Physica Scripta

PAPER

A four-dimensional no-equilibrium chaotic system with multi-scroll chaotic hidden attractors and its application in image encryption

RECEIVED
28 June 2024

REVISED
6 August 2024

ACCEPTED FOR PUBLICATION
21 August 2024

PUBLISHED
4 September 2024

Pengfei Ding*, Jingge Zhu and Juan Zhang

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

Keywords: multi-scroll hidden attractors, nonlinear function, image encryption, scrambling and diffusion

Abstract

In recent years, constructing hidden attractors with multi-scroll has become a key discussion point in the research and application fields of chaos science. In this paper, with the existing four-dimensional (4D) chaotic system as the base, a new four-dimensional chaotic system featuring significant characteristics of multi-scroll hidden attractors is constructed by adding a nonlinear function. Comprehensive studies including theoretical analyses and numerical simulations have been carried out on the dynamic properties of the new chaotic system, and all the results show that this system exhibits extremely complex chaotic behaviours and excellent unpredictability, which has great value in image encryption. Therefore, an image encryption scheme based on the new chaotic system is proposed, which cleverly integrates the new scrambling algorithm based on parity coordinate transformation and the new rotational diffusion algorithm. And the effectiveness of this encryption algorithm has been thoroughly analyzed and tested. The results based on the experiments show that this encryption algorithm exhibits significant advantages in performance, which can greatly enhance the security of images during encryption and transmission.

1. Introduction

The widespread occurrence of chaos in natural science and human society has received extensive attention from researchers [1]. Chaos represents a significant phenomenon within nonlinear dynamical systems [2]. Studying the laws of chaotic motion can guide people to understand chaos, control chaos, and utilize chaos. Chaos is a state in which a deterministic dynamical system exhibits long-term unpredictable and uncertain motion at a given initial state. This unpredictability does not arise from external random disturbances, but from the inherent complexity and sensitivity within the system. This unique property makes chaos show remarkable potential application value in various fields such as image encryption [3, 4], watermarking encryption [5, 6], chaotic synchronization [7], chaotic neural network [8, 9], and secure communication [10–13].

System with hidden attractors is more sensitive to slight changes in its initial value. This sensitivity gives the system higher complexity and unpredictability. The attraction domain of the hidden attractor does not intersect with the equilibrium point domain [14], which makes it difficult to locate and reconstruct in phase space. This property provides a strong guarantee for confidential communication and chaotic encryption, which makes data encrypted using the hidden attractors chaotic system almost unbreakable.

Multi-scroll chaotic systems have a larger key space and better unpredictability than single-scroll or double-scroll chaotic systems [15]. Therefore, the intensive study of multi-scroll chaotic systems has received extensive attention from scholars. Lü *et al* constructed chaotic attractors of one-dimension n-scroll, two-dimension $n \times m$-scroll, and three-dimension $n \times m \times l$-scroll via the proposed a saturated function series [16]. Wang *et al* proposed a higher-order Chua circuit capable of generating multi-direction multi-scroll chaotic attractors by introducing RC structures and nonlinear functions [17]. Kuate *et al* established a new Rössler chaotic system based on the nonlinear properties of Chua's diode, and investigated the coexistence characteristics and

amplitude control of multi-scroll attractors [18]. Wang *et al* presented a chaotic system capable of generating multi-scroll chaotic attractors with a saddle-shaped arrangement, which was implemented in hardware by a field-programmable gate array [19]. He *et al* designed a method to generate multi-scroll and multi-wing chaotic attractors based on double-scroll and double-wing chaotic systems by introducing a nested COS-PWL nonlinear function [20]. Lin *et al* proposed a universal design approach using the universal variable extension method to construct different multi-scroll/wing chaotic systems that can generate one-dimension and two-dimension multi-scroll/wing chaotic attractors [21].

Multi-scroll chaotic systems are capable of generating chaotic signals with highly complex dynamics, which gives them greater potential and advantages in several application scenarios. In recent years, chaotic systems with multi-scroll hidden attractors have garnered a great deal of interest and study. Jafari *et al* successfully constructed a multi-scroll chaotic system without equilibrium points by incorporating a sinusoidal function into Sprott A system [22]. Hu *et al* applied a novel nonlinear function to the improved Sprott A system to construct a hidden attractor chaotic system without equilibrium point, and its number of scrolls is controllable [23]. Escalante-González *et al* constructed a multi-scroll chaotic system without equilibrium point by using a multi-segmented linear function, and investigated its mechanism of generating multi-scroll chaotic attractors [24]. Zhang *et al* improved the Jerk system and proposed a multi-scroll hyperchaotic system, which has infinite equilibrium points [25]. Dolvis *et al* proposed a four-scroll hyperchaotic system with no equilibrium points, and analyzed the phenomena of multiple period bifurcations and multistability, which provided insights into the understanding of the dynamic properties of hyperchaos [26]. Tang *et al* designed a general incomplete no-equilibria transformation approach to construct a multi-scroll chaotic system without equilibrium point [27]. Zhang *et al* established a five-dimensional hidden grid multi-scroll chaotic system by introducing two multi-stable memristors to the Sprott A system, which has the property that its scroll number can be modulated [28].

To protect the security of digital images, researchers have proposed various encryption algorithms, which include chaos-based encryption [29–31], bit plane-based encryption [32, 33] and reversible cellular automata algorithms [34]. Among them, chaotic image encryption has received extensive attention from researchers due to the maturity of chaos theory. Cao *et al* employed the proposed 2D infinite collapse map to produce two pseudo-random matrices, which are used to perform two rounds of scrambled and diffused operations to obtain effective image encryption [35]. Li *et al* designed an image encryption algorithm that combined a 6D chaotic system with DNA coding, which scrambled and diffused the image with random chaotic sequences, and encrypting the scrambled and diffused image twice with different chaotic sequences at the DNA level [36]. Wang *et al* used the proposed dynamic coupled map lattices with nonlinear perturbations to diffuse the image first, then scramble the decompositioned bit plane, and finally perform a mutual diffusion operation on the high and low planes to achieve image encryption [37]. Wang *et al* put forward a new chaotic image encryption algorithm by combining the semi-tensor product and the composite key to realize efficient and secure encryption [38]. Lin *et al* designed a novel image encryption algorithm with superior performance based on the proposed grid multi-butterfly memristive Hopfield neural network [9]. Ding *et al* proposed a new image encryption algorithm with good encryption effect by combining a newly proposed four-dimensional multi-scroll chaotic system without equilibrium points and ribonucleic acid coding technology [39].

Inspired by previous outstanding research work, this paper presents an image encryption algorithm based on a new 4D chaotic system. The main contributions of this paper are summarised as follows: (1) A 4D chaotic system with controllable multi-scroll hidden attractors is constructed by applying a nonlinear function [40] into a 4D system, and this chaotic system has a complex dynamics. (2) Based on this new chaotic system, a novel encryption algorithm with the core structure of scrambling and diffusion is proposed, which mainly includes the new scrambling algorithm based on parity coordinate transformation and the new rotational diffusion algorithm. A series of performance tests and evaluations have proven that the encryption algorithm is safe, reliable, highly efficient, not only does it possess a key space that is big enough, but also it's very effective in resisting various attacks.

The subsequent sections of this paper are structured as follows: section 2 introduces a novel 4D chaotic system with multi-scroll hidden attractors, and its dynamical properties are comprehensively analyzed. Section 3 elaborates an image encryption algorithm based on this chaotic system. The evaluation and detailed analysis of the performance of the encryption algorithm are presented in section 4. Section 5 provides a concluding summary.

2. Model of the new chaotic system

A 4D autonomous system model is obtained by applying a state feedback to the modified Sprott B system [41], which can be described as equation (1).

$$\begin{cases} \dot{x}_1 = -x_2 x_3 - d x_4 \\ \dot{x}_2 = x_3^2 - 1 \\ \dot{x}_3 = a x_1 - b x_2 x_3 - x_3 \\ \dot{x}_4 = c x_3 \end{cases} \tag{1}$$

Here x_1, x_2, x_3, and x_4 represent state variables and a, b, c, and d are system parameter. The variable x_2 in the right of equation (1) is substituted with a nonlinear function [40] to obtain the new chaotic system, which is presented as follows:

$$\begin{cases} \dot{x}_1 = -g(x_2) x_3 - d x_4 \\ \dot{x}_2 = x_3^2 - 1 \\ \dot{x}_3 = a x_1 - b g(x_2) x_3 - x_3 \\ \dot{x}_4 = c x_3 \end{cases} \tag{2}$$

In equation (2), g(x_2) is a composite nonlinear function that consists of two key components: a nonlinear component and a linear component. The nonlinear component is the product of a sine function and a sign function, and the linear component is the product of state variable x_2 and a sign function, and $g(x_2)$ is described by equation (3).

$$g(x_2) = M \sin(2\pi k x_2)[\mathrm{sgn}(x_2 - m) - \mathrm{sgn}(x_2 - n)] + x_2[2 - \mathrm{sgn}(x_2 - m) + \mathrm{sgn}(x_2 - n)] \tag{3}$$

3. Dynamic analysis of the new chaotic system

3.1. Equilibrium point analysis

The equilibrium point of the system (2) can be obtained by setting $\dot{x}_1 = 0$, $\dot{x}_2 = 0$, $\dot{x}_3 = 0$, $\dot{x}_4 = 0$ which is:

$$\begin{cases} -g(x_2) x_3 - d x_4 = 0 \\ x_3^2 - 1 = 0 \\ a x_1 - b g(x_2) x_3 - x_3 = 0 \\ c x_3 = 0 \end{cases} \tag{4}$$

Looking at the fourth equation of equation (4), if let $x_3 = 0$, which contradicts the second equation of equation (4). Consequently, there is no equilibrium point in new system (2).

3.2. Phase diagram analysis

3.2.1. Phase diagram analysis of the variation with parameter b

Fix the parameters as $a = 15$, $c = 1$, $d = 0.01$, $k = 0.5$, $m = -3$, $n = 6.8$, $M = 0.4$, and initial conditions as (1, 1, 1, 1), When $b = 12$, the system exhibits 2 double-scroll chaotic attractors as shown in figure 1(a). When $b = 10.5$, the system shows 3 double-scroll chaotic attractors as shown in figure 1(b). When $b = 10.15$ and $b = 11.63$, the system exhibits 4 double-scroll chaotic attractors and 5 double-scroll chaotic attractors respectively, as presented in figures 1(c) and (d). These results demonstrate that the new system can generate different number of double-scroll chaotic attractors.

3.2.2. Phase diagram analysis of the variation with initial condition $x_1(0)$

The state of a chaotic system exhibits pronounced sensitivity to its initial values, no matter how insignificant difference in its initial conditions, this difference will rapidly amplify over time, resulting in significant differences in the subsequent states of the system. When the system parameters of the new chaotic system are selected to be $a = 15$, $b = 10$, $c = 1$, $d = 0.01$, $k = 0.5$, $m = -3$, $n = 6.8$, $M = 0.4$, and initial conditions of ($x_1(0)$, $x_2(0)$, $x_3(0)$, $x_4(0)$) are chosen as (1, 1, 1, 1) and ($1+10^{-15}$, 1, 1, 1), respectively, the system exhibits different multi-scroll chaotic attractors, and they are displayed in figures 2(a) and (b). From figure 2, it is abvious that a small change in $x_1(0)$ leads to different chaotic attractors are generated by the new chaotic system.

3.2.3. Phase diagram analysis of the variation with the simulation time T

With system parameters chosen as $a = 15$, $b = 10$, $c = 1$, $d = 0.01$, $k = 0.5$, $m = -3$, $n = 6.8$, $M = 0.4$, and initial conditions equal to (1, 1, 1, 1), when the simulation time T is set to 100, 200, 400, and 700, respectively, chaotic attractors with 1 double-scroll, 3 double-scroll, 4 double-scroll and 5 double-scroll are produced, as depicted in figure 3. It can be found that the number of multi-scroll attractors increases over time as the simulation time prolongs, reflecting the system's self-oscillating dynamics.

