-wing hyperchaotic attractor, Fig.1 shows the phase diagram of the system. Take the system parameter k=20, when the initial conditions and other

parameters are not changed, the system at this time is a 2 × 3 grid multi-wing hyperchaotic attractor and shown in Fig.2. When the system parameter k=22, the phase diagram of the system at this time is a 1 × 3 multi-wing hyperchaotic attractor and shown in Fig.3. Continue to change the system parameter k and adjust its value to k=27, the system shown in Fig.4 is a 1 × 2 multi-wing hyperchaotic system.



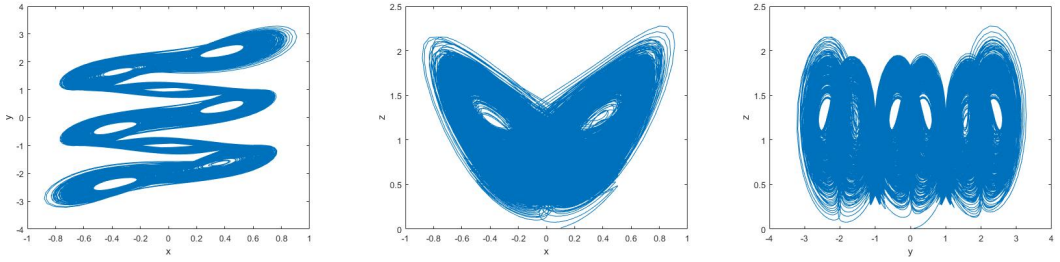
(a) (b) (c)

Fig.1 3 × 3 grid multi-wing chaotic attractor phase diagram. (a)on x-y plane (b)on x-z plane (c)on y-z plane



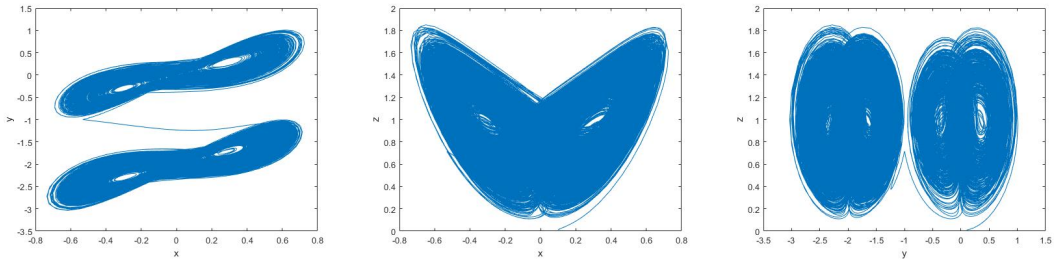
(a) (b) (c)

Fig.2 2 × 3 grid multi-wing chaotic attractor phase diagram.. (a)on x-y plane (b)on x-z plane (c)on y-z plane



(a) (b) (c)

Fig.3 1 × 3 grid multi-wing chaotic attractor phase diagram.. (a)on x-y plane (b)on x-z plane (c)on y-z plane



(a) (b) (c)

Fig.4 1 × 2 grid multi-wing chaotic attractor phase diagram.. (a)on x-y plane (b)on x-z plane (c)on y-z plane

3 Dynamic analysis of the new chaotic system

3.1 Dissipative analysis of multi-wing hyperchaotic systems

The dispersion of the system is calculated by equations (1) and (2) as:

$$\nabla V = \frac{\partial \dot{x}_0}{\partial x_0} + \frac{\partial \dot{y}_0}{\partial y_0} + \frac{\partial \dot{z}_0}{\partial z_0} + \frac{\partial \dot{w}_0}{\partial w_0} = -a - 1 - b \tag{3}$$

Substituting the system parameters of a=10, b=8/3 into equation (3) yields $\nabla V = -\frac{41}{3} < 0$, so the system is dissipative and shrinks exponentially.

$$\frac{dV}{dt} = e^{-\frac{41}{3}t} \tag{4}$$

It follows that the volume elements of the volume element V_0 at moment t contract to $e^{-\frac{41}{3}t}V_0$.

3.2 Equilibrium point analysis of multi-wing hyperchaotic systems

Calculating the equilibrium point of the new multi-wing hyperchaotic system can be obtained by making the right-hand side of equation (1) equal to zero:

$$\begin{cases} a(f(y) - g(x)) = 0 \\ cg(x) - f(y) - kzg(x) + w = 0 \\ kf(y)g(x) - bz = 0 \\ -hg(x) = 0 \end{cases} \tag{5}$$

Where the system parameters a=10, b=8/3, c=28, k=19.9, h=4.7 are selected, it can be concluded from the equations that there is only one equilibrium point of the system as (0,0,0,0), and the Jacobi matrix J of this equilibrium point can be expressed as follows:

$$J = \begin{bmatrix} -a & a & 0 & 0 \\ c & -1 & 0 & 1 \\ 0 & 0 & -b & 0 \\ -h & 0 & 0 & 0 \end{bmatrix} \tag{6}$$

After substituting the system parameters, the characteristic equation of the system at this equilibrium point is derived as:

$$\lambda^4 + 11\lambda^3 - 270\lambda^2 + 47\lambda = 0 \tag{7}$$

The characteristic root of this equilibrium point can be calculated as (0, −22.887, 0.175, 11.712), which is an unstable saddle point according to the judgment of Routh-Hurwit stability, and it is clear that the system is unstable and produces chaotic attractors.

3.3 Bifurcation diagram and Lyapunov exponent analysis

Fixed system parameters a=10, b=8/3, c=28, h=4.7 and initial conditions are set to (0.1,0.1,0.01,0.01). Fig.5 presents the corresponding Lyapunov exponential spectrum as well as the bifurcation diagram for the system parameter k as it is incremented from 15 to 25. From Fig.5, it can be found that the system is always in a chaotic state. k=16, and k=23 are chosen to simulate the chaotic system under this system parameter

respectively, its phase diagrams are shown in Fig.5.

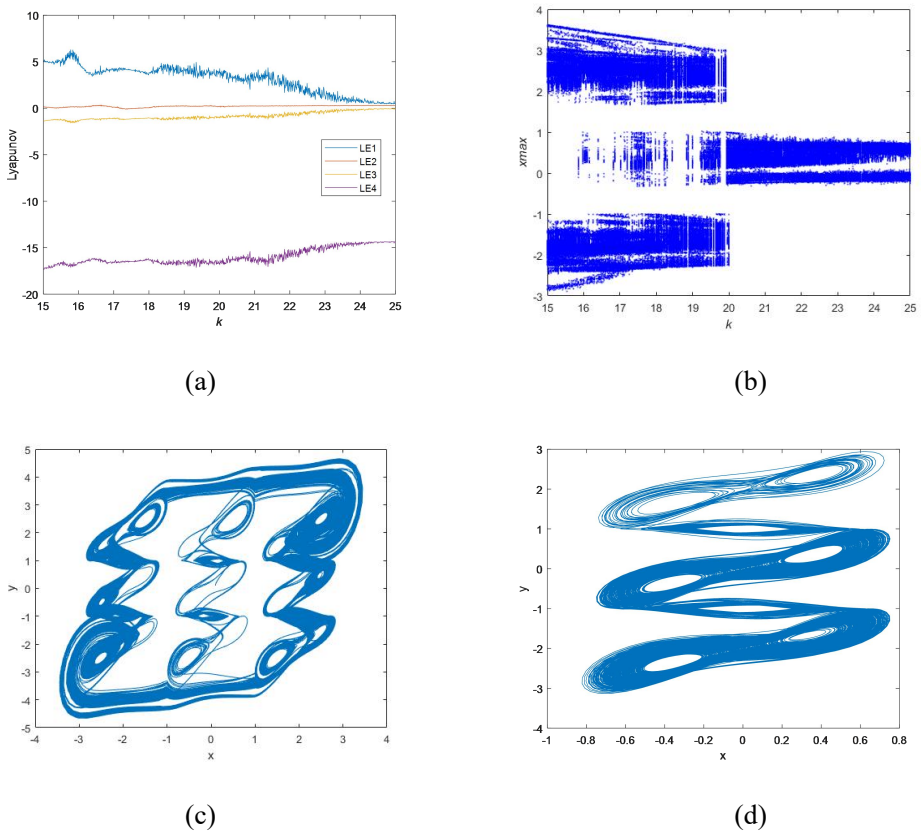
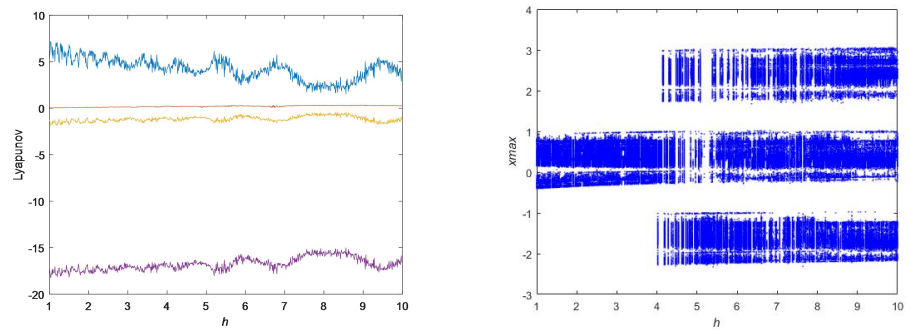
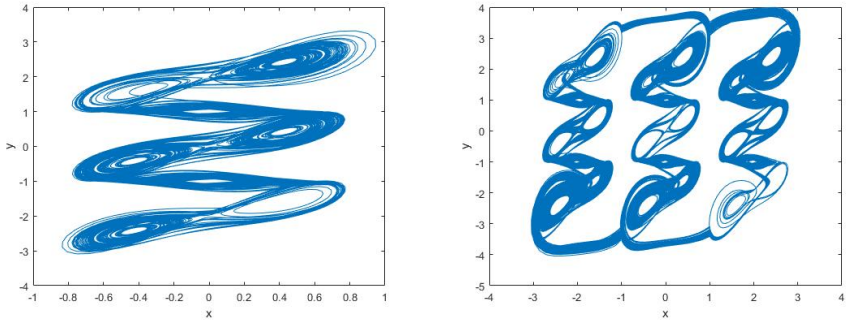


Fig.5 (a) Lyapunov exponents spectrum with $k\in(15,25)$; (b) bifurcation diagram with $k\in(15,25)$. (c) phase diagram with k=16; (d) phase diagram with k=23

The bifurcation diagrams and Lyapunov exponential spectra for the parameter h in the range from 1 to 10 are plotted in Fig.6, with the other system parameters and initial values fixed, and with h as a variable control parameter. The analysis shows that the system is continuously in a chaotic state in this parameter range, and the phase diagrams for h=2 and h=9 are also given in Fig.6.





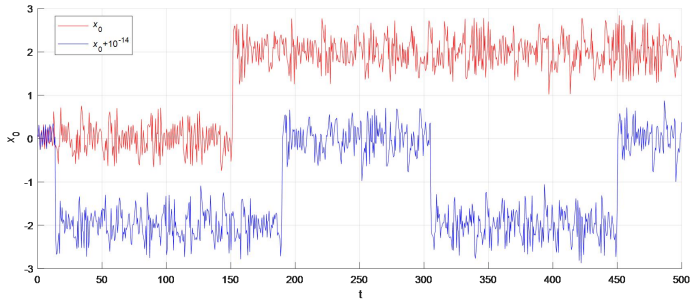
(c) (d)

Fig.6 (a)Lyapunov exponents spectrum with $h \in (1,10)$; (b)bifurcation diagram with h $\in$ (1,10). (c)phase diagram with h=2; (d)phase diagram with h=9

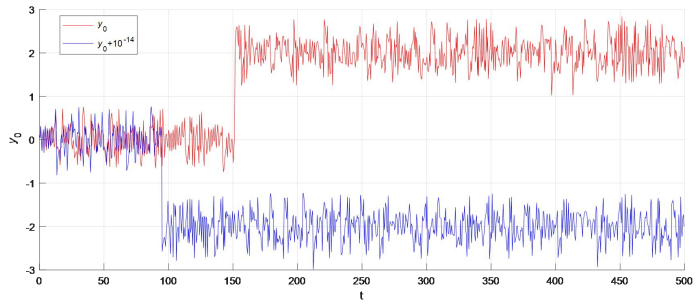
3.4 Sensitivity test of the system to initial conditions

For the given system equations (1) and (2), we systematically analyze the sensitivity of the initial parameter changes to the system behavior, a=10, b=8/3, c=28, k=19.9, h=4.7, The initial state of the system is set as x_0=0.1, y_0=0.1, z_0=0.01 and w_0=0.01, based on these parameters, we analyses its impact on the dynamic behavior of the system by introducing extremely small perturbations to the initial values (adjusting x_0 from 0.1 to $0.1+10^{-14}$).