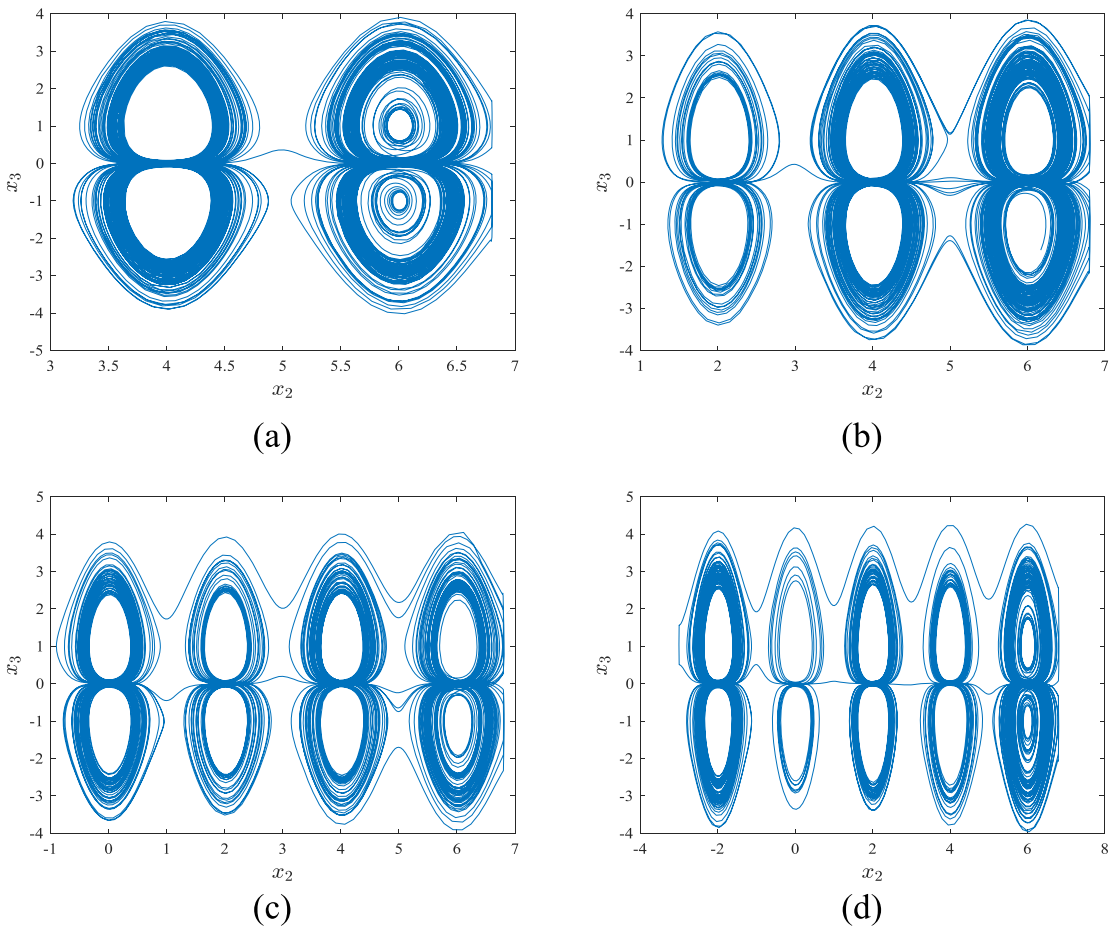

Figure 1. Phase diagrams with $a = 15, c = 1, d = 0.01, k = 0.5, m = -3, n = 6.8, M = 0.4$, and initial conditions of (1, 1, 1, 1). (a) 2 double-scroll chaotic attractors with $b = 12$. (b) 3 double-scroll chaotic attractors with $b = 10.5$. (c) 4 double-scroll chaotic attractors with $b = 10.15$.(d) 5 double-scroll chaotic attractors with $b = 11.63$.

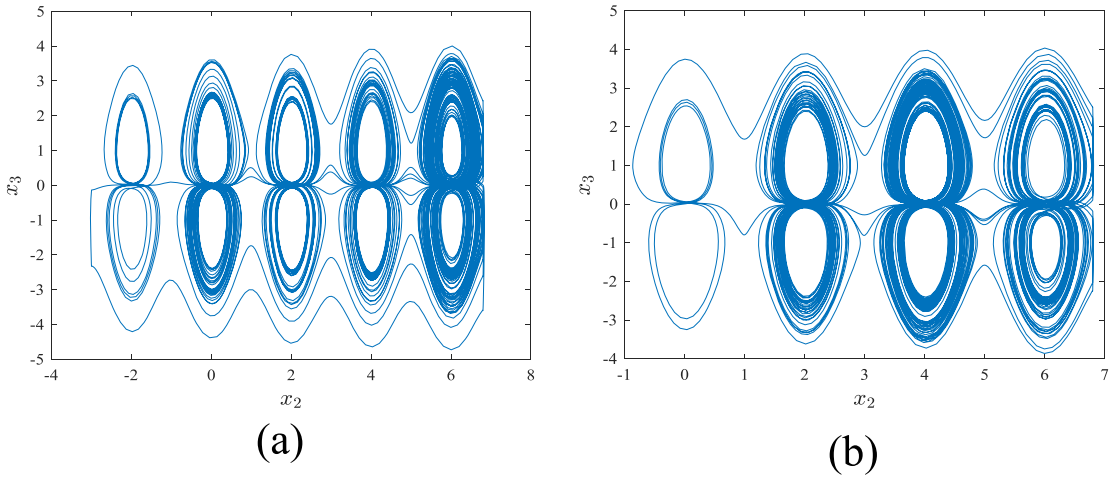

Figure 2. Phase diagrams with $a = 15, b = 10, c = 1, d = 0.01, k = 0.5, m = -3, n = 6.8, M = 0.4$. (a) 5 double-scroll chaotic attractors with initial condition of (1, 1, 1, 1). (b) 4 double-scroll chaotic attractors with initial condition of $(1+10^{-15}, 1, 1, 1)$.

3.2.4. Phase diagram analysis of the variation with the parameter m and n.

Let the system parameters $a = 15, b = 10, c = 1, d = 0.01, k = 0.5, M = 0.4$ and the initial condition equals to (1, 1, 1, 1). The nonlinearity of the novel chaotic system is bounded, and its dynamic range is determined by the system parameters m and n. Therefore, the number of multi-scroll attractors can be varied by selecting different values for the system parameters m and n. The phase planes with various numbers of multi-scroll attractors and

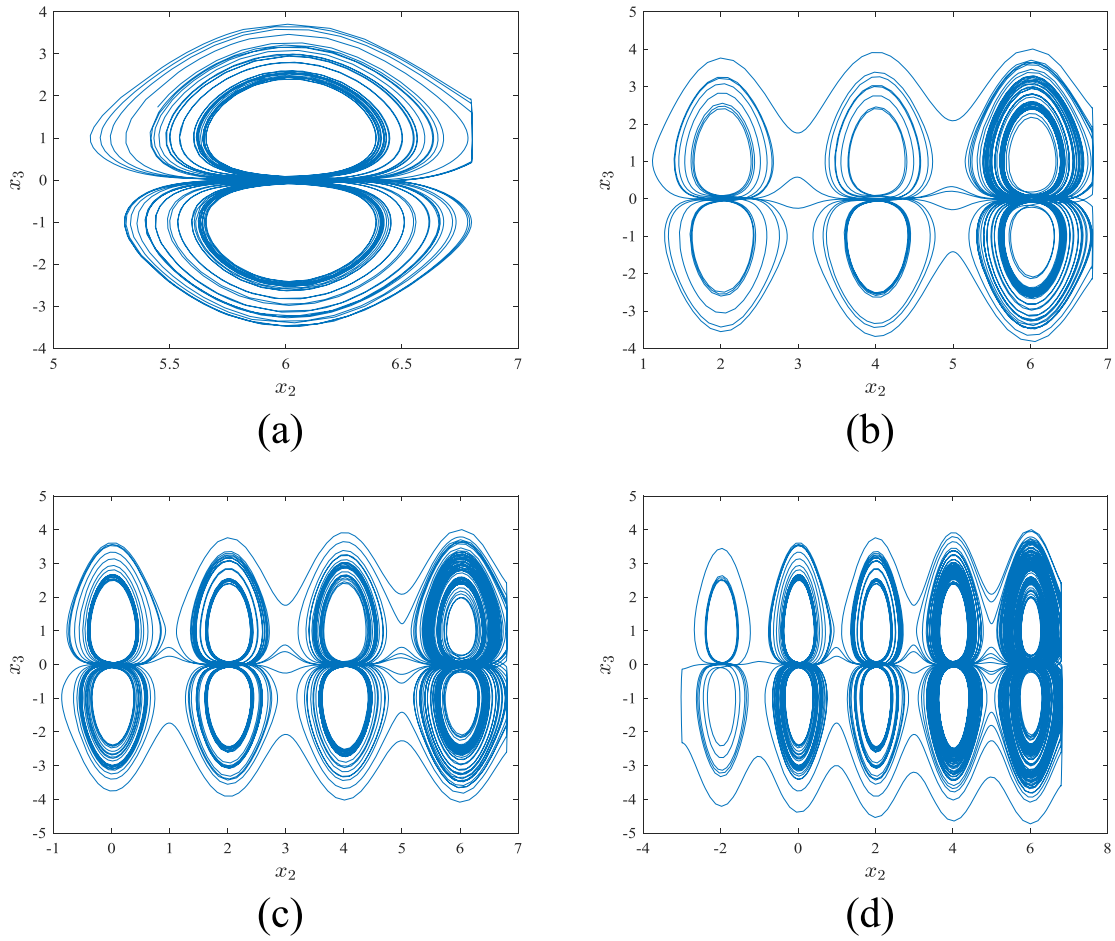

Figure 3. Phase diagrams with $a = 15, b = 10, c = 1, d = 0.01, k = 0.5, m = -3, n = 6.8, M = 0.4$ and initial conditions of (1, 1, 1, 1). (a) 1 double -scroll chaotic attractors with $T = 100$. (b) 3 double -scroll chaotic attractors with T = 200. (c) 4 double -scroll chaotic attractors with $T = 400$. (d) 5 double -scroll chaotic attractors with T = 700.

their corresponding nonlinear functions are given in figure 4. From figures 4(a), (c), (e), (g), it is evident that as the parameters m, n are varied, different multi-scroll chaotic attractors are generated in the x_2 direction and the range is limited between m and n. This is mainly due to the effect of the nonlinear function $g(x_2)$, which allows the new chaotic system to generate distinct multi-scroll chaotic attractors, as shown in figures 4(b), (d), (f), (h).

3.3. Poincaré mapping analysis

Although the phase diagram can qualitatively show the morphology of the trajectory line and its topology, it cannot quantitatively reflect the changing state of physical quantities over time. In order to gain a more comprehensive understanding of the global image of the motion of the dynamical system and determine the morphology of its motion more precisely, the fixed system parameter values $a = 15, b = 10, c = 1, d = 0.01$, $k = 0.5, m = -3, n = 6.8, M = 0.4$, and initial conditions with (1, 1, 1, 1), the Poincaré cross-section in the $x_3 = 0$ plane is selected for analysis, as shown in figure 5. It can be clearly seen Poincaré cross-section is a number of pieces of dense points with fractal structure, demonstrating that the system is in a chaotic state.

3.4. Lyapunov exponent and bifurcation diagram analysis

The system parameters $a = 15, c = 0.1, d = 0.01, k = 0.5, m = -3, n = 6.8, M = 0.4$, set the initial condition as (1, 1, 1, 1), and varying the parameter $b \in [10,\ 12]$, the Lyapunov exponent spectrum and bifurcation diagram of the system's state variable x_2 with respect to the parameter b are shown in figure 6. It can observed that $LE1 > 0$, $LE2 = 0, LE3 < 0$ and $LE4 < 0$ in figure 6(a), which effectively indicates that the system is in a chaotic state. From figure 6(b), it can be observed that the system produces varying numbers of chaotic attractors with the variation of the parameter b, such as 2 double-scroll chaotic attractors, 3 double-scroll chaotic attractors, 4 double-scroll chaotic attractors, 5 double-scroll chaotic attractors. Through the analysis of the Lyapunov exponent spectrum and the bifurcation diagram, it is discovered that bifurcation diagram and the Lyapunov exponent diagram can confirm each other and mutually reveal the dynamic behavioral traits of the system.

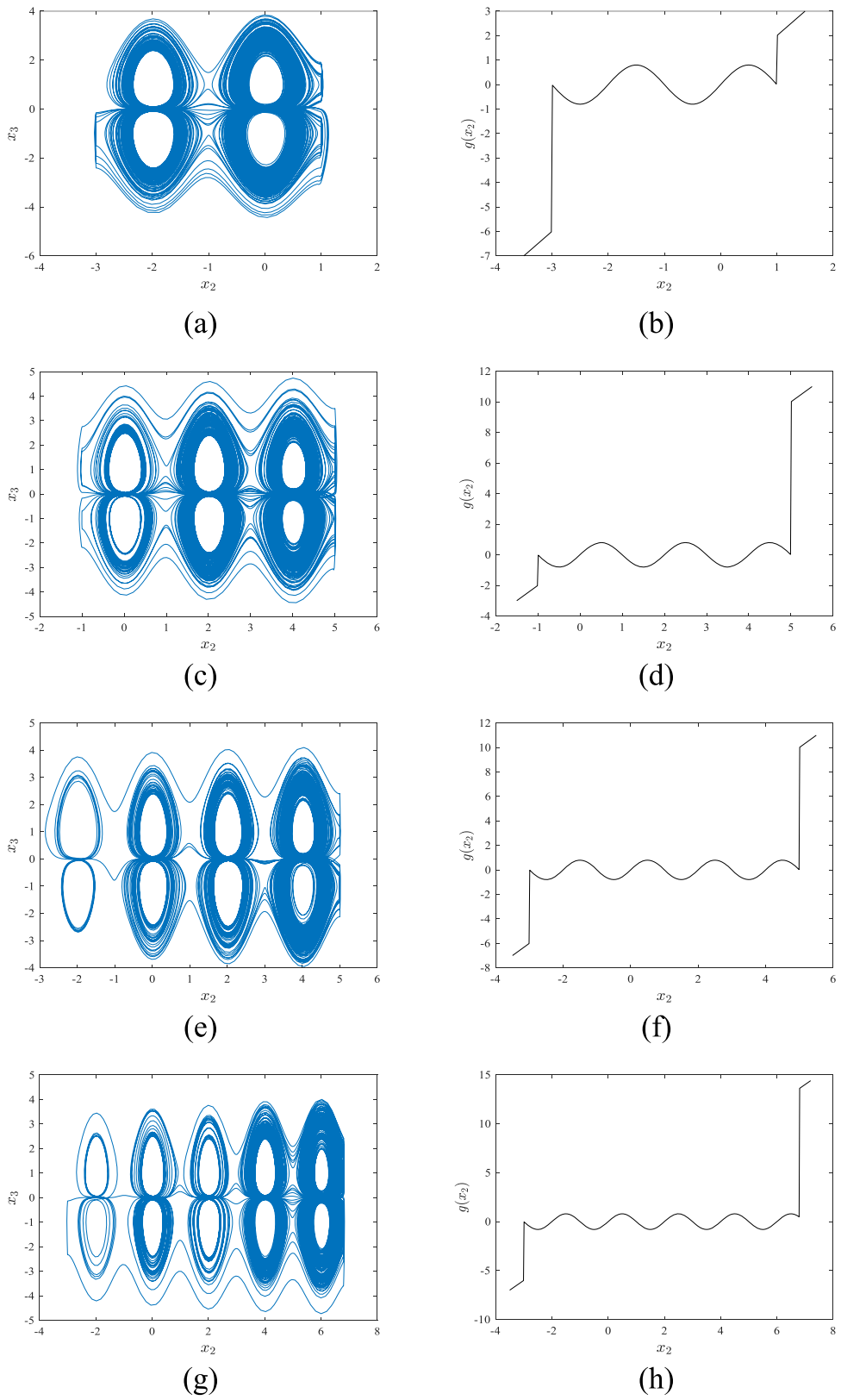

Figure 4. Phase diagrams with $a = 15, b = 10, c = 1, d = 0.01, k = 0.5, M = 0.4$ and initial condition of (1, 1, 1, 1). (a) 1 double-scroll chaotic attractors with $m = -3$ and $n = 1$. (b) nonlinear function with $x_2 \in [-3.5,\ 1.5]$. (c) 3 double-scroll chaotic attractors with $m = -1$ and $n = 5$. (d) nonlinear function with $x_2 \in [-1.5,\ 5.5]$. (e) 4 double-scroll chaotic attractors with $m = -3$ and $n = 5$. (f) nonlinear function with $x_2 \in [-3.5,\ 5.5]$. (g) 5 double-scroll chaotic attractors with $m = -3$ and $n = 6.8$. (h) nonlinear function with $x_2 \in [-3.5,\ 7.2]$.