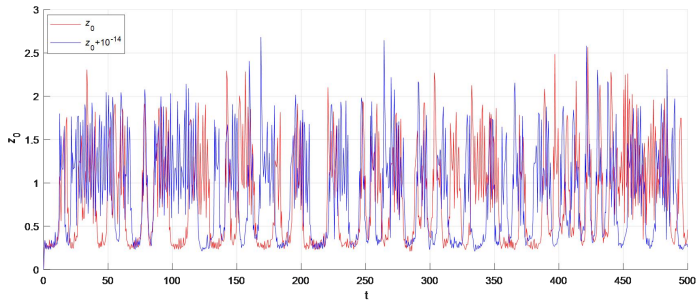
Specifically, we simulate the time series of the state variable x under two initial conditions, an unperturbed initial value of x_0 = 0.1, and an initial value of x_0 = $0.1 + 10^{-14}$ with a small perturbation. As shown in Fig.7(a), this small variation in the initial value leads to a significant separation of the time series of the state variable x after a finite amount of time has elapsed. In addition, we further verify the dynamic behavior of other state variables (y_0, z_0, w_0) under the condition of having different initial values. From the simulation results presented in Fig.7(b)-(d), it can be seen that the time series of the state variables are very significantly different due to the small changes in the initial values. This phenomenon intuitively confirms that the system possesses the property of being highly sensitive to the initial values.



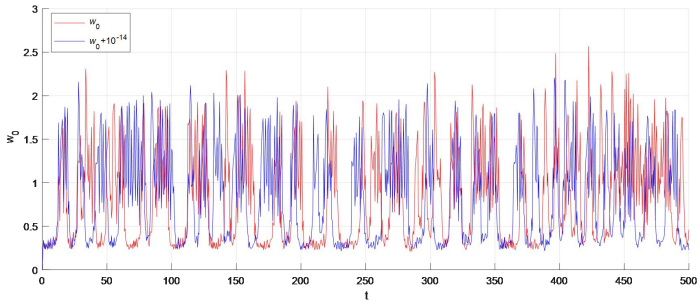
(a)



(b)



(c)



(d)

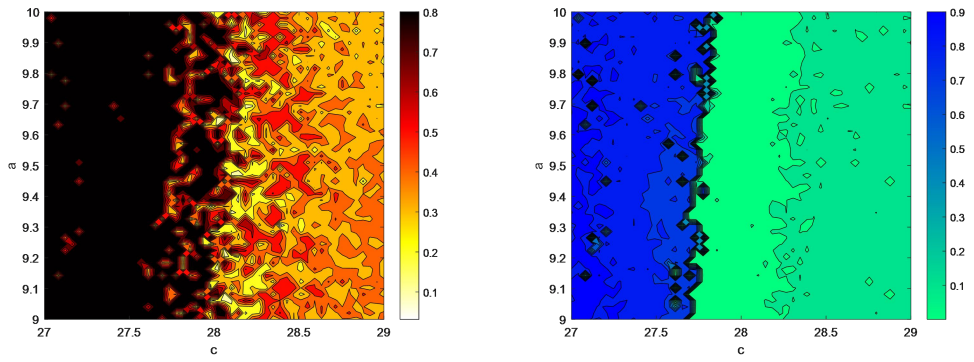
Fig.7 State variable curves over time: (a) time - state variable x_0 curve, (b) time - state variable y_0 curve, (c) time - state variable z_0 curve, (d) time - state variable w_0 curve

3.5 complexity analysis

In the analysis of chaotic systems, complexity effectively quantifies the similarity between chaotic and random sequences, where a higher randomness corresponds to a larger complexity, and the unpredictability of the sequence can be assessed by the complexity. The analysis of complexity can be developed from two dimensions: behavioral complexity and structural complexity. Specifically, in this paper, spectral entropy (*SE*) is used to assess behavioral complexity, while structural complexity is analyzed using the C_0-algorithm.

With the initial conditions of the system set as x_0=0.1, y_0=0.1, z_0=0.01, w_0=0.01 and the parameters fixed as a=10, b=8/3, c=28, k=19.9, h=4.7, we conducted an in-depth analysis of the complexity level for the parameters a and c, and the results are displayed in Fig.8. The diagram uses color shades to show, in an intuitive and clear way, the differences in the complexity of chaotic systems within a given parameter range. The darker areas in the figure correspond to more complex states of

the chaotic system, while the lighter areas reflect the relatively lower complexity of the chaotic system.



(a) (b)

Fig.8 Complexity analysis with respect to parameters *a* and *c*. (a)*SE* complexity; (b)C_0–algorithm complexity

3.6 NIST statistical test

Chaotic pseudo-random number generator, as a key component of chaotic secure communication system, the statistical properties of its output sequence directly determine the cryptographic security of the system. To ensure that the generated sequence possesses the unpredictability of cryptographic strength, this study uses the SP.800-22 statistical test suite developed by the National Institute of Standards and Technology (NIST) for rigorous evaluation. As shown in Table 1, the experimental data adopts *P*-value as the statistical significance criterion: when the *P*-value tends to 1, the sequence is characterized as having ideal randomness, while tends to 0, it indicates the existence of significant non-random patterns. According to the standard specification of cryptography, the significance threshold a=0.01 is set as the judgment benchmark for hypothesis testing, and the randomness hypothesis of the sequence is accepted when the *P* value is not lower than this threshold. The results of the tests show that the chaotic sequences generated by this system obtain *P*-values significantly higher than the threshold level (P>0.01) in all the test items. This fully demonstrates that the chaotic dynamics system designed in this study is capable of generating pseudo-random sequences with excellent statistical properties.

Table 1. NIST statistical test result.

NO.	Statistical test	Reference *P*-value	*P*-value	Results
01	Frequency	≥0.01	0.472819	pass
02	Block frequency	≥0.01	0.732838	pass
03	Runs	≥0.01	0.526914	pass
04	Longest run	≥0.01	0.372271	pass
05	Rank	≥0.01	0.193647	pass
06	Discrete Fourier Transform	≥0.01	0.482718	pass
07	Overlapping template	≥0.01	0.345727	pass
08	Non-overlapping template	≥0.01	0.493282	pass
09	Universal	≥0.01	0.883616	pass
10	Linear complexity	≥0.01	0.473821	pass

11	Approximate entropy	≥0.01	0.327178	pass
12	Cumulative sums	≥0.01	0.572738	pass
13	Serial	≥0.01	0.664873	pass
14	Random excursions	≥0.01	0.394821	pass
15	Random excursions variant	≥0.01	0.193841	pass

4 The image encryption algorithm proposed

4.1 Generation of initial states and parameters

The key generation architecture of this cryptosystem adopts a hybrid construction mechanism, which is synergistically generated by two key structure: one is the user-defined initial key parameter, and the other is obtained by preprocessing the original image through the cryptographic hash function. The 64-bit hexadecimal key obtained by processing a plaintext image using SHA-256 cryptographic hashing algorithm. This is converted to a 256-bit binary sequence K_1, which is then the XOR algorithm is performed with the randomly generated 256-bit secret K_2. K can be obtained from equation (8).

$$K = K_1 \oplus K_2 \tag{8}$$

K is divided into 8 sets of keys $\{k_1, k_2, k_3, k_4, k_5, k_6, k_7, k_8\}$ for every 32 bits. These eight sets of keys are used to generate the initial values of the multi-wing hyperchaotic system as well as the relevant parameters of the system. The operations are shown in equation (9) and equation (10).

$$\begin{cases} x_0 = \mathrm{mod}(\dfrac{k_1}{k_2 + k_3 + k_4}, 1) + x_0' \\ y_0 = \mathrm{mod}(\dfrac{k_2}{k_1 + k_3 + k_4}, 1) + y_0' \\ z_0 = \mathrm{mod}(\dfrac{k_3}{k_1 + k_2 + k_4}, 1) + z_0' \\ w_0 = \mathrm{mod}(\dfrac{k_4}{k_1 + k_2 + k_3}, 1) + w_0' \end{cases} \tag{9}$$

$$\begin{cases} a = \mathrm{mod}(\dfrac{k_5}{k_6 + k_7 + k_8}, 1) + a' \\ b = \mathrm{mod}(\dfrac{k_7}{k_5 + k_6 + k_8}, 1) + b' \\ c = \mathrm{mod}(\dfrac{k_2 + k_3 + k_4}{k_6 + k_7 + k_8}, 1) + c' \\ k = \mathrm{mod}(\dfrac{k_1 + k_2 + k_3}{k_5 + k_6 + k_7}, 1) + k' \\ h = \mathrm{mod}(\dfrac{k_4}{k_5 + k_6 + k_7 + k_8}, 1) + h' \end{cases} \tag{10}$$

In equation(10), x_0, y_0, z_0, w_0 represents the starting value of the proposed multi-wing hyperchaotic system; parameters a, b, c, h, k are the initial setup parameters

of this multi-wing hyperchaotic system, and $x_0', y_0', z_0', w_0', a', b', c', h', k'$ are the custom key values.

4.2 image encryption algorithm

4.2.1 Decomposition of the bit plane

A image has pixel values ranging from 0-255, and each pixel is stored in the computer system in the form of an 8-bit binary code. According to the binary bit weights, the image can be decomposed into 8 independent bit planes ($i \in [1,8]$), where the i-th bit plane consists of the i-th binary bit of all pixels. Experimental data show that the high 4-bit planes (i=5-8) concentrate more than 94% of the visual information of an image, while the low four-bit planes (i=1-4) contain only a few detailed features. Therefore, we used a bit plane merging approach to merge the 8-bit planes two by two. The schematic diagram of bit-plane decomposition and merging is shown in Fig.9 .

Step 1: P' is divided into 8-bit planes $P_7P_6P_5P_4P_3P_2P_1P_0$. The size of the plaintext image P' is assumed to be $M \times N$.

Step 2: Separate the bit planes P_7P_0, P_6P_1, P_5P_2, P_4P_3.

Step 3: The bit planes P_7 and P_0 bit planes are merged into E_1, the bit planes P_6 and P_1 bit planes are merged into E_2, the bit planes P_5 and P_2 bit planes are merged into E_3, and the bit planes P_4 and P_3 bit planes are merged into E_4.