3.5. 0-1 Test analysis

In order to deeply investigate the chaotic behaviour of the system, a brand new method of the 0-1 test is introduced to the new system. In the 0-1 test, a series of mathematical transformations of the time series are

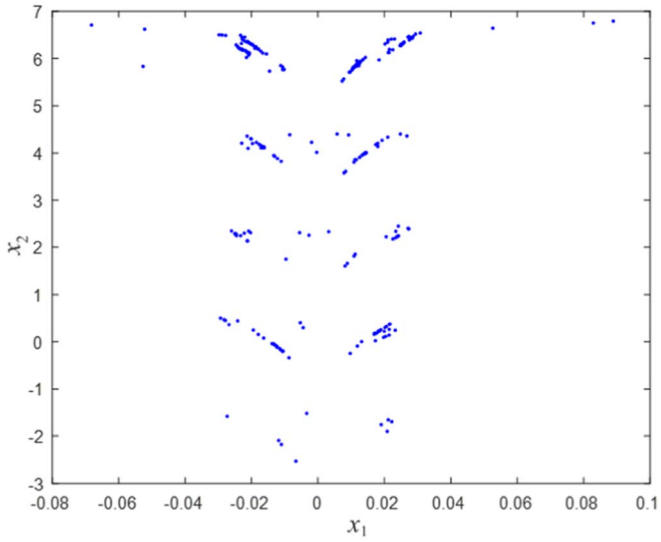

Figure 5. Poincaré mapping with $a = 15, b = 10, c = 1, d = 0.01, k = 0.5, m = -3, n = 6.8, M = 0.4$, and initial conditions of (1, 1, 1, 1).

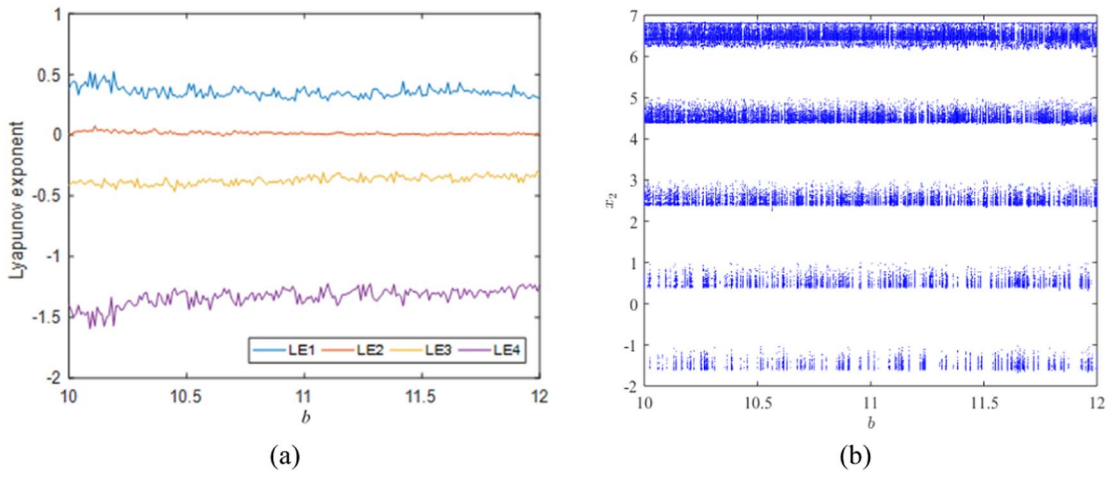

Figure 6. The Lyapunov exponent diagram and the bifurcation diagram with $a = 15, c = 1, d = 0.01, k = 0.5, m = -3, n = 6.8$, $M = 0.4$, and initial conditions of (1, 1, 1, 1). (a) The Lyapunov exponent diagram with $b \in [10, 12]$. (b) The bifurcation diagram with $b \in [10, 12]$.

required to generate *p-s* trajectory plots. The variables $p(m)$ and $s(m)$ represent transformations of the time series of $\varphi(j)$, which are computed according to equation (5).

$$\begin{cases} \varepsilon(k) = k\gamma + \sum_{i=1}^{k} \varphi(j),\ k = 1, 2, \ldots m \\ p(m) = \sum_{k=1}^{m} \varphi(j)\cos(\varepsilon(k)),\ k = 1, 2, \ldots M \\ s(m) = \sum_{k=1}^{m} \varphi(j)\sin(\varepsilon(k)),\ k = 1, 2, \ldots M \end{cases} \tag{5}$$

Where γ is any positive real number between 0 and π. In the case where the time series exhibits non-chaotic behavior, the *p-s* trajectory map does not spread infinitely and is bounded. Conversely, for chaotic time series, the *p-s* trajectory map diffuses continuously and is unbounded, which similar to Brownian motion. When system parameters $a = 15, b = 10, c = 1, d = 0.01, k = 0.5, M = 0.4$ and the initial condition is (1, 1, 1, 1). The results of the 0-1 test for the new system are displayed in figure 7. The *p-s* trajectory diagram exhibits unstable unbounded motion, which indicates the chaotic properties of the system.

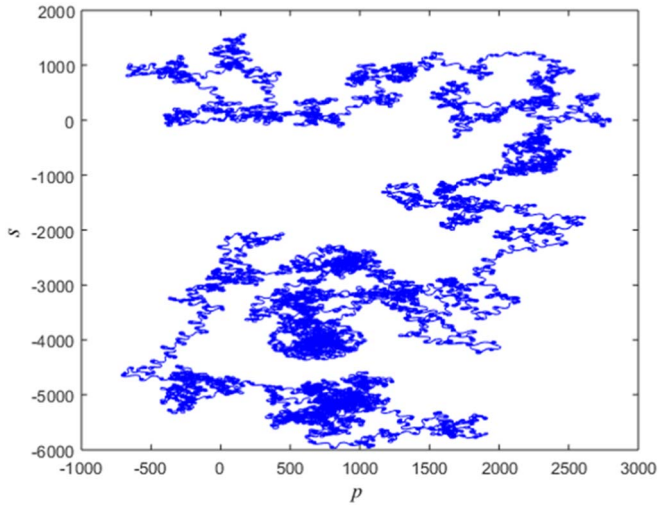

Figure 7. The *p*-*s* trajectory diagram with $a = 15$, $b = 10$, $c = 1$, $d = 0.01$, $k = 0.5$, $m = -3$, $n = 6.8$, $M = 0.4$, and initial conditions of (1, 1, 1, 1).

3.6. Complexity analysis

There is no more fascinating study object in chaotic systems than the chaotic interval. It is important to note that a chaotic system's degree of chaos is positively correlated with its complexity. When the value of complexity rises, the degree of chaos deepens, and the pseudo-randomness of the generated sequences becomes more and more significant, and this strong pseudo-randomness makes the key generated by the system more difficult to be cracked, thus improving the security of the information. Researchers often use C_0-algorithm and Spectral entropy (*SE*) to evaluate the chaotic system complexity. Firstly, $T(l)$ is obtained by the Fourier transform of the time series $t(m)$ of length M, as in equation (6). Secondly, $\tilde{T}(l)$ is obtained by removing the non-regular part, as in equation (7). Then $\tilde{t}(m)$ is obtained from the $\tilde{T}(l)$ Fourier inverse transformation, as in equation (8). Finally C_0-algorithm is obtained from equation (9). P_l is the relative power-spectrum probability of the sequence calculated from equation (10). *SE* is obtained from equation (11).

$$T(l) = \sum_{m=0}^{M} t(m) e^{j\frac{2\pi}{M}ml} \tag{6}$$

$$\tilde{T}(l) = \begin{cases} T(l), \ |T(l)|^2 > \dfrac{1}{M}\displaystyle\sum_{m=0}^{M-1} |T(l)|^2 \\ 0, \ |T(l)|^2 \leqslant \dfrac{1}{M}\displaystyle\sum_{m=0}^{M-1} |T(l)|^2 \end{cases} \tag{7}$$

$$\tilde{t}(m) = \frac{1}{M}\sum_{l=0}^{M-1} \tilde{T}(l) e^{j\frac{2\pi}{M}ml} \tag{8}$$

$$C_0 = \sum_{m=0}^{M-1} |t(m) - \tilde{t}(m)|^2 \Big/ \sum_{m=0}^{M-1} |t(m)|^2 \tag{9}$$

$$P_l = \frac{|T(l)|^2}{\sum_{l=0}^{M/2-1} |T(l)|^2} \tag{10}$$

$$SE = -\sum_{l=0}^{M/2-1} P_l \ln P_l \tag{11}$$

The system parameters are given as $a = 15$, $d = 0.01$, $k = 0.5$, $m = -3$, $n = 6.8$, $M = 0.4$, and initial condition with (1, 1, 1, 1). When $b \in [10,\ 12]$ and $c \in [0,\ 1]$, the C_0-algorithm and *SE* complexity are illustrated in figure 8. It is evident from analyzing complexity value in figure 8 that the system is chaotic. In summary, this chaotic system is suitable for applications in image encryption.

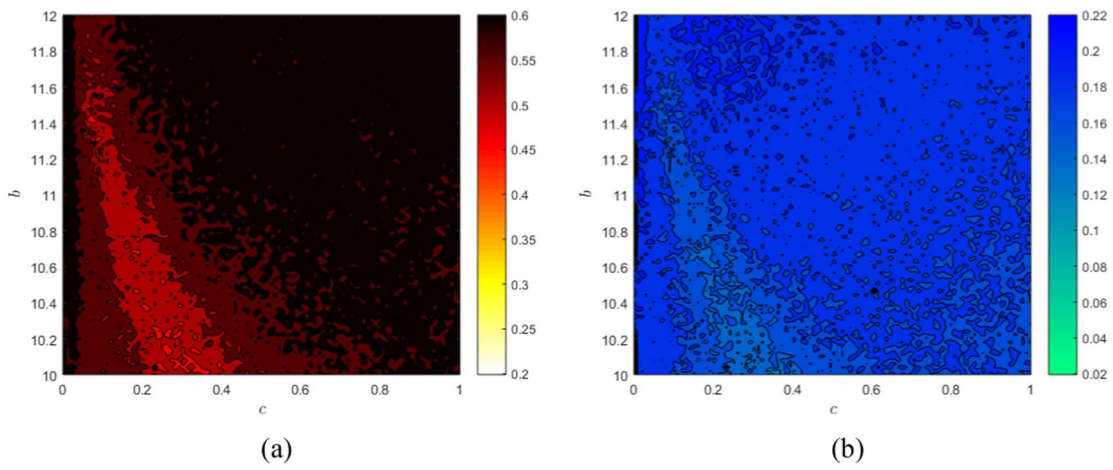

Figure 8. Complexity with $a = 15, d = 0.01, k = 0.5, m = -3, n = 6.8, M = 0.4$, and initial conditions equal to (1, 1, 1, 1). (a) C_0 Complexity with $b \in [10,\ 12]$, $c \in [0,\ 1]$. (b) SE Complexity with $b \in [10,\ 12]$, $c \in [0,\ 1]$.