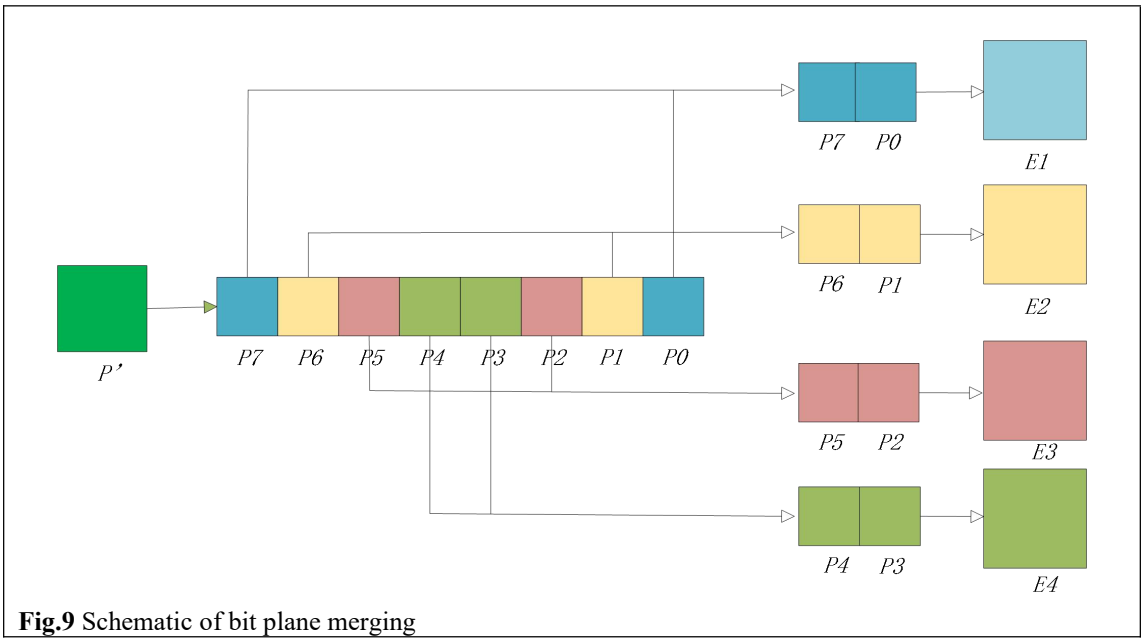


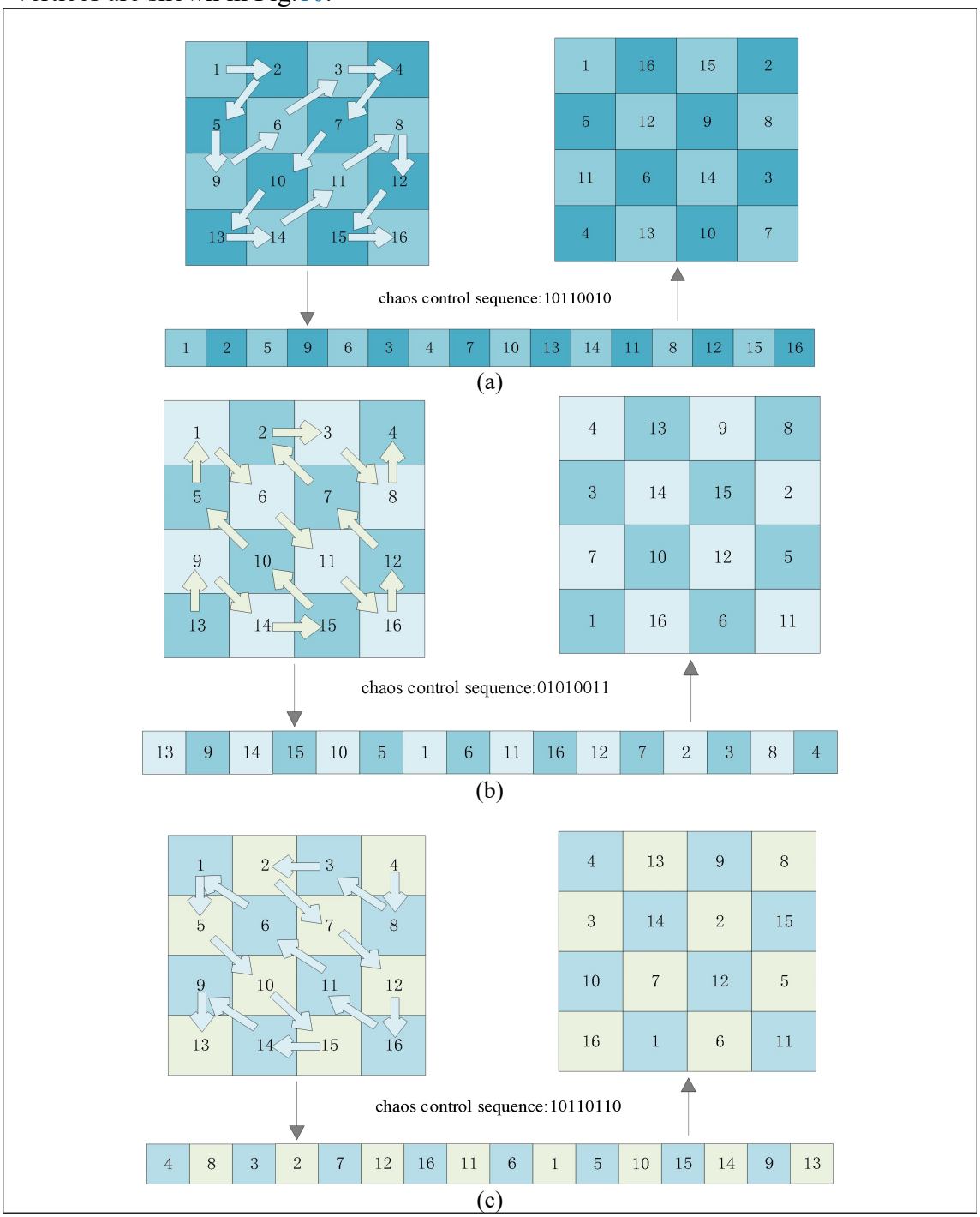
Fig.9 Schematic of bit plane merging

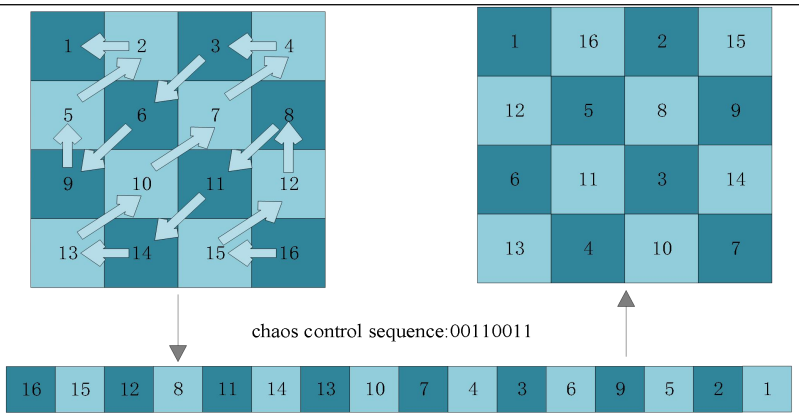
4.2.2 Zigzag disorder

Zigzag transform is a common method for transforming pixel positions, which works by scanning a matrix in a Z fashion and placing the scanned data into a one-dimensional sequence, and then converting the vectors into a two-dimensional matrix in order.

Decompose the bit plane and process the four planes E_1, E_2, E_3 and E_4 to perform Zigzag dislocation with different vertices in the upper left corner, lower left corner, upper right corner, and lower right corner, respectively. Firstly, these four planes are represented by matrices respectively, which are scanned in a Z-scanning mode

according to the start of different vertices and reorganize into one-dimensional sequences in traversal order. The value of the chaotic sequence then determines the scanning direction: when the start element is 1, the left-right alternating extraction strategy is adopted, the elements are extracted from the start and end positions of the one-dimensional matrix synchronously; when the start element is 0, the right-left alternating filling mode is executed. The Zigzag dislocations starting from different vertices are shown in Fig.10.





(d)

Fig.10 Zigzag scrambling.

4.2.3 forward diffusion

The following steps delineate the precise procedures how to complete the forward diffusion, the specific steps give the forward diffusion of the scrambled image P. The image obtained by performing the forward diffusion is denoted as A matrix.

Step1: Using the Q matrix generated by the chaotic cryptography generator, change the grey value of the pixels in the first row of the plaintext image according to equation (11). That is, $P(1,j)$ is transformed into $A(1,j)$, where r_1 and r_2 are 8-bit random integers in the interval [0,255].

$$\begin{cases} A(1,1) = \mathrm{mod}(P(1,1) + Q(1,1) + r_1 + r_2, 256) \\ A(1,j) = \mathrm{mod}(P(1,j) + Q(1,j) + A(1,j-1), 256), j = 2,3,\ldots,N \end{cases} \tag{11}$$

Step2: Using the Q matrix generated by the chaotic cryptographic generator, change the grey value of the pixels in the first column of the plaintext image according to equation (12), thus transform $P(i,1)$ into $A(i,1)$.

$$A(i,1) = \mathrm{mod}(P(i,1) + Q(i,1) + A(i-1,1), 256), i = 2,3,\ldots,M \tag{12}$$

Step3: Using the Q matrix obtained from the chaotic password generator, the gray values of the remaining pixels of the plaintext image except for the first row and the first column are adjusted according to the rules of equation (13), thus realizing the conversion of $P(i,j)$ to $A(i,j)$.

$$A(i,j) = \mathrm{mod}(P(i,j) + A(i-1,j) + A(i,j-1) + Q(i,j), 256) \tag{13}$$

4.2.4 backward diffusion

The backward diffusion of encryption for obtaining the ciphertext image C associated with intermediate ciphertext A, which is transformed from the original image.

Step1: Using the W matrix generated by the chaotic cryptography generator, where r_3 and r_4 are 8-bit random integers with values in the interval [0,255]. The grey values of all pixels in row M of the intermediate ciphertext matrix A are transformed according to equation (14), $A(M,j)$ is transformed into $C(M,j)$.

$$\begin{cases} C(M,N) = \mathrm{mod}((A(M,N) + W(M,N) + r_3 + r_4), 256) \\ C(M,j) = \mathrm{mod}((A(M,j) + W(M,j) + C(M,j+1)), 256), j = N-1, N-2,\ldots,1 \end{cases} \tag{14}$$

Step2: The grey value of the pixel in the N-th column of A is changed in line with equation (15) and W matrix, $A(i, N)$ is transformed into $C(i, N)$.

$$C(i,N) = \mathrm{mod}((A(i,N) + W(i,N) + C(i+1,N)), 256), i = M-1, M-2, \ldots, 1 \quad (15)$$

Step3: By virtue of equation (16) and the W matrix, the conversion from $A(i,j)$ to $C(i,j)$ is realized by implementing the conversion for all pixel gray values in the A matrix excluding row M and column N.

$$C(i,j) = \mathrm{mod}((A(i,j) + C(i+1,j) + C(i,j+1)W(i,j), 256) \quad (16)$$

4.3. Image encryption method

Step 1: Enter the image P' to be encrypted, which has a size of $M \times N$.

Step 2: First, set the initial key vector to iteratively evolve the multi-wing hyperchaotic system, and extract the four chaotic sequences $\{x_i, y_i, z_i, w_i\}$ with good statistical properties by discarding the first 2000 transient points.

Step 3: The plaintext image P' is decomposed into 8 bit planes $P_7P_6P_5P_4P_3P_2P_1P_0$ according to equation (17), combining the bit planes P_7 and P_0 bit planes into E_1, combining the bit planes P_6 and P_1 bit planes into E_2, combining the bit planes P_5 and P_2 bit planes into E_3, and combining the bit planes P_4 and P_3 bit planes into E_4.

$$R_n(i,j) = \frac{R(i,j)}{2^{n-1}} \bmod 2, R_n(i,j) \in [0,1] \quad (17)$$

Step 4: According to equation (18), the chaotic sequence will be transformed into the 01 Sequence. The x_i is used to control E_1 to perform the Zigzag disruption in the upper left corner, and y_i, z_i, w_i are used to control E_2, E_3, E_4 to perform the Zigzag disruption in the lower left corner, the upper right corner, and the lower right corner, respectively. Finally, the four disrupted planes are merged into a bit plane in order to obtain the image P.

$$\begin{cases} x_i = \mathrm{mod}(floor(x_i \times 10^{14}), 2) \\ y_i = \mathrm{mod}(floor(y_i \times 10^{14}), 2) \\ z_i = \mathrm{mod}(floor(z_i \times 10^{14}), 2) \\ w_i = \mathrm{mod}(floor(w_i \times 10^{14}), 2) \end{cases} \quad (18)$$

Step 5: The random sequences x_i and y_i generated iteratively using the chaotic system are used to generate matrices Q and W according to equation (19) and equation (20), where $u = 1, 2, \ldots, M, v = 1, 2, \ldots, N$.

$$Q(u,v) = \mathrm{mod}(floor((\frac{r_1+1}{r_1+r_3+2} x_{(u-1)\times N+v} + \frac{r_3+1}{r_1+r_3+2} y_{(u-1)\times N+v}) \times 10^{14}), 256) \quad (19)$$

$$W(u,v) = \mathrm{mod}(floor((\frac{r_2+1}{r_2+r_4+2} x_{(u-1)\times N+v} + \frac{r_4+1}{r_2+r_4+2} y_{(u-1)\times N+v}) \times 10^{13}), 256) \quad (20)$$

Step 6: According to Sections 4.2.3 and 4.2.4 forward diffusion and backward diffusion are applied to the scrambled image using matrices Q and W, respectively, to obtain the encrypted image C.

4.4 image decryption method

Step 1: Input the encrypted ciphertext image C.

Step 2: Using the provided initial key, iterate the multi-wing hyperchaotic system and discard the results of the first 2000 iterations to ensure a high degree of randomness in the sequence and generate four chaotic sequences $\{x_i,y_i,z_i,w_i\}$.

Step 3: Using the generated chaos matrix W and random integers r_3, r_4, the gray value of each pixel in the encrypted image C is subjected to an inverse diffusion operation that reduces it to a matrix A.

Step 4: The inverse diffusion operation is performed on matrix A using the chaos matrix Q and random integers r_1, r_2 to gradually recover the image P.

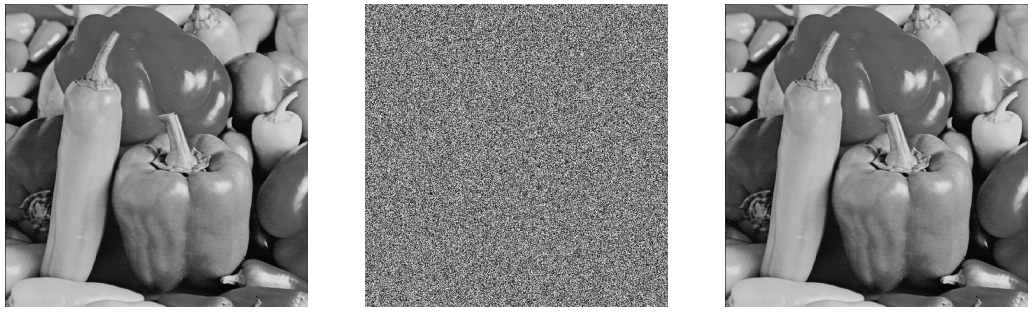
Step 5: Decompose the image P into the planes E_1, E_2, E_3, E_4, perform Zigzag inverse permutation on them, and then restore the positional arrangement of each plane according to the chaotic sequence $\{x_i, y_i, z_i, w_i\}$, and then recombine the 4-bit planes after the restoration to the original bit plane arrangement $P_7P_6P_5P_4P_3P_2P_1P_0$.

Step 6: According to the merge rule, the bit planes are re-merged into the complete plaintext image P'.

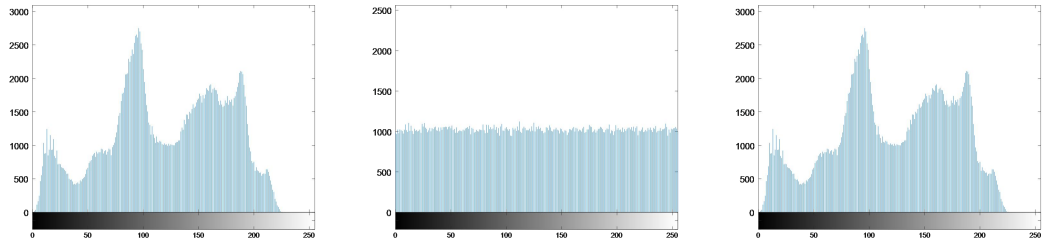
5 Simulation experiment and performance analysis

5.1 Histogram analysis

Histograms are a commonly used statistical and visualization tool for dataset. In a normal plaintext image, its distribution of grey values is usually uneven. Differences in image content can cause the distribution of grey values in the histogram to show significant fluctuations and variations. In image encryption technology, the core objective of the encryption algorithm is to guarantee information security by destroying the statistical regularity of the plaintext image. Specifically, it is necessary to eliminate the statistical characteristics of pixel values through nonlinear transformation, so that the histogram of the encrypted image is homogenized, thus resisting the statistical attack based on the histogram, and ensuring that the attacker can not obtain the original image information through regular statistical analysis. Encrypted and decrypted images of ‘Baboon’, ‘Peppers’, ‘House’ and ‘Plane’ with size 512×512 are shown in Fig.11, 12, 13 and 14 and their histograms are shown in Fig.11, 12, 13 and 14 .

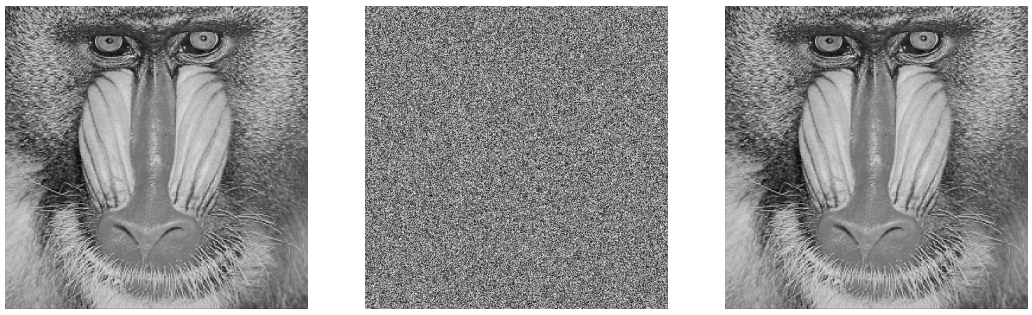


(a) (b) (c)

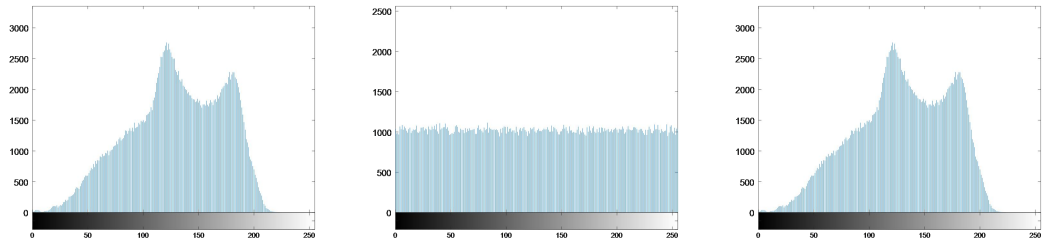


(d) (e) (f)

Fig.11 Visualization of the encryption and decryption results for 'Peppers': (a) Plaintext image (b) Ciphertext image (c) Decrypted image (d) Histogram of plaintext image (e) Histogram of Ciphertext image (f) Histogram of Decrypted image

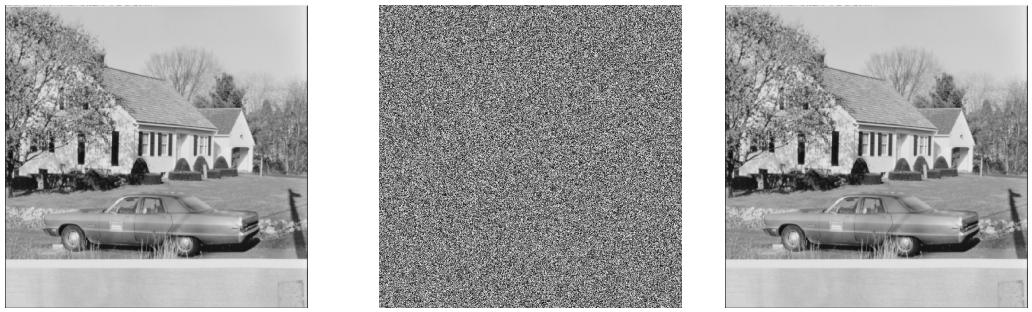


(a) (b) (c)

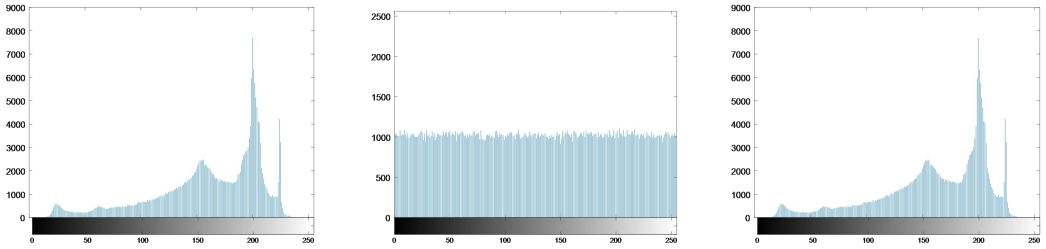


(d) (e) (f)

Fig.12 Visualization of the encryption and decryption results for 'Baboon': (a) Plaintext image (b) Ciphertext image (c) Decrypted image (d) Histogram of plaintext image (e) Histogram of Ciphertext image (f) Histogram of Decrypted image

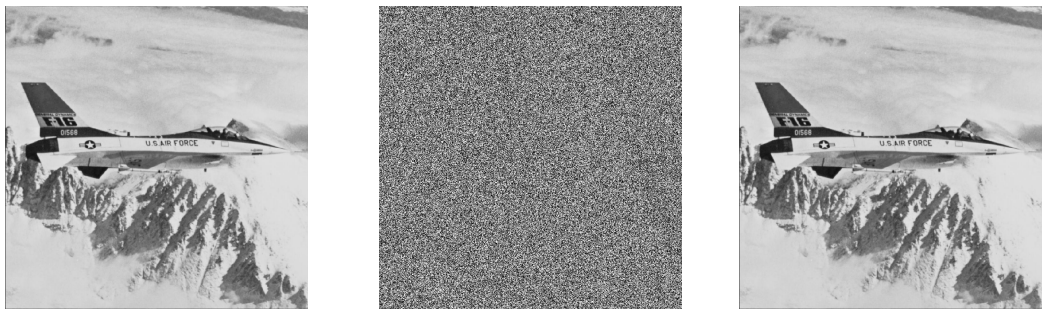


(a) (b) (c)

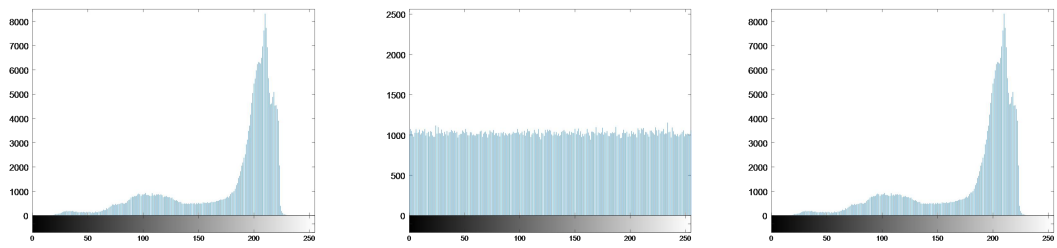


(d) (e) (f)

Fig.13 Visualization of the encryption and decryption results for 'House': (a) Plaintext image (b) Ciphertext image (c) Decrypted image (d) Histogram of plaintext image (e) Histogram of Ciphertext image (f) Histogram of Decrypted image



(a) (b) (c)



(d) (e) (f)

Fig.14 Visualization of the encryption and decryption results for 'Plane': (a) Plaintext image (b) Ciphertext image (c) Decrypted image (d) Histogram of plaintext image (e) Histogram of Ciphertext image (f) Histogram of Decrypted image

5.2 Information entropy analysis

Information entropy is an important mathematical tool for measuring the degree of randomness or disorder with in a dataset. In the field of image processing and encryption, information entropy is widely used to quantify the randomness of image data so as to evaluate the effectiveness of encryption algorithms. For the calculation of image information entropy, Shannon Entropy formula is usually used:

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i \tag{21}$$

Where p_i denotes the probability of occurrence of the i- th gray level, and n is the number of gray levels (usually 256). Because of the regularity of pixel distribution, the information entropy value of plaintext image is usually low. While the ideal encrypted image should present very high random characteristics and its entropy value should approach the theoretical maximum entropy value of 8. The experimental results in Table 2 show that the entropy value of the encryption results of all the test images is close to the theoretical maximum value. Comparing with the existing encryption techniques through Table 3, the entropy value achieved by this algorithm on the Peppers image is significantly better than that of the other encryption algorithms and the pixel distribution of the ciphertext image presents a higher degree of disorder. This indicates that this encryption system further improves the security of the encryption system by enhancing the complexity of the pixel arrangement.