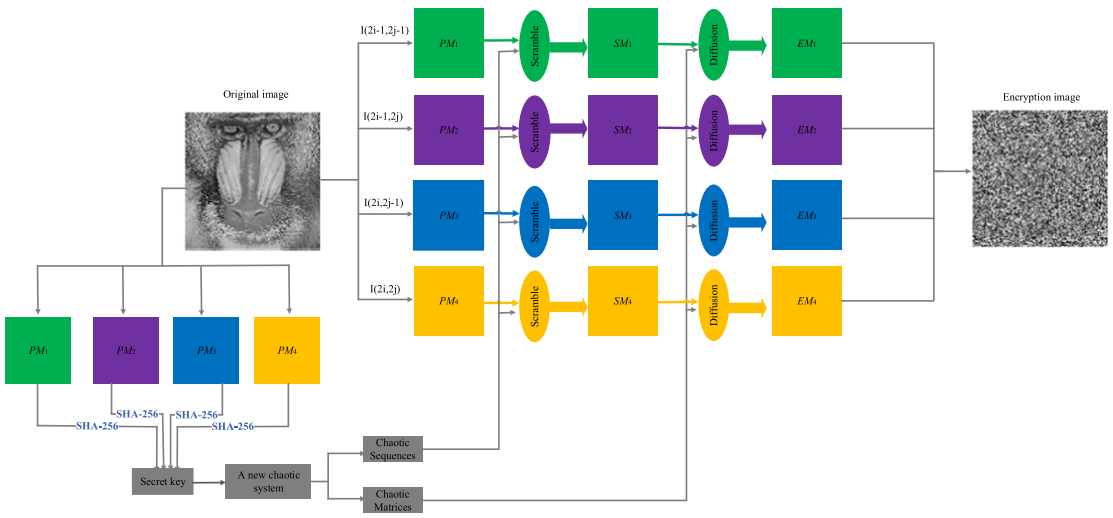

Figure 9. The block diagram of the image encryption algorithm.

4. The proposed algorithm of image encryption

4.1. Image Encryption Algorithm

The comprehensive block diagram of the image encryption algorithm based on the new chaotic system is illustrated in figure 9. Firstly, a grayscale plaintext image of size $m \times n$ is inputted. Subsequently, the initial value and parameter value of the new chaotic system are generated by the secret key to get the chaotic matrices and chaotic sequences. The position of the image pixels is changed by the scrambling algorithm based on the parity coordinate transformation. Following this process, the pixel value is changed by the rotational diffusion algorithm to get the encrypted image at last. The detailed steps are given in the next subsection.

4.1.1. Generation of initial states and parameters

Step1: The original image I with size of $m \times n$ is divided into four matrices PM_1, PM_2, PM_3 and PM_4 of size $1/2m \times 1/2n$ by equation (12).

$$\begin{cases} PM_1 = I(2i-1,\ 2j-1) \\ PM_2 = I(2i-1,\ 2j) \\ PM_3 = I(2i,\ 2j-1) \\ PM_4 = I(2i,\ 2j) \end{cases} \quad 1 \leqslant j \leqslant \frac{1}{2}nj \in Z,\ 1 \leqslant i \leqslant \frac{1}{2}mi \in Z \tag{12}$$

Step2: Use Secure Hash Algorithm (SHA-256) on the four matrices of PM_1, PM_2, PM_3 and PM_4 respectively to get four keys of K_1, K_2, K_3 and K_4 with a length of 256 bits.

$$\begin{cases} K_1 = f_{SHA256}(PM_1) = r_1 r_2 \cdots r_{256} \\ K_2 = f_{SHA256}(PM_2) = s_1 s_2 \cdots s_{256} \\ K_3 = f_{SHA256}(PM_3) = t_1 t_2 \cdots t_{256} \\ K_4 = f_{SHA256}(PM_4) = v_1 v_2 \cdots v_{256} \end{cases} \tag{13}$$

Step3: Generate a new 256-bit K by alternately splicing $r_1 r_5 ... r_{249} r_{253}$ of K_1 , $s_2 s_6 ... s_{250} s_{254}$ of K_2, $t_3 t_7 ... t_{251} t_{255}$ of K_3 and $v_4 v_8 ... v_{252} v_{256}$ of K_4.

$$K = r_1 s_2 t_3 v_4 r_5 s_6 t_7 v_8 \cdots r_{253} s_{254} t_{255} v_{256} \tag{14}$$

Step4: 256-bit hash key K, every 8-bit group to get 32 keys, through the 32 keys, we can obtain the initial value of the new chaotic system $x_1(0), x_2(0), x_3(0), x_4(0)$ and the system parameters a, b, c, d.

$$K = k_1 k_2 k_3 \cdots k_{30} k_{31} k_{32} \tag{15}$$

$$\begin{cases} x_1(0) = ((k_1 + k_5)/(k_9 + k_{13})) \times 10^{-2} + x_1{}' \\ x_2(0) = ((k_2 + k_6)/(k_{10} + k_{14})) \times 10^{-2} + x_2{}' \\ x_3(0) = ((k_3 + k_7)/(k_{11} + k_{15})) \times 10^{-2} + x_3{}' \\ x_4(0) = ((k_4 + k_8)/(k_{12} + k_{16})) \times 10^{-2} + x_4{}' \end{cases} \tag{16}$$

$$\begin{cases} a = ((k_{17} + k_{21})/(k_{25} + k_{29})) \times 10^{-2} + a' \\ b = ((k_{18} + k_{22})/(k_{26} + k_{30})) \times 10^{-2} + b' \\ c = ((k_{19} + k_{23})/(k_{27} + k_{31})) \times 10^{-2} + c' \\ d = ((k_{20} + k_{24})/(k_{28} + k_{32})) \times 10^{-2} + d' \end{cases} \tag{17}$$

Here $x_1{}'$, $x_2{}'$, $x_3{}'$, $x_4{}'$, a', b', c' and d' are the custom key values.

4.1.2. Scrambling based on parity coordinate transformation

In this paper, a scrambling algorithm based on parity coordinate transformation is proposed, which divides the input image with size of $m \times n$ into four matrices PM_1, PM_2, PM_3 and PM_4 of size $1/2m \times 1/2n$ according to the parity of the coordinates, and then four random sequences generated by the new chaotic system are used to disorder the four matrices of PM_1, PM_2, PM_3 and PM_4 respectively, which sufficiently disrupts the pixel positions, and the operation diagram of this algorithm is illustrated in figure 10. The detailed steps are described below:

Step1: The initial value and parameter value of the new chaotic system is generated by equations (16) and (17), and the discretisation of the new chaotic system is realized by using the fourth-order Runge–Kutta algorithm. During the iteration process, the first 1000 unstable points are discarded in order to enhance the randomness and ensure the stability of the results. Four random sequences x_1, x_2, x_3, x_4 are obtained, and the four random sequences x_1, x_2, x_3, x_4 are transformed into the integer sequences L_1, L_2, L_3, L_4 by equation (18).

$$\begin{cases} L_1 = floor\left(\mathrm{mod}\left(a, b, s, (x_1), \times, 10^6, , \frac{1}{4}, m, n\right)\right) + 1 \\ L_2 = floor\left(\mathrm{mod}\left(a, b, s, (x_2), \times, 10^6, , \frac{1}{4}, m, n\right)\right) + 1 \\ L_3 = floor\left(\mathrm{mod}\left(a, b, s, (x_3), \times, 10^6, , \frac{1}{4}, m, n\right)\right) + 1 \\ L_4 = floor\left(\mathrm{mod}\left(a, b, s, (x_4), \times, 10^6, , \frac{1}{4}, m, n\right)\right) + 1 \end{cases} \tag{18}$$

Where floor means that the value will be 'rounded down' to the nearest integer, mod denotes the remainder operation, abs means to take the absolute value.

Step2: Operate on the four random sequences L_1, L_2, L_3, L_4 respectively, and eventually generate four random sequences $S\_x_1, S\_x_2, S\_x_3, S\_x_4$ of size $1 \times 1/4\,mn$ with no repeated values.

Step3: Sort the four random sequences $S\_x_1, S\_x_2, S\_x_3, S\_x_4$ in ascending order to get $indexS\_x_1$, $indexS\_x_2$, $indexS\_x_3$ and $indexS\_x_4$. The procedure is shown in equation (19).

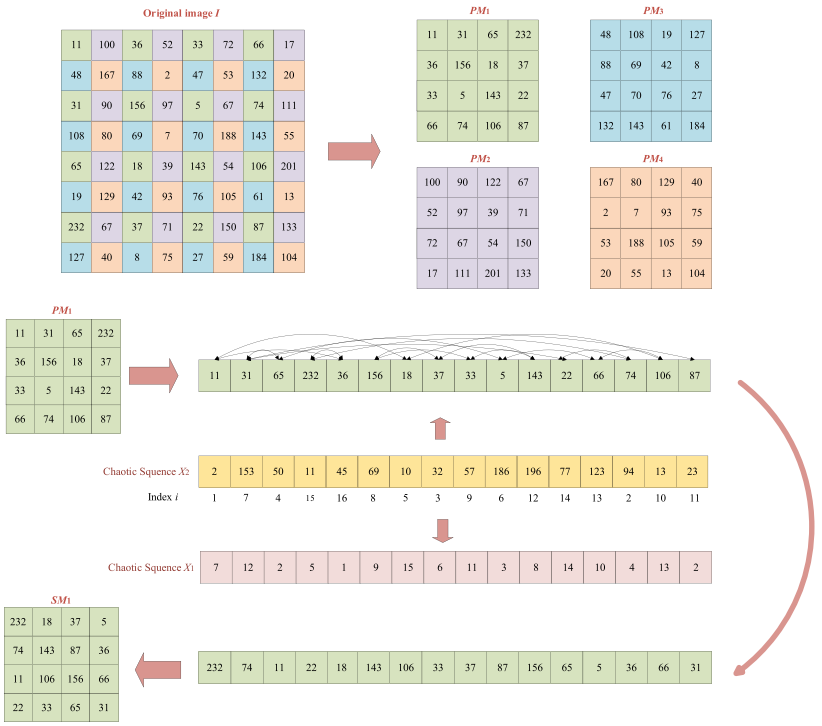

Figure 10. Operation diagram of scrambling of parity coordinate transformations.

$$\begin{cases} [\sim, indexS\_x_1] = sort(S\_x_1, 'ascend') \\ [\sim, indexS\_x_2] = sort(S\_x_2, 'ascend') \\ [\sim, indexS\_x_3] = sort(S\_x_3, 'ascend') \\ [\sim, indexS\_x_4] = sort(S\_x_4, 'ascend') \end{cases} \tag{19}$$

Where sort is the sorting function.

Step4: Expand the four block matrices PM_1, PM_2, PM_3 and PM_4, respectively, into vectors IM_1, IM_2, IM_3 and IM_4 of size $1/4mn$ by columns.

Step5: Select random sequence S_x_1 and index sequence $indexS\_x_2$ rearrangement vector IM_1, exchange $IM_1(indexS\_x_2)$ with $IM_1(S\_x_1(indexS\_x_2))$ to get the disordered vector LM_1. Choose random sequence S_x_2 and index sequence $indexS\_x_3$ rearrangement vector IM_2, swap $IM_2(indexS\_x_3)$ with $IM_2(S\_x_2(indexS\_x_3))$ to get the disordered vector LM_2. Select the random sequence S_x_3 and the index sequence $indexS\_x_4$ to re-order the vector IM_3, and interchange $IM_3(indexS\_x_4)$ with $IM_3(S\_x_3(indexS\_x_4))$ to get the disordered vector LM_3. The random sequence S_x_4 and index sequence $indexS\_x_1$ to rearrange vector IM_4, interchange $IM_4(indexS\_x_1)$ with $IM_4(S\_x_4(indexS\_x_1))$ to get the disordered vector LM_4.

Step6: The vectors LM_1, LM_2, LM_3, LM_4 are reshaped into four matrices SM_1, SM_2, SM_3 and SM_4 of size $1/2m \times 1/2n$ respectively through equation (20), which gives the scrambled image matrices.

$$\begin{cases} SM_1 = reshape\left(LM_1, \frac{1}{2}m, \frac{1}{2}n\right) \\ SM_2 = reshape\left(LM_2, \frac{1}{2}m, \frac{1}{2}n\right) \\ SM_3 = reshape\left(LM_3, \frac{1}{2}m, \frac{1}{2}n\right) \\ SM_4 = reshape\left(LM_4, \frac{1}{2}m, \frac{1}{2}n\right) \end{cases} \tag{20}$$

4.1.3. Diffusion of rotation

Throughout the entire image encryption process , the scrambling algorithm exclusively adjusts the order of the pixels, but does not change the numerical size of the pixels themselves. Therefore, relying on the scrambling algorithm only is not enough to effectively defend against plaintext attacks. Optimally, the image encryption process should have the diffusion property, which means that any pixel information in the original image can be

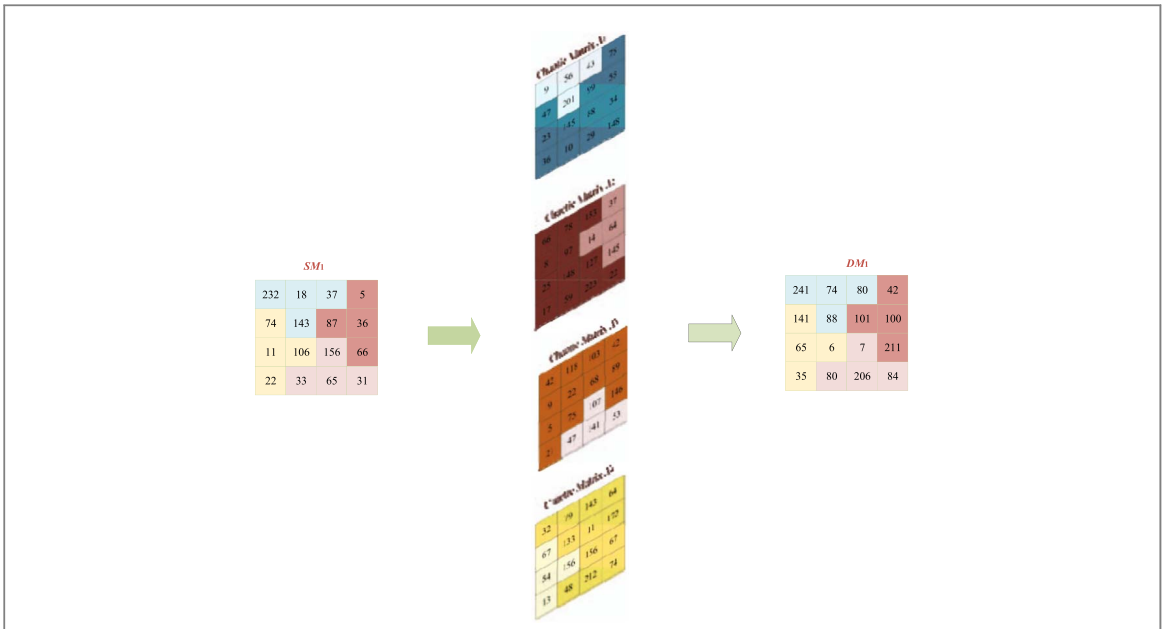

Figure 11. Operation diagram of rotational diffusion.

distributed and concealed in many encrypted pixels by altering the pixel values, thus enhancing the security of encryption. The operation diagram of the rotational diffusion algorithm in this paper is depicted in figure 11. The detailed steps are described as follows:

Step1: Four chaotic matrices X_1, X_2, X_3, X_4 are obtained for changing the pixel values of the scrambled image matrices SM_1, SM_2, SM_3, SM_4 by performing the operation as in equation (21) on the four random sequences x_1, x_2, x_3, x_4 generated by the new chaotic system.

$$\begin{cases} X_1 = reshape\left(\mathrm{mod}\left(round\left(abs\left(x_1\left(1{:}\ \frac{1}{4}mn\right)\right)\times 10^4\right), 256\right), \frac{1}{2}m, \frac{1}{2}n\right) \\ X_2 = reshape\left(\mathrm{mod}\left(round\left(abs\left(x_2\left(1{:}\ \frac{1}{4}mn\right)\right)\times 10^4\right), 256\right), \frac{1}{2}m, \frac{1}{2}n\right) \\ X_3 = reshape\left(\mathrm{mod}\left(round\left(abs\left(x_3\left(1{:}\ \frac{1}{4}mn\right)\right)\times 10^4\right), 256\right), \frac{1}{2}m, \frac{1}{2}n\right) \\ X_4 = reshape\left(\mathrm{mod}\left(round\left(abs\left(x_4\left(1{:}\ \frac{1}{4}mn\right)\right)\times 10^4\right), 256\right), \frac{1}{2}m, \frac{1}{2}n\right) \end{cases} \tag{21}$$

Step2: The operations of equations (22)–(25) are performed on the scrambled matrices SM_1, SM_2, SM_3 and SM_4 using four random matrices X_1, X_2, X_3, X_4 respectively to obtain the matrices DM_1, DM_2, DM_3 and DM_4.

$$\begin{cases} DM_1(r\_s, c\_s{:}\ c\_e) = \mathrm{mod}(SM_1(r\_s, c\_s{:}\ c\_e) + X_1(r\_s, c\_s{:}\ c\_e), 256) \\ DM_1(r\_s+1{:}\ r\_e{-}1, c\_e) = \mathrm{mod}(SM_1(r\_s+1{:}\ r\_e{-}1, c\_e) + X_2(r\_s+1{:}\ r\_e{-}1, c\_e), 256) \\ DM_1(r\_e, c\_e{:}\ {-}1{:}\ c\_s+1) = \mathrm{mod}(SM_1(r\_e, c\_e{:}\ {-}1{:}\ c\_s+1) + X_3(r\_e, c\_e{:}\ {-}1{:}\ c\_s+1), 256) \\ DM_1(r\_e{:}\ {-}1{:}\ r\_s+1, c\_s) = \mathrm{mod}(SM_1(r\_e{:}\ {-}1{:}\ r\_s+1, c\_s) + X_4(r\_e{:}\ {-}1{:}\ r\_s+1, c\_s), 256) \end{cases} \tag{22}$$

$$\begin{cases} DM_2(r\_s, c\_s{:}\ c\_e) = \mathrm{mod}(SM_2(r\_s, c\_s{:}\ c\_e) + X_1(r\_s, c\_s{:}\ c\_e), 256) \\ DM_2(r\_s+1{:}\ r\_e{-}1, c\_e) = \mathrm{mod}(SM_2(r\_s+1{:}\ r\_e{-}1, c\_e) + X_2(r\_s+1{:}\ r\_e{-}1, c\_e), 256) \\ DM_2(r\_e, c\_e{:}\ {-}1{:}\ c\_s+1) = \mathrm{mod}(SM_2(r\_e, c\_e{:}\ {-}1{:}\ c\_s+1) + X_3(r\_e, c\_e{:}\ {-}1{:}\ c\_s+1), 256) \\ DM_2(r\_e{:}\ {-}1{:}\ r\_s+1, c\_s) = \mathrm{mod}(SM_2(r\_e{:}\ {-}1{:}\ r\_s+1, c\_s) + X_4(r\_e{:}\ {-}1{:}\ r\_s+1, c\_s), 256) \end{cases} \tag{23}$$

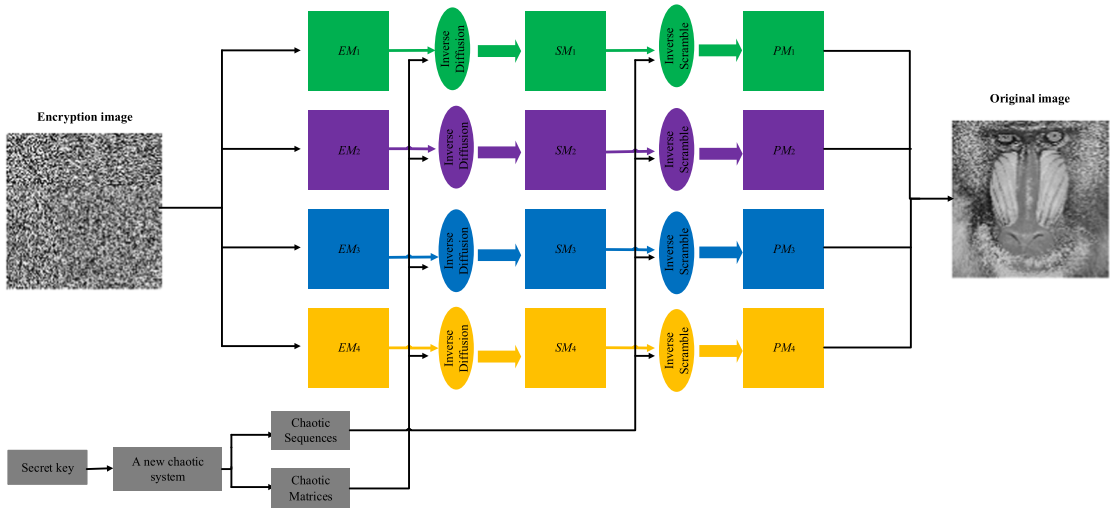

Figure 12. The block diagram of the image decryption algorithm.

$$\begin{cases} DM_3(r\_s,\ c\_s{:}\ c\_e) = \mathrm{mod}(SM_3(r\_s,\ c\_s{:}\ c\_e) + X_1(r\_s,\ c\_s{:}\ c\_e),\ 256) \\ DM_3(r\_s+1{:}\ r\_e{-}1,\ c\_e) = \mathrm{mod}(SM_3(r\_s+1{:}\ r\_e{-}1,\ c\_e) + X_2(r\_s+1{:}\ r\_e{-}1,\ c\_e),\ 256) \\ DM_3(r\_e,\ c\_e{:}\ {-}1{:}\ c\_s+1) = \mathrm{mod}(SM_3(r\_e,\ c\_e{:}\ {-}1{:}\ c\_s+1) + X_3(r\_e,\ c\_e{:}\ {-}1{:}\ c\_s+1),\ 256) \\ DM_3(r\_e{:}\ {-}1{:}\ r\_s+1,\ c\_s) = \mathrm{mod}(SM_3(r\_e{:}\ {-}1{:}\ r\_s+1,\ c\_s) + X_4(r\_e{:}\ {-}1{:}\ r\_s+1,\ c\_s),\ 256) \end{cases} \tag{24}$$

$$\begin{cases} DM_4(r\_s,\ c\_s{:}\ c\_e) = \mathrm{mod}(SM_4(r\_s,\ c\_s{:}\ c\_e) + X_1(r\_s,\ c\_s{:}\ c\_e),\ 256) \\ DM_4(r\_s+1{:}\ r\_e{-}1,\ c\_e) = \mathrm{mod}(SM_4(r\_s+1{:}\ r\_e{-}1,\ c\_e) + X_2(r\_s+1{:}\ r\_e{-}1,\ c\_e),\ 256) \\ DM_4(r\_e,\ c\_e{:}\ {-}1{:}\ c\_s+1) = \mathrm{mod}(SM_4(r\_e,\ c\_e{:}\ {-}1{:}\ c\_s+1) + X_3(r\_e,\ c\_e{:}\ {-}1{:}\ c\_s+1),\ 256) \\ DM_4(r\_e{:}\ {-}1{:}\ r\_s+1,\ c\_s) = \mathrm{mod}(SM_4(r\_e{:}\ {-}1{:}\ r\_s+1,\ c\_s) + X_4(r\_e{:}\ {-}1{:}\ r\_s+1,\ c\_s),\ 256) \end{cases} \tag{25}$$

Where initially $r\_s=1, r\_e=m/2, c\_s=1, c\_e=n/2$, then keep updating the boundary indexes $r\_s=r\_s+1$, $r\_e=r\_e$-1, $c\_s=c\_s+1, c\_e=c\_e$-1, and keep looping until the boundary indexes meet.

Step 3: equation (26) operation is performed on the four matrices DM_1, DM_2, DM_3 and DM_4 to obtain the diffused matrices EM_1, EM_2, EM_3, EM_4.

$$\begin{cases} EM_1 = \mathrm{mod}(DM_1 - DM_2,\ 256) \\ EM_2 = \mathrm{mod}(DM_2 - EM_1,\ 256) \\ EM_3 = \mathrm{mod}(DM_3 - EM_2,\ 256) \\ EM_4 = \mathrm{mod}(DM_4 - EM_3,\ 256) \end{cases} \tag{26}$$

Step 4: The matrices EM_1, EM_2, EM_3, EM_4 are connected as in equation (27) to get the encrypted image E_P.

$$E\_P = reshape\,[EM_1EM_2,\ EM_3EM_4] \tag{27}$$

4.2. Image decryption algorithm

The decryption algorithm and encryption algorithm are inverse operations of each other. Firstly, the encrypted image undergoes inverse rotational diffusion processing to restore it to its pre-diffusion state. Secondly, by applying the inverse scrambling operation based on parity coordinate transformation, the image is restored to its original arrangement order before scrambling. Finally, the original plaintext image can be obtained. The decryption algorithm is shown in figure 12, which illustrates the whole decryption algorithm and key steps.

5. Simulation experiment and performance analysis

The performance evaluation of image encryption/decryption algorithms has been carried out on a computer equipped with Intel(R) Core(TM) i5-6300HQ CPU @ 2.30GHz processor and 8GB RAM using MATLAB 2016b platform under Windows 10 operating system environment. Five grayscale images of Peppers, Clock, Boat, City and Baboon with size of 256 × 256 and 512 × 512 from the USC-SIPI image database are selected as samples for this test. In order to comprehensively evaluate the performance of the new algorithm, several dimensions of

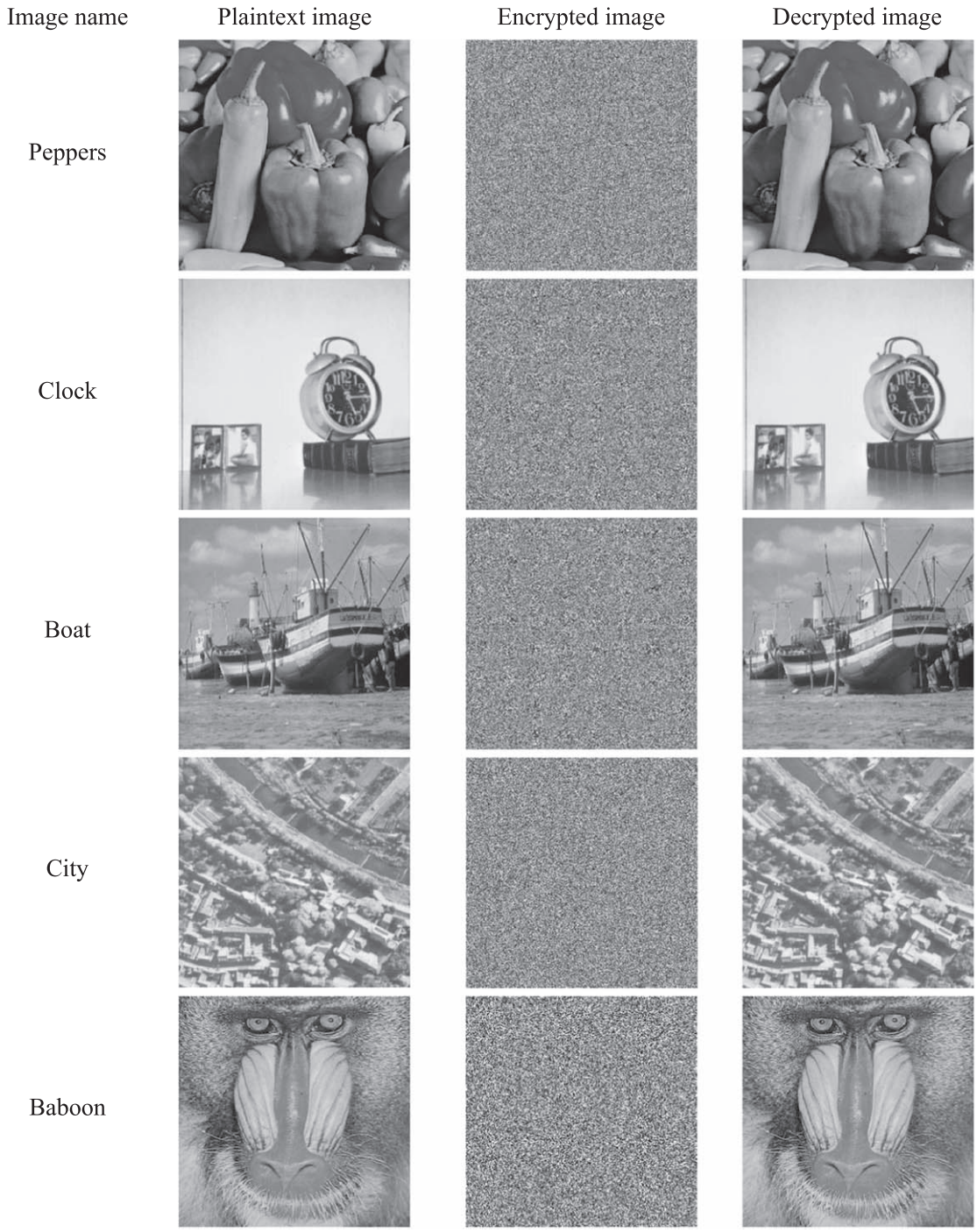

Figure 13. The encryption and decryption simulation results.

visual, histogram, information entropy, key space, key sensitivity, correlation, differential attack test, chosen-plaintext attacks test, chosen-ciphertext attacks test, robustness and algorithm complexity are meticulously analyzed respectively.