Table 2. Information entropy analysis of four images.

Images	Plaintexts	Ciphertexts
Baboon	7.3674	7.9996
House	7.3254	7.9992
Peppers	7.5836	7.9995
Plane	7.0134	7.9993

Table 3. Information entropy comparison of various algorithms on the Peppers image.

Encryption Algorithms	Proposed	Ref[22]	Ref[23]	Ref[24]	Ref[25]	Ref[26]
Information entropy	7.9995	7.9993	7.9994	7.9987	7.9982	7.9860

5.3 Correlation analysis

The neighboring pixel correlation of an image is one of the core metrics for assessing its statistical properties. In plaintext images, neighboring pixels usually exhibit high correlation with each other, especially in the vertical, horizontal and diagonal directions. This high degree of correlation makes plaintext images vulnerable to statistical attacks, as an attacker can infer the content and features in an image by analyzing the relationships between adjacent pixels. Correlation is computed in the

following way:

$$H(p)=\frac{1}{n}\sum_{i=1}^{N}p_i \tag{22}$$

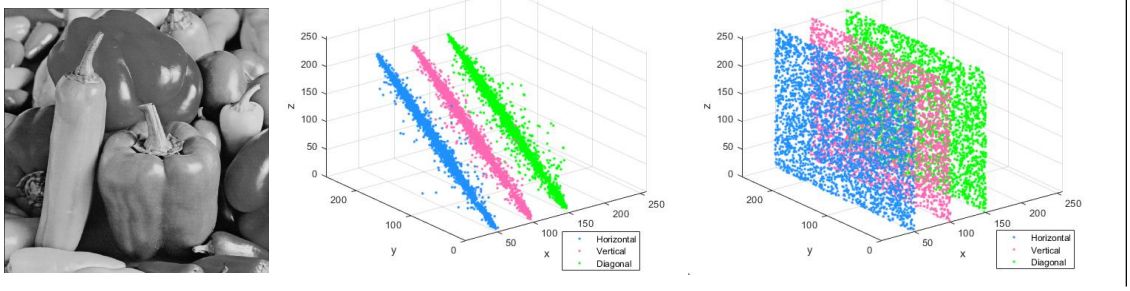
$$G(p)=\frac{1}{n}\sum_{i=1}^{n}\left[p_i-H(p_i)\right]^2 \tag{23}$$

$$\mathrm{cov}(p,q)=\frac{1}{n}\sum_{i=1}^{n}\left[p_i-H(p_i)\right]\left[q_i-H(q_i)\right] \tag{24}$$

$$r_{pq}=\frac{\mathrm{cov}(p,q)}{\sqrt{D(p)D(q)}} \tag{25}$$

Here, p and q are used to represent the pixel values of neighboring pixel points within the image. The variable n signifies the overall quantity of pixels that have been chosen from the image. $H(p)$ specifically indicates the expectation of the gray value, while $D(p)$ stands for the variance of the gray value. Moreover, cov(p,q) represents the covariance between the relevant pixel values.

In the correlation analysis experiment, we calculated the correlation coefficients of pixels in different directions. It can be clearly seen from Fig. *15* that the gray values of the plaintext images all exhibit a very clear clustered distribution, which indicates that there is a strong correlation between neighboring pixels. On the other hand, the pixel distribution of the ciphertext image tends to be homogenized, indicating that the encryption method effectively eliminates the correlation properties of the original image. As shown in Table 4, the correlation coefficients of the plaintext images all exhibit a high degree of linear correlation. And the correlation coefficients of the encrypted images are close to 0. From Table 5 we can see that the correlation coefficients of the present algorithm are significantly lower than that of the existing encryption techniques in the three directions of the pepper image, which suggests that this encryption algorithm achieves better pixel decorrelation effect, thus effectively resisting statistical attacks based on spatial correlation.



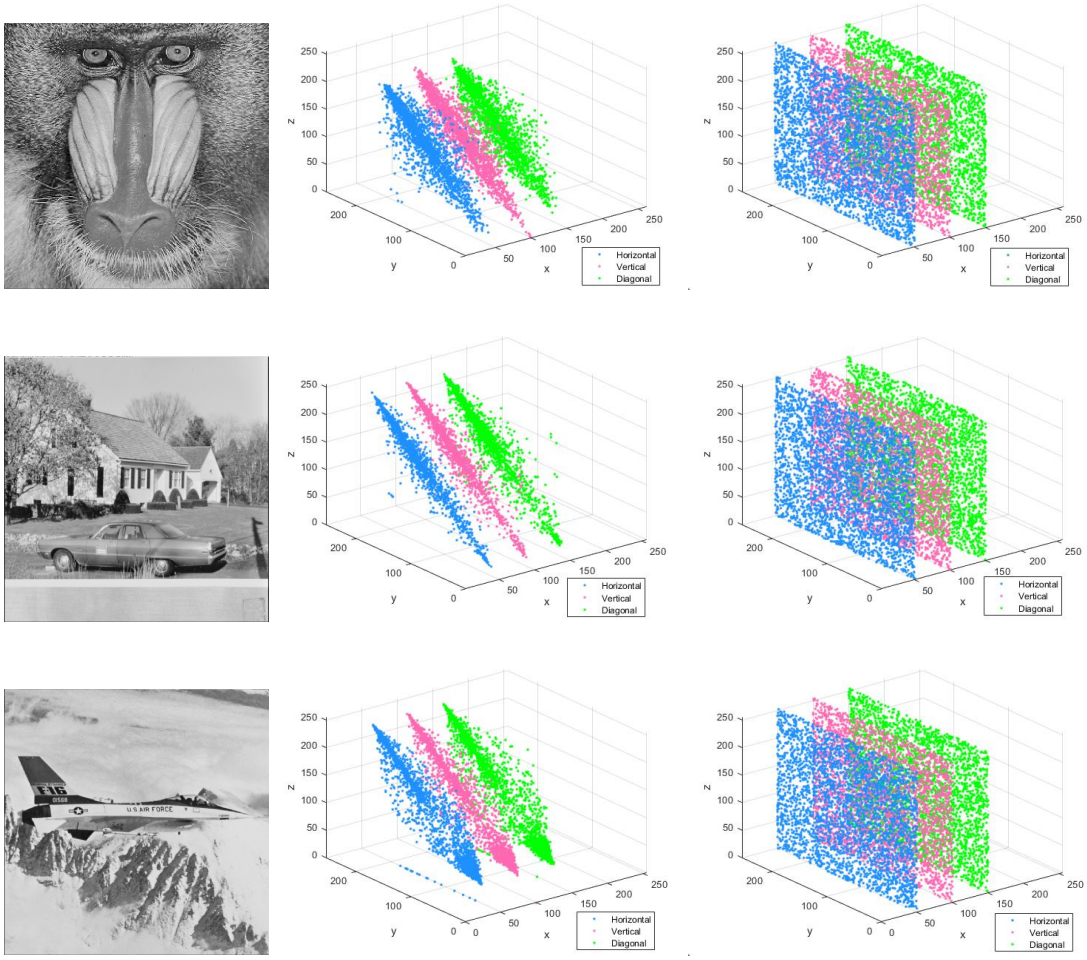


Fig.15 Correlation analysis results of plaintext images and ciphertext images

Table 4. Correlation analysis of four samples.

	Plaintexts			Ciphertexts		
Images	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Peppers	0.9847	0.8745	0.9456	-0.0017	-0.0036	-0.0023
Baboon	0.8976	0.8729	0.9264	0.0018	-0.0003	0.0027
House	0.9463	0.8956	0.9543	0.0027	0.0015	-0.0032
Plane	0.9356	0.9457	0.9245	-0.0019	-0.0016	-0.0009

Table 5. Comparison of correlation coefficients with other algorithms.

Algorithms	Horizontal	Vertical	Diagonal
Original image	0.9847	0.8745	0.9456
Proposed scheme	-0.0017	-0.0036	-0.0023
Ref[27]	-0.0033	0.0011	0.0070
Ref[28]	-0.0052	0.0086	-0.0020
Ref[29]	0.0001	0.0015	0.0078
Ref[30]	0.0131	0.0022	0.0030
Ref[31]	0.0052	0.0039	0.0215

5.4 Key space analysis

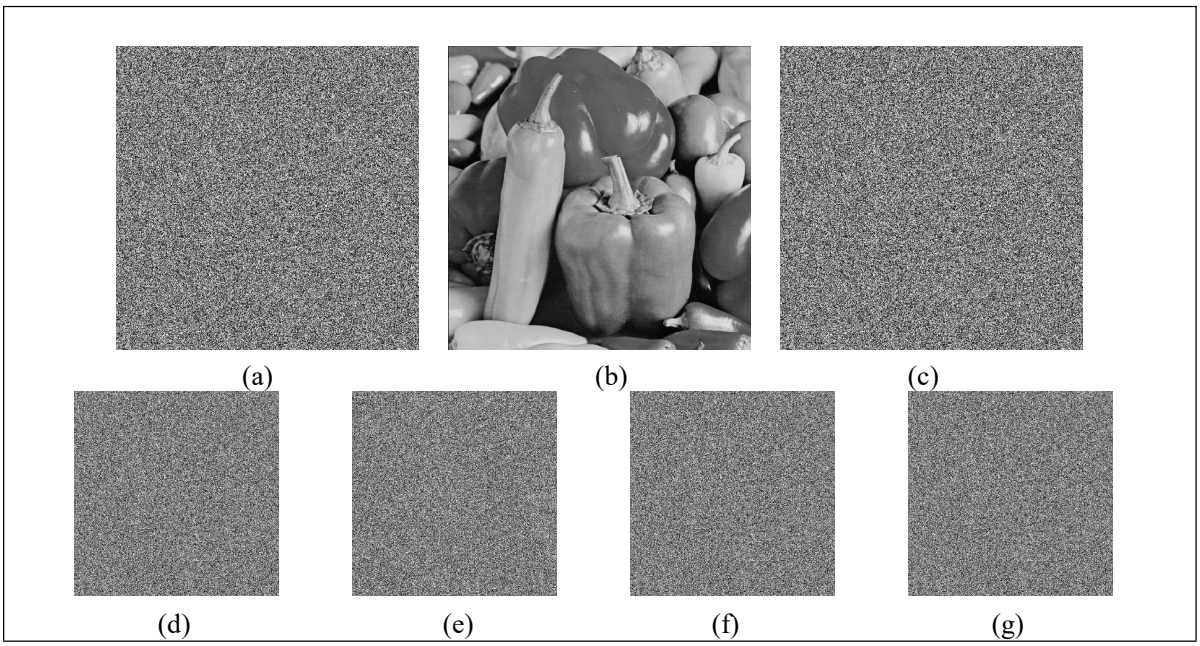
The key space, as a core metric for evaluating the security of an encryption algorithm, refers to the set consisting of all legitimate keys. Using the system parameters and initial values of the multi-wing hyperchaotic system as the encryption key and assuming the minimum precision of the computer is 10^{-14}, the key space of the algorithm is thus calculated to be: $(10^{14})^9 \approx 2^{418}$, which exceeds the minimum standard of 2^{100} recommended in cryptographic theory, demonstrating a very high level of resistance to brute force attacks. The comparative results in Table 6 further emphasize the combined advantages of the algorithm in terms of key space.