5.1. Visual analysis

Detecting dissimilarities and similarities between plaintext images, encrypted images and decrypted images by means of visual analysis is an intuitive and effective method. The encryption and decryption simulation results are presented in figure 13. It is evident from figure 13 that the encrypted image differs significantly from the plaintext image, with no discernible information related to the plaintext image visible in the encrypted image. Furthermore, the decrypted image closely resembles the plaintext image, which demonstrates that the scheme not only performs excellently in protecting the security of the image information, but also is able to accurately

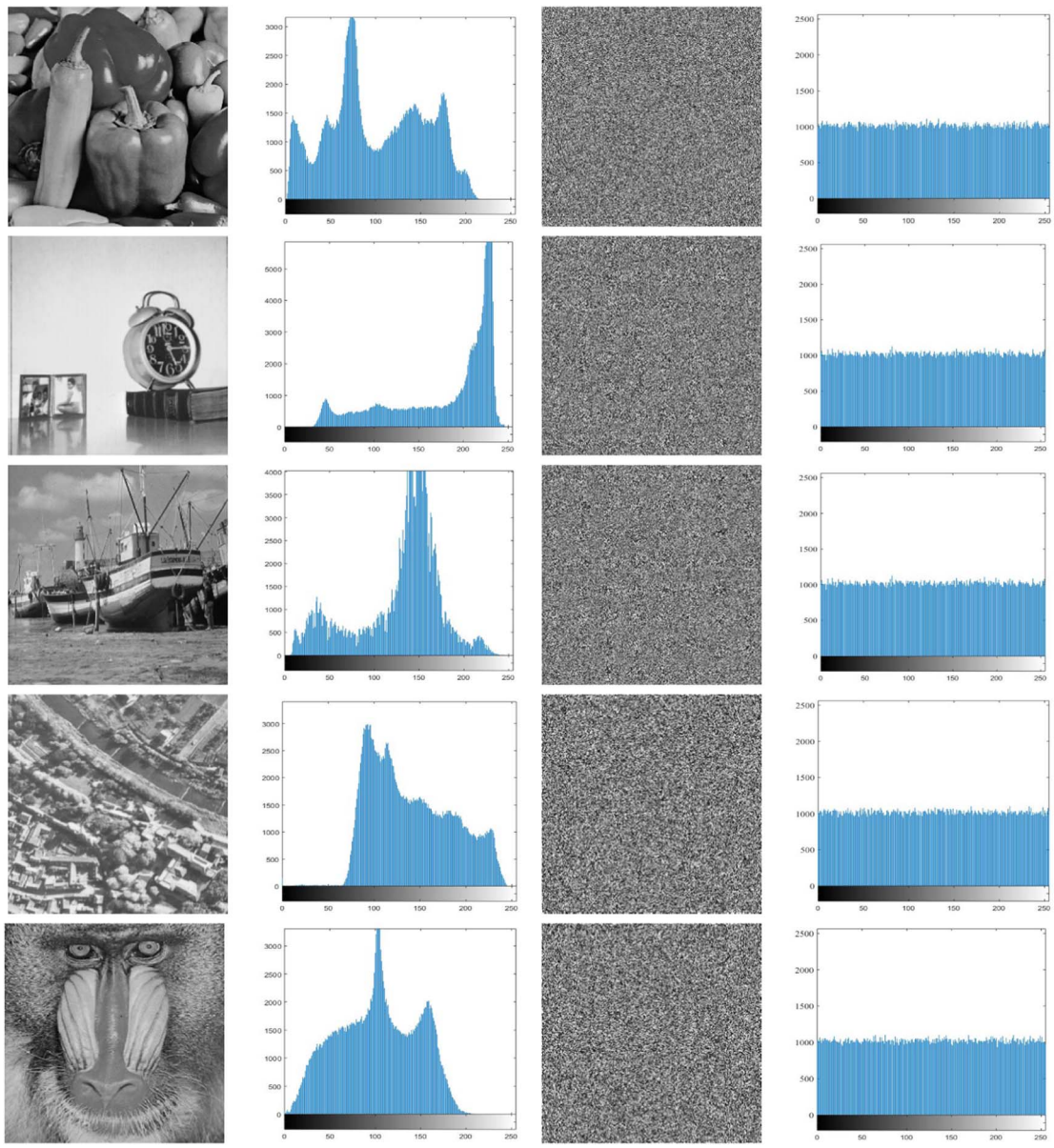

Figure 14. Histograms of original images and encrypted images.

restore the original image during the decryption process, and recovers the original details of the image with almost no loss.

5.2. Histogram analysis

Histogram, as a common approach for analyzing the statistical characteristics of image, can visualize the distribution of pixel values. An efficient and secure image encryption algorithm should be able to conceal the plaintext image's pixel distribution information , resulting in a uniform distribution of pixel values in the encrypted image [42], thus effectively resisting statistical attacks. As depicted in figure 14, the uneven distribution of pixel values in the plaintext image is indicated by the large fluctuations and variations seen in the plaintext image histogram. However, the histogram of the ciphertext image exhibits significant even distribution characteristics, which proves that the encryption algorithm successfully hides the pixel distribution information of the plaintext image, thus it can effectively avoiding statistical attacks.

5.3. Information entropy analysis

Information entropy is an important index to quantify the uncertainty of image data, and it is widely used in the field of image processing to assess the degree of randomness or chaos in an image. It is computed as follows.

Table 1. Information entropy of various images.

Image size	Images	Plaintext image	Ciphertext image
256 × 256	Peppers	7.5681	7.9970
	Clock	6.7057	7.9974
	Boat	7.1587	7.9970
	City	7.3118	7.9972
	Baboon	7.3897	7.9977
512 × 512	Peppers	7.5395	7.9994
	Clock	6.7224	7.9993
	Boat	7.1914	7.9994
	City	7.3169	7.9993
	Baboon	7.3876	7.9994

Table 2. Information entropy comparison of various algorithms on the Baboon image.

Encryption algorithms	Proposed	[44]	[45]	[46]	[47]	[48]
Information entropy	7.9994	7.9989	7.9993	7.9986	7.9971	7.9993

$$H(v) = -\sum_{i=1}^{L} p(v_i)\log p(v_i) \tag{28}$$

Where $p(v_i)$ denotes the probability of the ith pixel value of image v and L is the gray level of the image. A higher information entropy value indicates greater complexity and randomness, with a maximum theoretical value of 8 [43]. Table 1 presents the results of testing the ciphertext images with different sizes, which indicate that the information entropy values approach the maximum of 8, suggesting high levels of randomness. To further verify the efficacy of the algorithm, information entropy results obtained after encrypting the Baboon test image are compared with those reported in other investigations, as depicted in table 2. This comparison highlights how the suggested algorithm's information entropy closely approaches the maximum theoretical value, which further verifies that this algorithm has a superior advantage in encryption effect and security.

5.4. Correlation analysis

The correlation of neighbor pixel is an important indicator of the degree of correlation between pixels at adjacent locations within an image. When the correlation coefficient tends to zero, it indicates a weak correlation between pixels; while a large value of correlation coefficient indicates stronger pixel correlation. Neighbouring pixels in a plaintext image show a high degree of correlation between them, therefore, to safeguard image confidentiality, encrypted images should be near to zero correlation coefficients [49]. The correlation coefficient is calculated as shown in equation (29).

$$\begin{cases} E(x) = \dfrac{1}{N}\displaystyle\sum_{i=1}^{N} x_i \\ D(x) = \dfrac{1}{N}\displaystyle\sum_{i=1}^{N}(x_i - E(x)^2) \\ \mathrm{cov}(x, y) = \dfrac{1}{N}\displaystyle\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \\ \rho_{xy} = \dfrac{\mathrm{cov}(x, y)}{\sqrt{D(x)}\sqrt{D(y)}} \end{cases} \tag{29}$$

Where $E(x)$, $D(x)$, $\mathrm{cov}(x,y)$, ρ_{xy} are mean, variance, covariance and correlation respectively. Where N represents the pair of any adjacent pixel, and its gray value is (x_i, y_i), $i = 1, 2, \ldots, N$, the vector $x = \{x_i\}$, the vector $y = \{y_i\}$. Correlation analysis was performed on the plaintext image and the encrypted ciphertext image by randomly picking 5000 pairs of neighbouring pixels from the horizontal, vertical and diagonal directions of each image, respectively. The findings are illustrated in figure 15 and table 3, and it denote that the correlation scatter of the ciphertext image has a uniform distribution, and its correlation coefficient is extremely near to 0, which demonstrates that this encryption algorithm possess excellent performance in breaking the pixel correlation, and it can effectively protect the image information.

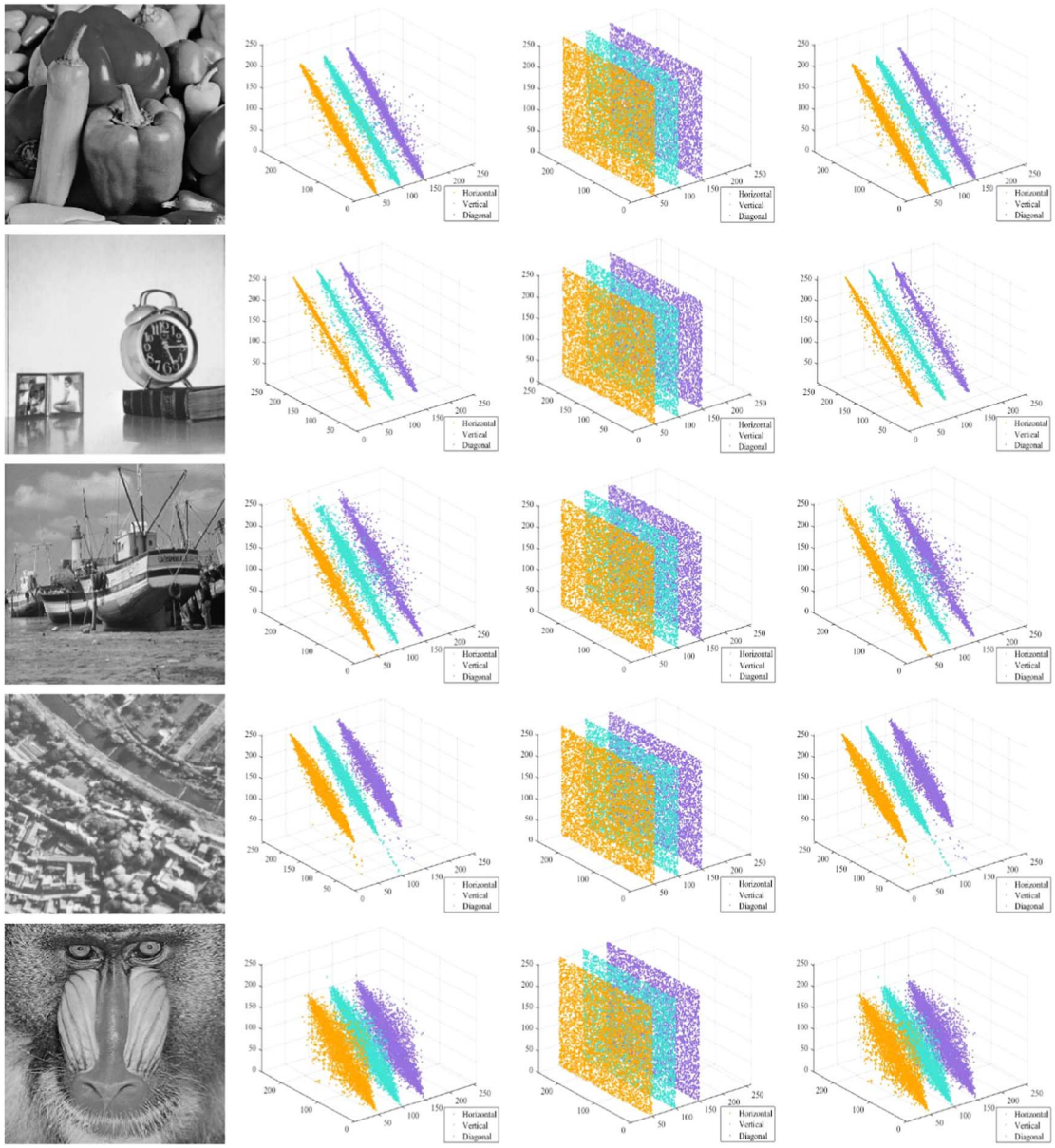

Figure 15. Correlation analysis results of plaintext images and ciphertext images.