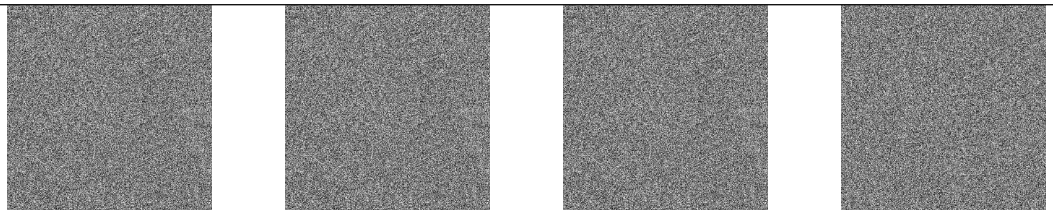
Table 6. Key space size comparison

algorithms	Proposed	Ref[32]	Ref[33]	Ref[34]	Ref[35]	Ref[36]
Key space	2^{418}	2^{232}	2^{194}	2^{196}	2^{260}	2^{116}

5.5 Key sensitivity analysis

An efficient and secure encryption algorithm not only needs to have a large key space to resist brute-force attacks but must also exhibit extremely high sensitivity to key changes. Even if an extremely small change occurs in a parameter of the key, the result of decrypting should be completely wrong and the original image cannot be recovered. This key sensitivity can effectively prevent attackers from carrying out trial attacks through similar keys, thus further improving the security of the algorithm. As verified by the experimental data in Fig.16, the proposed encryption algorithm exhibits extremely high key sensitivity, which can effectively avoid correct decryption in case of small changes in the key parameters. This high sensitivity not only enhances the algorithm's ability to resist trial attack, but also ensures the irreversibility of the ciphertext image, thus significantly improving the overall security of the system.





(h) (i) (j) (k)

Fig.16 The decryption diagram obtained by decrypting ciphertext image 'Peppers' after changing different key values. (a) ciphertext image;(b) decrypted image with correct keys;(c)$x_1(0)=x_1+10^{-14}$;(d) $y_1(0)=y_1+10^{-14}$; (e)$z_1(0)=z_1+10^{-14}$; (f)$w_1(0)=w_1+10^{-14}$; (g)$a=10+10^{-14}$; (h)$b=8/3+10^{-14}$; (i)$c=28+10^{-14}$; (j)k=19.9+10^{-14}; (k)$h=4.7+10^{-14}$

5.6 Differential Attack Analysis

Differential attacks are used to quantitatively assess the sensitivity of an encryption system to input changes by applying subtle alterations to the original plaintext image, and comparing and analyzing the difference features corresponding to the original and the perturbed plaintext image. Differential attack is an important security analysis tool to visually assess the sensitivity of the encryption algorithm to the plaintext image through two key metrics, NPCR and UACI. The calculations are shown in equation (26) and (27).

$$NPCR = \frac{1}{K \times L} \sum_{p=1}^{K} \sum_{q=1}^{L} G(p,q) \times 100\% \tag{26}$$

$$UACI = \frac{1}{K \times L} \left\{ \sum_{p=1}^{K} \sum_{q=1}^{L} \frac{H_1(p,q) - H_2(p,q)}{255} \right\} \times 100\% \tag{27}$$

In this expression, K and L represent the width and height of the pixel matrix, respectively, and G refers to the gray value of the pixel. H_1 is the ciphertext matrix formed by encrypting the original image, while H_2 is the ciphertext matrix generated by changing the value of a pixel in the plaintext. For a pixel at coordinate position (p,q), if $H_1(p,q)$ is not equal to $H_2(p,q)$, then $H(p,q)$ takes the value of 1; conversely, $G(p,q)$ takes the value of 0. If the proposed encryption algorithm in the experiments exhibits high NPCR value (nearly 99.6094%) and high UACI value (nearly 33.4635%), it indicates that the algorithm has a good resistance to differential attacks. Table 7 records the NPCR and UACR values. It can be noticed that the values of NPCR and UACI almost meet the expected criteria. In addition, Table 8 shows the values of this algorithm and other algorithms doing comparison. By comparing these results, it can be clearly concluded that our proposed encryption algorithm performs very well against differential attacks.

Table7. NPCR and UACI analysis across various images.

Image	NPCR (%)	UACI (%)
Baboon	99.61	33.39
Peppers	99.62	33.44
Plane	99.61	33.51
House	99.59	33.46

Table 8. The NPCR and UACI comparison of various algorithms on the Peppers image

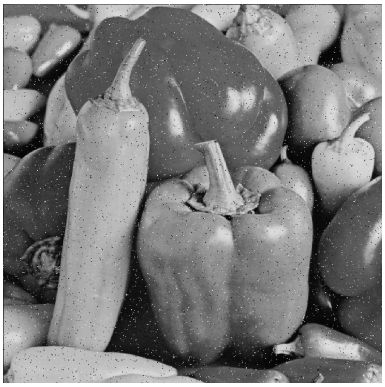
Algorithms	Proposed	Ref[37]	Ref[38]	Ref[39]	Ref[40]	Ref[41]
NPCR (%)	99.62	99.63	99.85	99.61	99.61	99.60
UACI (%)	33.44	33.30	33.59	32.68	33.48	33.52

5.7 Robustness Analysis

Robust analysis aims to investigate whether the decryption algorithm can still effectively restore the key information of the original image when the ciphertext image encounters various kinds of external disturbances, such as noise pollution, cropping attack, pixel tampering and so on. This property is crucial for data security in practical applications, especially in unreliable transmission environments or long-term storage conditions.

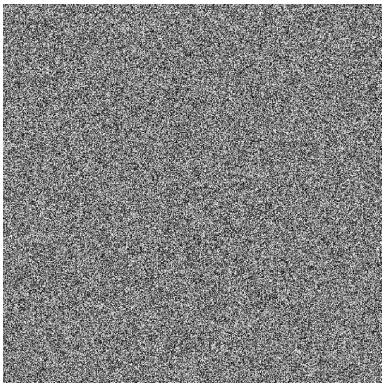
In order to evaluate the performance of the encryption algorithm against noise interference, salt and pepper noise interference of different intensities is introduced to the ciphertext image in the experiment, and the noise levels are set to be 0.5%, 5% and 10%. For each noise level, the decryption operation is performed on the interfered ciphertext image, and the degree of information retention is evaluated by quantitative analysis and visual quality of the decrypted image. The experimental results (shown in Fig.17, 18) show that the encryption scheme exhibits significant resistance to noise interference at different noise levels. Even if the ciphertext image is contaminated by high-intensity noise, the decrypted image can still restore the content of the original image in a basically complete way, which verifies the robustness of the algorithm in noise attack scenarios.

0.5% SPN decrypted image



(a)

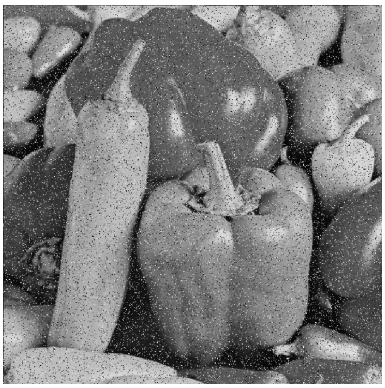
0.5% SPN encrypted image

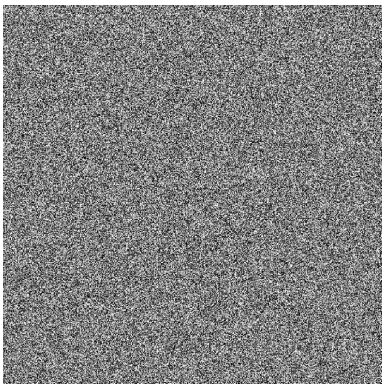


(d)

5% SPN decrypted image

5% SPN encrypted image



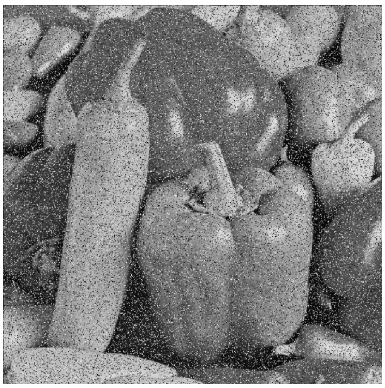


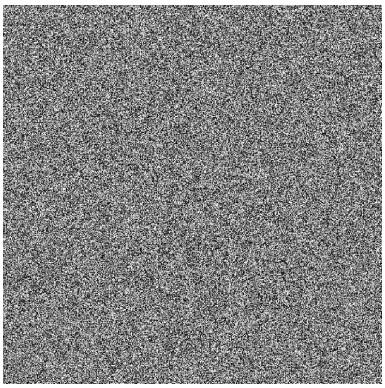
(b)

(e)

10% SPN decrypted image

10% SPN encrypted image





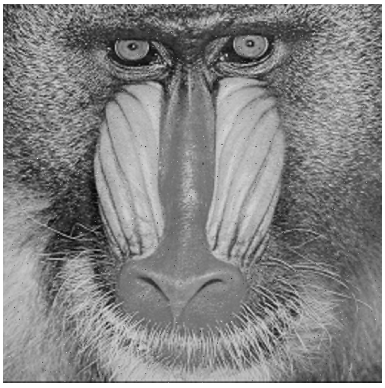
(c)

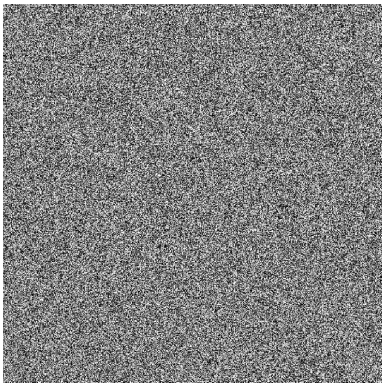
(f)

Fig.17 Different levels of noise attacks.(a)-(C) Ciphertext under different noise attacks; (d)-(f)the corresponding decrypted images

0.5% SPN decrypted image

0.5% SPN encrypted image



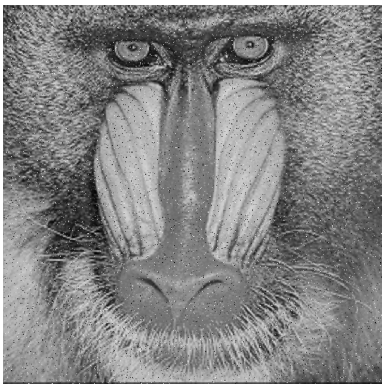


(a)

(d)

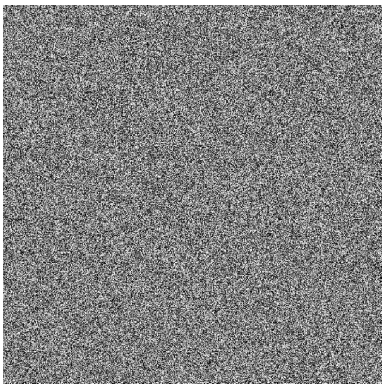
5% SPN decrypted image

5% SPN encrypted image



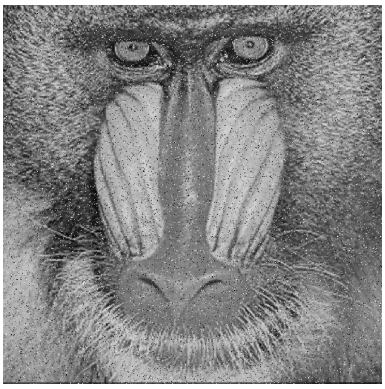
(b)

10% SPN decrypted image

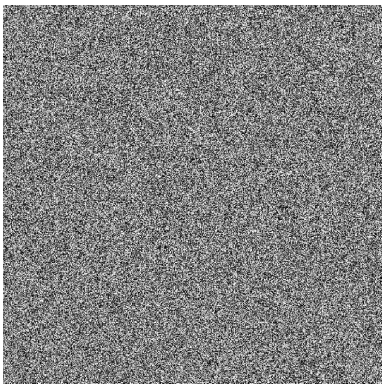


(e)

10% SPN encrypted image



(c)

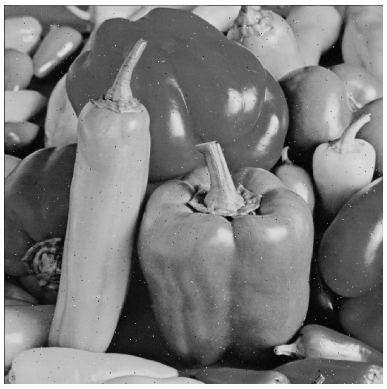


(f)

Fig.18 Different levels of noise attacks.(a)-(c) Ciphertext under different noise attacks; (d)-(f)the corresponding decrypted images

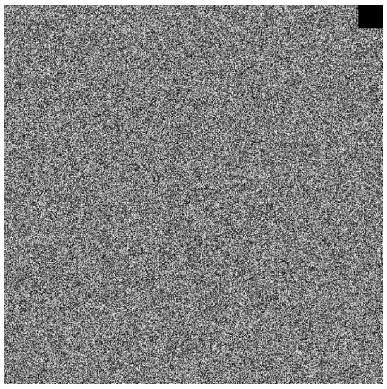
Cropping attacks are another common form of channel interference. During data transmission, the ciphertext image may suffer from partial pixel data loss (cropping) due to channel failures, data packet loss, or other man-made interference. The robustness analysis evaluates the decryption effectiveness of the encryption algorithm in the case of missing pixels by applying a local cropping operation to the ciphertext image. On this basis, we further tested the algorithm in a targeted manner, and the relevant experimental results are visualized by the simulated data Fig.19 and Fig.20. The results of the experiments show that the encryption scheme exhibits significant robustness to cropping attack scenarios by cropping the ciphertext image at multiple levels and analyzing the results. This robustness makes the algorithm more practical in unreliable transmission environments.

Cut 32*32 Encrypted image



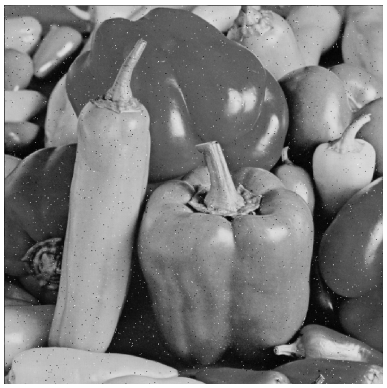
(a)

Cut 32*32 Decrypted image



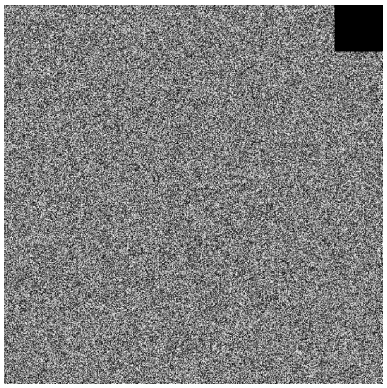
(d)

Cut 64*64 Encrypted image



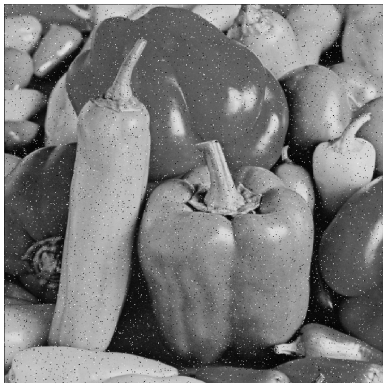
(b)

Cut 64*64 Decrypted image



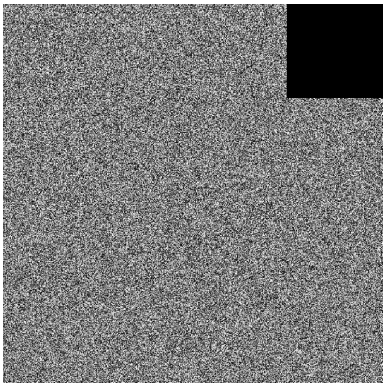
(e)

Cut 128*128 Encrypted image



(c)

Cut 128*128 Decrypted image



(f)

Fig.19 Cropping attacks on Peppers; (a)-(c) Ciphertext subjected to varying degrees of cropping; (d)-(f) the corresponding decrypted images

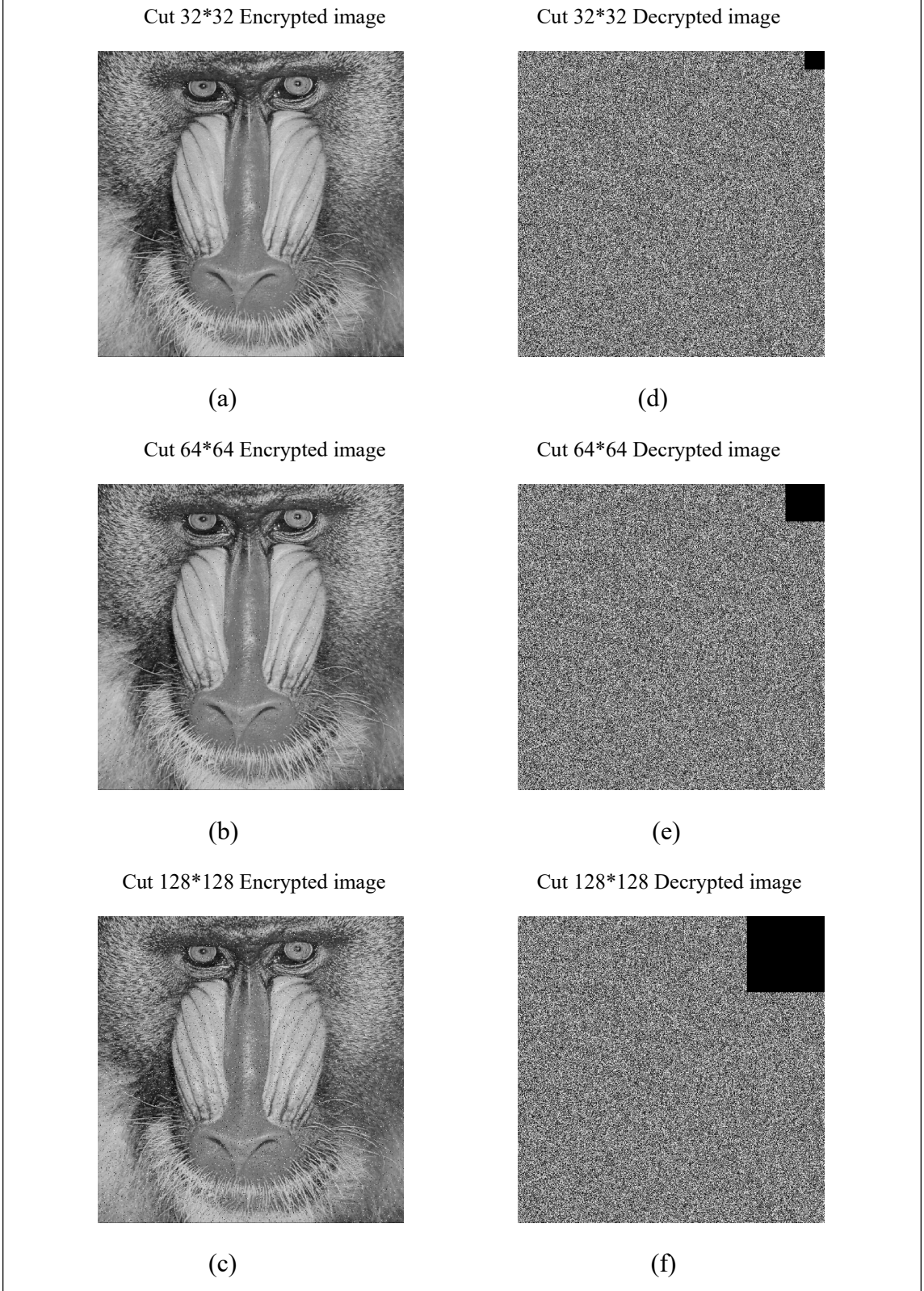


Fig.20 Cropping attacks on Baboon; (a)-(c) Ciphertext subjected to varying degrees of cropping; (d)-(f) the corresponding decrypted images

To show the fault tolerance more clearly, Using equations (28) and (30), the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) are

calculated between the original plaintext image and the resulting image after decryption of the ciphertext image with different levels of attacks, respectively.

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right) \tag{28}$$

$$MSE = \frac{\sum_{x=1}^{m}\sum_{y=1}^{n}\left[q_1(x,y) - q_2(x,y)\right]^2}{m \times n} \tag{29}$$

$$SSIM = \frac{\left(2\overline{q}_1\overline{q}_2 + \delta_1\right)\left(2\sigma_{q_1}\sigma_{q_2} + \delta_2\right)}{\left(\overline{q}_1^2 + \overline{q}_2^2 + \delta_1\right)\left(\sigma_{q_1}^2 + \sigma_{q_2}^2 + \delta_2\right)} \tag{30}$$

In equation (29), where m is the height of the original plaintext image and n is the width of the original plaintext image. $q_1(x, y)$ refers to the gray intensity value at the position of point (x, y) in the original image, and $q_2(x, y)$ denotes the gray intensity value at the position of the corresponding point (x, y) in the decrypted image. In equation (30), $\delta_2 = (\mu_1 t)$, $\delta_2 = (\mu_2 t)$, $\mu_1 = 0.01$, $\mu_2 = 0.03$ $t = 255$. And $\sigma_{q_1}^2$ and $\sigma_{q_2}^2$ represent the variances of q_1 and q_2 .

Table 9 . SSIM between the plain images and the decrypted images under the SPN and cropping attacks.

Test image	SPN level	Proposed	Cropping size	Proposed
Baboon	0.5%	0.9357	32*32	0.9459
	5%	0.8265	64*64	0.6842
	10%	0.5287	128*128	0.4872
Peppers	0.5%	0.9281	32*32	0.8625
	5%	0.7281	64*64	0.3829
	10%	0.3871	128*128	0.2872
House	0.5%	0.8723	32*32	0.8271
	5%	0.6382	64*64	0.3028
	10%	0.2829	128*128	0.1982
Plane	0.5%	0.9018	32*32	0.8723
	5%	0.6235	64*64	0.4827
	10%	0.2587	128*128	0.2745

Table 10 . PSNR between the plain images and the decrypted images under the Nosie and Cropping attacks.