Table 3. Correlation coefficient comparison between plaintext and ciphertext images.

Image size	Images	Plaintexts			Ciphertexts		
		Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
256 × 256	Peppers	0.9680	0.9718	0.9405	−0.0076	0.0318	0.0021
	Clock	0.9479	0.9785	0.9350	−0.0006	0.0155	−0.0070
	Boat	0.9267	0.9505	0.8896	−0.0187	−0.0100	0.0082
	City	0.9008	0.8314	0.7958	−0.0008	0.0194	−0.0067
	Baboon	0.6993	0.6054	0.6033	0.0240	−0.0034	0.0078
512 × 512	Peppers	0.9804	0.9833	0.9721	0.0030	0.0143	0.0114
	Clock	0.9917	0.9953	0.9870	0.0008	0.0042	−0.0112
	Boat	0.9367	0.9716	0.9193	−0.0118	0.0075	−0.0032
	City	0.9772	0.9692	0.9566	−0.0114	0.0112	0.0092
	Baboon	0.8668	0.7544	0.7243	−0.0007	−0.0014	−0.0072

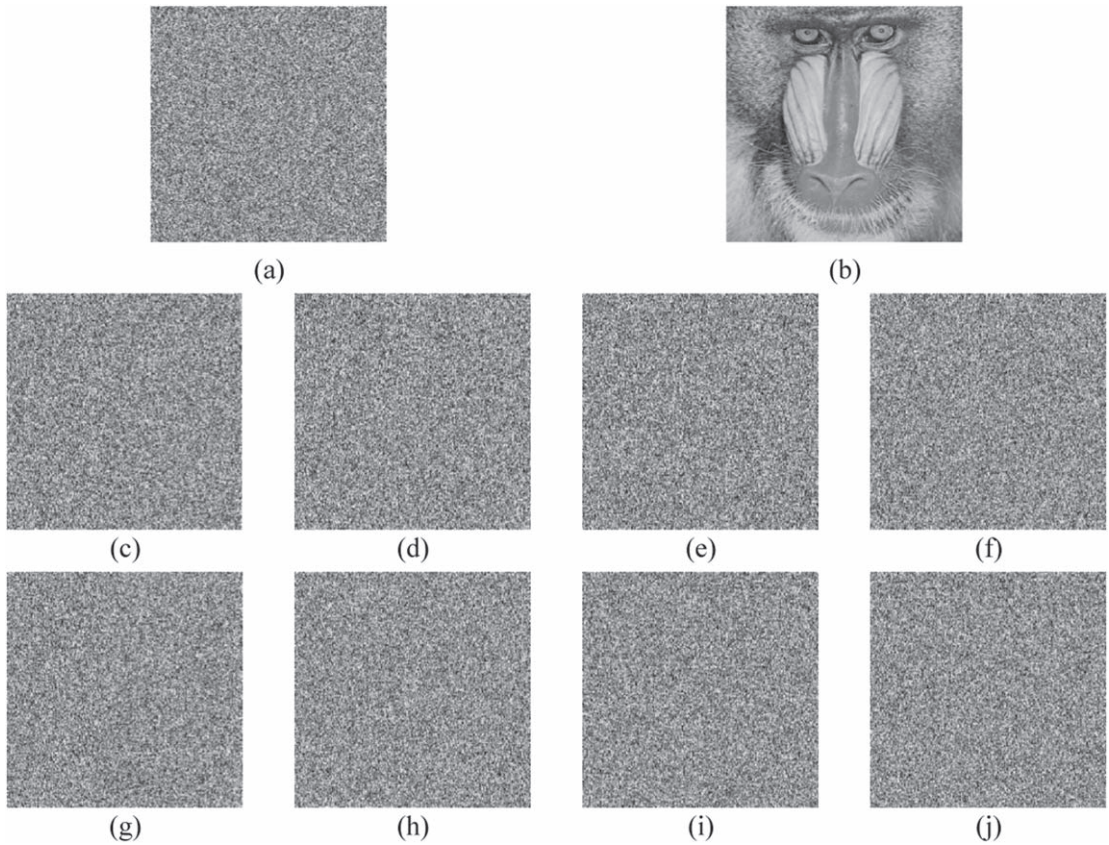

Figure 16. The decryption diagram obtained by decrypting ciphertext image 'Baboon' after changing different key values. (a) ciphertext image;(b) decrypted image with correct keys; (c) $x_1(0) = x_1(0) + 10^{-15}$; (d) $x_2(0) = x_2(0) + 10^{-15}$; (e) $x_3(0) = x_3(0) + 10^{-15}$; (f) $x_4(0) = x_4(0) + 10^{-15}$; (g) $a = a_0 + 10^{-15}$; (h) $b = b_0 + 10^{-15}$; (i) $c = c_0 + 10^{-15}$; (j) $d = d_0 + 10^{-15}$.

Table 4. Key space size comparison.

Algorithms	Proposed	[51]	[52]	[53]	[54]	[55]
Key space	2^{398}	2^{196}	2^{260}	2^{116}	2^{194}	2^{232}

5.5. Key space analysis

To make the ciphertext image secure against brute force attacks, the key space must be sufficiently large. In general, when the key space exceeds 2^{100}, the encryption algorithm has sufficient defense against brute force attacks [50]. In this encryption algorithm, the key consists of four initial values $x_1(0), x_2(0), x_3(0), x_4(0)$ and four system parameters a, b, c, d. Each component has a space size of 10^{15}, hence, the total key space is $(10^{15})^8 \approx 2^{398}$, which is far more than the minimum defense value against brute force attack, so the algorithm has a strong resistance against brute force attack. Furthermore, the key space of this scheme is compared with the existing encryption algorithms, detailed in table 4. The comparison reveals that the key space of this algorithm significantly outperforms the other encryption algorithms and has resistantance to brute force attacks.

5.6. Key sensitivity analysis

A good encryption algorithm not only requires a large key space to provide strong security, but must also need extreme sensitivity to key changes. This sensitivity is reflected in the fact that when even minor alterations in the key occurs, a decryption operation on the encrypted image will not result in a correct decrypted image. As shown in figure 16, when each parameter in the key is interfered by 10^{-15}, respectively, the generated key is unable to obtain the correct image information through the decryption operation. Correct decryption of the plaintext image is achievable solely with the correct key. This result fully demonstrates the excellent performance of the proposed encryption algorithm in terms of key sensitivity.

Table 5. NPCR and UACI analysis across various images.

Image size	Images	NPCR(%)	UACI(%)
256 × 256	Peppers	99.66	33.42
	Clock	99.59	33.45
	Boat	99.60	33.50
	City	99.60	33.49
	Baboon	99.59	33.41
512 × 512	Peppers	99.60	33.41
	Clock	99.58	33.55
	Boat	99.59	33.45
	City	99.62	33.47
	Baboon	99.60	33.44

Table 6. The NPCR and UACI comparison of various algorithms on the Baboon image.

Algorithms	**Proposed**	[57]	[58]	[46]	[47]	[59]
NPCR(%)	99.60	99.62	99.62	99.63	99.59	60.28
UACI(%)	33.44	27.89	33.42	33.17	31.56	21.40

5.7. Differential attacks analysis

Differential attack serves as an essential analytical tool used to evaluate the sensitivity of an encryption scheme to plaintext images. To improve security against differential attacks, the algorithm must be extremely sensitive to the plaintext alterations. When plaintext image and the plaintext image with a slight change are encrypted, and the two ciphertext images obtained are significantly different. The number of pixels change rate (NPCR) and the uniform average change intensity (UACI) are commonly used to evaluate the ability of encryption algorithms in terms of resistance to differential attacks. The NPCR and UACI are calculated by equation (30).

$$NPCR(\varepsilon_1, \varepsilon_2) = \frac{1}{S_m S_n}\sum_{i=1}^{S_m}\sum_{j=1}^{S_n}|\text{sign}(\varepsilon_1(i,j) - \varepsilon_2(i,j))| \times 100\%$$
$$UACI = \frac{1}{S_m S_n}\sum_{i=1}^{S_m}\sum_{j=1}^{S_n}\frac{|\varepsilon_1(i,j) - \varepsilon_2(i,j)|}{255 - 0} \times 100\% \quad (30)$$

Where $\varepsilon_1(i, j)$ is the encrypted images of the original plaintext image , and $\varepsilon_2(i, j)$ represents the encrypted images of the plaintext image with minor changes. S_m and S_n represent the number of rows and columns of the image, respectively. Theoretical benchmarks for NPCR and UACI are 99.6094% and 33.4635%, respectively [56]. Individual pixel values were randomly changed for several test images, and the NPCR and UACI of the cipher images before and after changes were subsequently calculated. The experimental results are listed in detail in table 5, it can be clearly observed that the NPCR and UACI values are very close to the desired benchmarks. Furthermore, in table 6 the NPCR and UACI values of Baboon image are compared with other encryption algorithms. The comparative results underscore the superior resistance of our encryption algorithm to differential attacks.

5.8. Chosen-plaintext attacks and chosen-ciphertext attacks analysis

When evaluating the security of a encryption algorithm, the ability to withstand chosen- plaintext attacks (CPA) and chosen-ciphertext attacks (CCA) are crucial criteria. If a encryption algorithm is weak in these two areas, its security will be seriously threatened. Attackers select a specific plaintext or ciphertext to try to crack the algorithm, often using all-black or all-white images for testing. Therefore the defence capability of the CCA can be assessed by testing against an all-black or all-white image. Figure 17 and table 7 show the simulation results and the values of each index after encrypting the all-black and all-white images, respectively, and it can be seen that the histogram of the ciphertext image is uniformly distributed, the correlation coefficient is close to 0, the information entropy is close to 8, and the values of NPCR and UACI are close to the desirable values. In addition, the secret key generation in the design of this encryption algorithm relies on the plaintext information and custom key, and this mechanism effectively improves the ability of the algorithm to resist CCA [9]. In summary, the present encryption algorithm has strong resistance to CPA and CCA, which ensures the security and reliability of the encryption process.

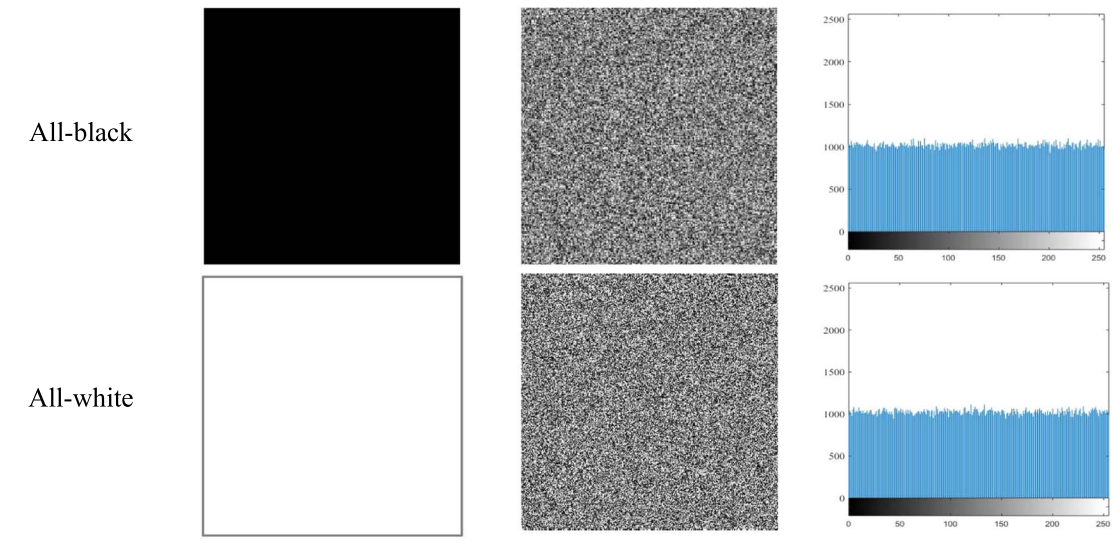

Figure 17. Simulation results after encrypting all-black and all-white images.

Table 7. Indicator values after encrypting all-black and all-white images.

Images	Information entropy	Correlation coefficient			NPCR(%)	UACI(%)
		Horizontal	Vertical	Diagonal		
All-black All-white	7.9992 7.9993	−0.0032 0.0061	0.0027 –0.0019	−.0047 –.0039	99.62 99.60	33.47 33.44

5.9. Robustness analysis

Excellent encryption algorithms should have the robustness against interference during transmission, ensuring that the decrypted image can still maintain a high degree of identifiability and integrity even in the face of noise, shearing and other interference factors. In order to test the resistance of the algorithm to noise, different degrees of saltpeter noise (SPN) with 0.1%, 1%, 10% are added to the encrypted image as shown in figure 18. Even with this noise, the decrypted image is still able to basically recover the original content. This result shows that the encryption algorithm can effectively withstand noise attacks.