Test image	SPN level	Proposed	Cropping size	Proposed
Baboon	0.5%	32.46	32*32	33.79
	5%	22.57	64*64	27.61
	10%	19.78	128*128	21.62
Peppers	0.5%	31.89	32*32	32.71
	5%	21.68	64*64	26.82
	10%	18.79	128*128	20.72

	0.5%	31.36	32*32	31.89
House	5%	21.46	64*64	25.89
	10%	18.36	128*128	20.34
	0.5%	31.49	32*32	32.28
Plane	5%	21.22	64*64	26.31
	10%	18.26	128*128	20.35

We carried out simulation experiments on four different sets of image samples, in which the experimental data about the noise attack scenario is presented in Table 9, while Table 10 shows the experimental data about the cropping attack scenario. After experimental verification and quantitative analysis, the encryption scheme shows stable decryption performance under different types of attacks, and its anti-differential analysis ability reaches the expected design index.

5.8 Encryption and decryption efficiency evaluation

Encryption and decryption time is a key indicator of the efficiency and complexity of image encryption algorithms. In practical applications, especially in real-time image encryption, video encryption and large-scale data processing scenarios, the image encryption and decryption speed is directly related to whether the algorithm can work or not. An efficient encryption algorithm should be able to complete the encryption and decryption operations wtih shortest possible time to meet the practical requirements. Table 11 presents the time consumed in encryption and decryption process for different images. Table 12 compares the time complexity of this algorithm with other algorithms, further verifying the good properties of the algorithm.

The data and results obtained from the tests clearly show that the proposed image encryption scheme has excellent performance in terms of encryption and decryption speed, and its time complexity meets the requirements of real-time application scenarios, and it is well suited for real-time image processing and large-scale image encryption tasks due to its fast execution time and good scalability. By further optimizing parallelization and hardware implementation, the algorithm is expected to demonstrate enhanced performance and applicability in a wider range of practical application scenarios.

Table 11 . Encryption and decryption speed

Image	Encryption time (s)	Decryption time (s)
Baboon	1.3384	1.4318
Peppers	1.3827	1.4327
Plane	1.4198	1.4873
House	1.3274	1.3972

Table 12 . Runing time of image Peppers compared to different algorithms.

Algorithm	Encryption time (s)	Decryption time (s)
Proposed	1.38	1.43
Ref[42]	1.65	1.63
Ref[43]	0.93	10.58
Ref[44]	5.16	3.10
Ref[45]	16.43	16.43

6 Conclusions

In this paper, an image encryption algorithm along with a multi-wing hyperchaotic system is proposed and its performance in various aspects is systematically analyzed. By introducing the newly designed segmented linear function into the Lorenz system, a multi-wing hyperchaotic system with high complexity and randomness is successfully constructed. Experiments show that the system has good dynamic properties and provides strong application potential for image encryption. The proposed algorithm achieves efficient image encryption through the steps of bit plane decomposition, Zigzag disruption, forward diffusion and backward diffusion, which, combined with the designed key generation mechanism, ensures the high randomness and security of the algorithm. In summary, the image encryption algorithm proposed in this paper shows excellent performance in security, robustness and efficiency by combining the complexity and randomness of the multi-wing hyperchaotic system, which has strong practical application value.

Competing Interests

The authors declare no conflict of interest.

Data Availability

No new data were created or analyzed in this study.

Authors Contributions

Conceptualization, P.F. Ding; methodology P.F. Ding; software, P.F. Ding and W.W. Hu; validation, P.F. Ding, J.Zhang, J.G.Zhu and W.W. Hu; formal analysis, P.F. Ding, W.W.Hu. and P.H. Geng; writing original draft, P.F. Ding and W.W. Hu; writing-review and editing, W.W. Hu, P.H. Geng J. Zhang and J.G.Zhu. All authors have read and agreed to the published version of the manuscript.

References

1. Zhu, F, F. Wang, and L. Ye, *Artificial switched chaotic system used as transmitter in chaos-based secure communication.* Journal of the Franklin Institute, 2020. **357**(15): p. 10997-11020.
2. Pisarchik, A.N., et al., *Secure chaotic communication based on extreme multistability.* Journal of the Franklin Institute, 2021. **358**(4): p. 2561-2575.
3. Ding, P., K. Li, and Z. Wang, *Generation multi-scroll chaotic attractors using composite sine function and its application in image encryption.* Physica Scripta, 2024. **99**(4).
4. Zehra, A., et al., *Physiological and chaos effect on dynamics of neurological disorder with memory effect of fractional operator: A mathematical study.* Computer Methods and Programs in Biomedicine, 2024. **250**.
5. Kong, X., et al., *Memristor-induced hyperchaos, multiscroll and extreme multistability in fractional-order HNN: Image encryption and FPGA implementation.* Neural Networks, 2024. **171**: p. 85-103.
6. Zhang, Z., et al., *Construction of a family of 5D Hamiltonian conservative hyperchaotic systems with multistability.* Physica A: Statistical Mechanics and its Applications, 2023. **620**.

7. Zhang, X., J. Xu, and A.J. Moshayedi, *Design and FPGA implementation of a hyperchaotic conservative circuit with initial offset-boosting and transient transition behavior based on memcapacitor.* Chaos, Solitons & Fractals, 2024. **179**.
8. Jin, M., K. Sun, and H. Wang, *Hyperchaos, extreme multistability, and hidden attractors in the novel complex nonlinear system and its adaptive hybrid synchronization.* Nonlinear Dynamics, 2022. **110**(4): p. 3853-3867.
9. Liu, S., et al., *Design of a new multi-wing chaotic system and its application in color image encryption.* Optik, 2023. **290**.
10. Toughi, S., M.H. Fathi, and Y.A. Sekhavat, *An image encryption scheme based on elliptic curve pseudo random and Advanced Encryption System.* Signal Processing, 2017. **141**: p. 217-227.
11. Zhang, D., L. Chen, and T. Li, *Hyper-chaotic color image encryption based on 3D orthogonal Latin cubes and RNA diffusion.* Multimedia Tools and Applications, 2023. **83**(2): p. 3473-3496.
12. Lorenz, E.N., *The Mechanics of Vacillation.* Journal of the Atmospheric Sciences, 1963. **20**(5): p. 448-465.
13. Li, T.-Y. and J.A. Yorke, *Period Three Implies Chaos.* The American Mathematical Monthly, 2018. **82**(10): p. 985-992.
14. Xin, J., H. Hu, and J. Zheng, *3D variable-structure chaotic system and its application in color image encryption with new Rubik's Cube-like permutation.* Nonlinear Dynamics, 2023. **111**(8): p. 7859-7882.
15. Chen, G., Y. Mao, and C.K. Chui, *A symmetric image encryption scheme based on 3D chaotic cat maps.* Chaos, Solitons & Fractals, 2004. **21**(3): p. 749-761.
16. Tang, L., et al., *A generalised incomplete no-equilibria transformation method to construct a hidden multi-scroll system with no-equilibrium.* International Journal of Computational Science and Engineering, 2024. **27**(1): p. 57-67.
17. Kaur, M. and V. Kumar, *Efficient image encryption method based on improved Lorenz chaotic system.* Electronics Letters, 2018. **54**(9): p. 562-564.
18. Valandar, M.Y., M.J. Barani, and P. Ayubi, *A fast color image encryption technique based on three dimensional chaotic map.* Optik, 2019. **193**.
19. Lai, Q., et al., *A novel pixel-split image encryption scheme based on 2D Salomon map.* Expert Systems with Applications, 2023. **213**.
20. Zhu, Z.-l., et al., *A chaos-based symmetric image encryption scheme using a bit-level permutation.* Information Sciences, 2011. **181**(6): p. 1171-1186.
21. Ding, P., et al., *Multi-wing chaotic system based on meminductor and its application in image encryption.* Physica Scripta, 2024. **99**(11).
22. Weihao, C., et al., *Image encryption algorithm based on multi-bit superposition and optical chaos.* Physica Scripta, 2023. **98**(7).
23. Ding, P., J. Zhu, and J. Zhang, *A four-dimensional no-equilibrium chaotic system with multi-scroll chaotic hidden attractors and its application in image encryption.* Physica Scripta, 2024. **99**(10).
24. Kumar, S. and D. Sharma, *A chaotic based image encryption scheme using elliptic curve cryptography and genetic algorithm.* Artificial Intelligence Review, 2024. **57**(4).

25. Jiang, X., et al., *Image encryption based on actual chaotic mapping using optical reservoir computing.* Nonlinear Dynamics, 2023. **111**(16): p. 15531-15555.
26. Lei, R. and L. Liu, *A two-dimensional chaotic model and its application in image encryption.* Physica Scripta, 2024. **99**(7).
27. Ullah, A., et al., *An Efficient Lightweight Image Encryption Scheme Using Multichaos.* Security and Communication Networks, 2022. **2022**: p. 1-16.
28. Iqbal, N., et al., *Dynamic 3D scrambled image based RGB image encryption scheme using hyperchaotic system and DNA encoding.* Journal of Information Security and Applications, 2021. **58**.
29. Liang, Q. and C. Zhu, *A new one-dimensional chaotic map for image encryption scheme based on random DNA coding.* Optics & Laser Technology, 2023. **160**.
30. Chai, X., et al., *A novel image encryption scheme based on DNA sequence operations and chaotic systems.* Neural Computing and Applications, 2017. **31**(1): p. 219-237.
31. Zhan, K., et al., *Cross-utilizing hyperchaotic and DNA sequences for image encryption.* Journal of Electronic Imaging, 2017. **26**(1).
32. Ayubi, P., S. Setayeshi, and A.M. Rahmani, *Deterministic chaos game: A new fractal based pseudo-random number generator and its cryptographic application.* Journal of Information Security and Applications, 2020. **52**.
33. Haider, M.I., et al., *Block cipher's nonlinear component design by elliptic curves: an image encryption application.* Multimedia Tools and Applications, 2020. **80**(3): p. 4693-4718.
34. Vidhya, R. and M. Brindha, *A novel approach for Chaotic image Encryption based on block level permutation and bit-wise substitution.* Multimedia Tools and Applications, 2021. **81**(3): p. 3735-3772.
35. Cheng, G., C. Wang, and H. Chen, *A Novel Color Image Encryption Algorithm Based on Hyperchaotic System and Permutation-Diffusion Architecture.* International Journal of Bifurcation and Chaos, 2019. **29**(09).
36. Hosny, K.M., S.T. Kamal, and M.M. Darwish, *A color image encryption technique using block scrambling and chaos.* Multimedia Tools and Applications, 2021. **81**(1): p. 505-525.
37. Zhu, S., et al., *Image Encryption Scheme Based on Newly Designed Chaotic Map and Parallel DNA Coding.* Mathematics, 2023. **11**(1).
38. Wen, H., et al., *Design and Embedded Implementation of Secure Image Encryption Scheme Using DWT and 2D-LASM.* Entropy, 2022. **24**(10).
39. Wang, X., et al., *A new image encryption algorithm based on Latin square matrix.* Nonlinear Dynamics, 2021. **107**(1): p. 1277-1293.
40. He, C., et al., *An algorithm based on 6D fractional order hyperchaotic system and knight tour algorithm to encrypt image.* Physica Scripta, 2024. **99**(5).
41. Zhu, L., et al., *A visually secure image encryption scheme using adaptive-thresholding sparsification compression sensing model and newly-designed memristive chaotic map.* Information Sciences, 2022. **607**: p. 1001-1022.
42. Gabr, M., et al., *Image Encryption via Base-n PRNGs and Parallel Base-n S-Boxes.* IEEE Access, 2023. **11**: p. 85002-85030.
43. Chai, X., et al., *A visually secure image encryption scheme based on compressive sensing.* Signal Processing, 2017. **134**: p. 35-51.

44. *Retracted: IIBE: An Improved Identity-Based Encryption Algorithm for WSN Security.* Security and Communication Networks, 2023. **2023**: p. 1-1.
45. Jithin, K.C. and S. Sankar, *Colour image encryption algorithm combining Arnold map, DNA sequence operation, and a Mandelbrot set.* Journal of Information Security and Applications, 2020. **50**.