To evaluate the anti-cropping ability of the encryption algorithm, the ciphertext image of 512×512 is cropped to different degrees of 32×32, 64×64 and 128×128, respectively, as shown in figure 19, and the image recovered by the decryption algorithm is still able to maintain the important information of the plaintext image, even in the presence of significant data loss due to cropping. This result demonstrates the excellent robustness of the encryption algorithm in the face of cropping attacks.

To quantitatively assess the quality difference between the decrypted image after an attack and the original image, two widely recognised statistical metrics are employed: peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM). PSNR is employed to measure the degree of degradation of image quality, with larger values of PSNR indicating less impact by the attack [60]; SSIM assesses the structural similarity between the plaintext and decrypted images. The closer the SSIM value is to 1, the more structurally similar the plaintext image is to the decrypted image [61]. PSNR and SSIM are calculated as follows.

$$\begin{aligned} MSE &= \frac{1}{S_{\mathrm{m}} \times S_{\mathrm{n}}} \sum_{i=1}^{S_m} \sum_{j=1}^{S_n} (P(i,j) - D(i,j))^2 \\ PSNR &= 10 \times \log_{10}\left(\frac{255^2}{MSE}\right) \\ SSIM &= \frac{(2\bar{P}\bar{D} + \nu_1)(2\sigma_P\sigma_D + \nu_2)}{(\bar{P}^2 + \bar{D}^2 + \nu_1)(\sigma_P^2 + \sigma_D^2 + \nu_2)} \end{aligned} \tag{31}$$

Where $P(i, j)$ and $D(i, j)$ denote the plaintext image and the decrypted image after the cipher image is attacked, respectively. S_m and S_n represent the number of rows and columns of the image, respectively. $\bar{P}$ and $\bar{D}$ are the

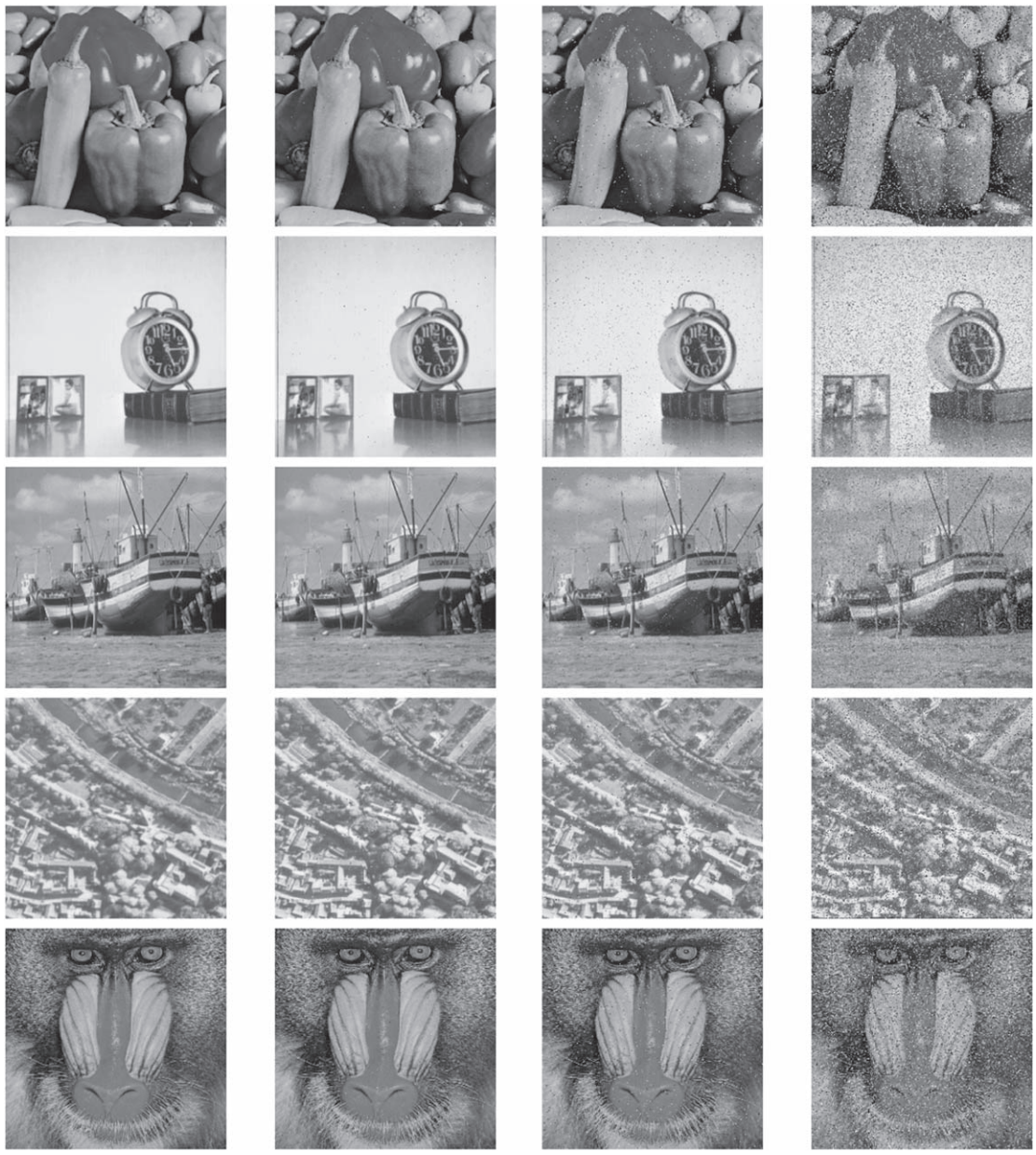

Figure 18. Different levels of noise attacks.

mean values of P and D respectively. σ_P^2 and σ_D^2 are the variances of P and D, respectively. $\nu_1 = (\xi_1 i)^2$, $\nu_2 = (\xi_2 i)^2$, $\xi_1 = 0.01$, $\xi_2 = 0.03$.

By calculating and analyzing PSNR and SSIM, it is possible to objectively assess the quality of the decrypted image while the encrypted image suffering from these attacks, so as to comprehensively reflect the advantages and disadvantages of the robustness of the encryption algorithm, and the results are shown in tables 8 and 9 respectively. Additionally, a comparison is made with the various encryption algorithms, as summarized in table 10. Through the experiments and analyses, it is found that this encryption algorithm can effectively recover the key information of the encrypted image with attacks.

5.10. Algorithm complexity analysis

Algorithm complexity is an important indicator of the efficiency of algorithm execution, the computational complexity and time efficiency are usually utilised to assess the complexity of an algorithm. For the encryption of an image with input size M × N, the scrambling phase, the computational complexity is O($M \times N$); the diffusion phase, the computational complexity is O($M \times N$), so the total computational complexity of the encryption

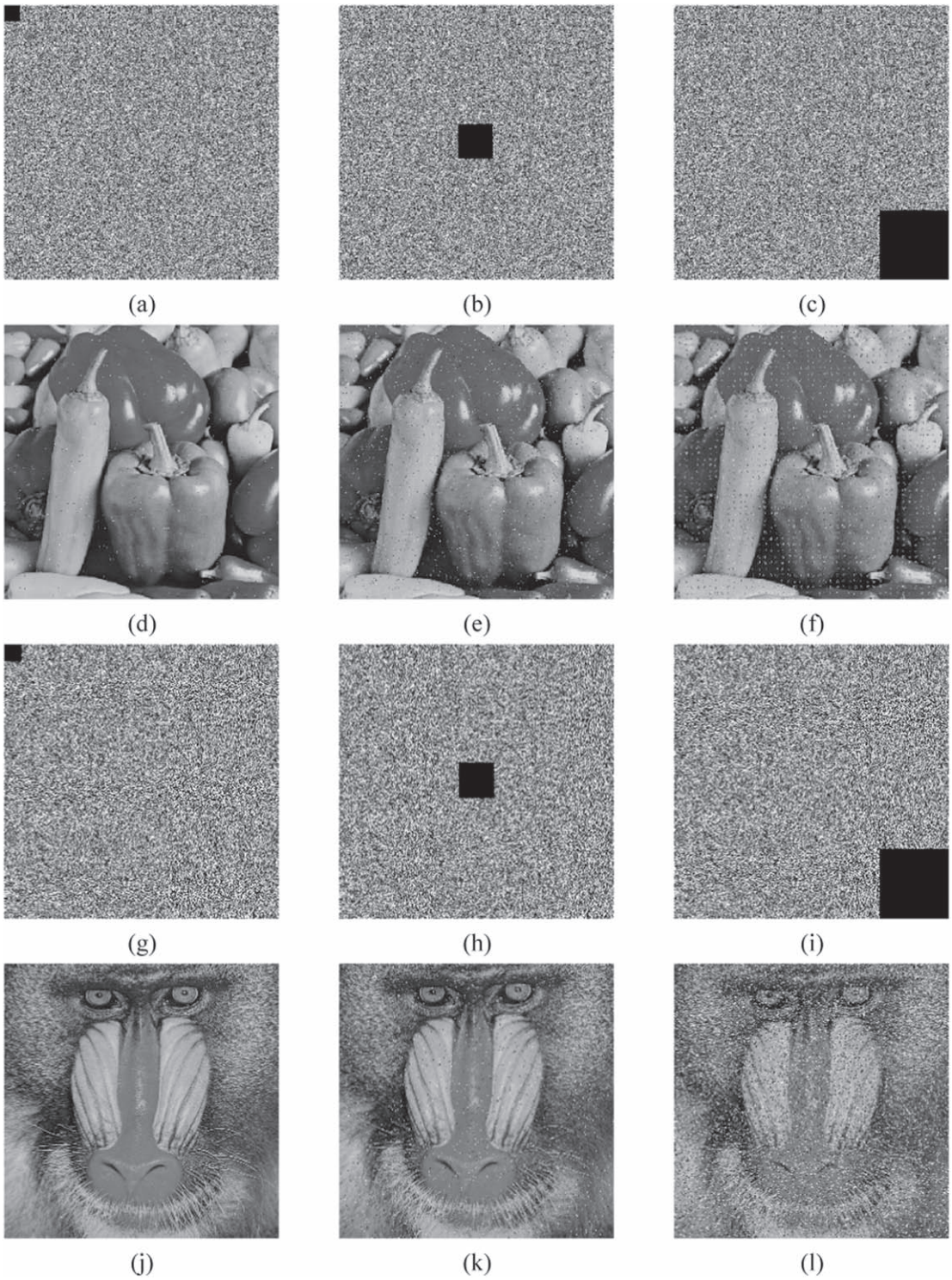

Figure 19. Cropping attacks on Peppers and Baboon. (a) Cut 32 × 32 of encrypted image. (b) Cut 64 × 64 of encrypted image. (c) Cut 128 × 128 of encrypted image. (d) Cut 32 × 32 of decrypted image. (e) Cut 64 × 64 of decrypted image. (f) Cut 128 × 128 of decrypted image. (g) Cut 32 × 32 of encrypted image. (h) Cut 64 × 64 of encrypted image. (i) Cut 128 × 128 of encrypted image. (j) Cut 32 × 32 of decrypted image. (k) Cut 64 × 64 of decrypted image. (l) Cut 128 × 128 of decrypted image.

algorithm is O($2M \times N$), indicating that its computational complexity is relatively low. In order to further evaluate the time efficiency performance of the encryption algorithm, time tests are performed on different sized images. Table 11 details the encryption time for encrypting different size images using this encryption algorithm. In addition, the encryption time of Baboon image was compared with other encryption algorithms as shown in table 12. The comparison shows that this encryption algorithm has high time efficiency.

Table 8. PSNR evaluation between plain and decrypted images under SPN and Cropping Attacks.

Image	SPN level	Proposed	Cropping size	Proposed
Peppers	0.001	36.34	32 × 32	29.81
	0.01	25.65	64 × 64	23.49
	0.1	15.81	128 × 128	20.55
Clock	0.001	35.13	32 × 32	29.47
	0.01	25.14	64 × 64	23.21
	0.1	15.51	128 × 128	20.17
Boat	0.001	37.12	32 × 32	30.38
	0.01	26.26	64 × 64	24.27
	0.1	16.55	128 × 128	21.34
City	0.001	35.91	32 × 32	30.44
	0.01	26.27	64 × 64	24.14
	0.1	16.54	128 × 128	21.25
Baboon	0.001	37.22	32 × 32	30.14
	0.01	26.24	64 × 64	23.90
	0.1	16.33	128 × 128	21.08

Table 9. SSIM comparison of various images under Noise and Cropping attacks.

Image	SPN level	Proposed	Cropping size	Proposed
Peppers	0.001	0.9697	32 × 32	0.8811
	0.01	0.7296	64 × 64	0.6107
	0.1	0.1720	128 × 128	0.4125
Clock	0.001	0.9548	32 × 32	0.8408
	0.01	0.7409	64 × 64	0.6209
	0.1	0.1384	128 × 128	0.4101
Boat	0.001	0.9782	32 × 32	0.9014
	0.01	0.7784	64 × 64	0.6839
	0.1	0.2404	128 × 128	0.5000
City	0.001	0.9761	32 × 32	0.9199
	0.01	0.8144	64 × 64	0.7282
	0.1	0.2912	128 × 128	0.5629
Baboon	0.001	0.9869	32 × 32	0.9407
	0.01	0.8614	64 × 64	0.7873
	0.1	0.3867	128 × 128	0.6508

Table 10. Resistance to Noise and Cropping Attacks comparison of various algorithms on the Baboon image.

Attack	Attack	PSNR			
Type	Intensity	[62]	[63]	[64]	Proposed
	0.001	—	—	—	37.22
SPN	0.01	20.42	19.97	23.70	26.24
	0.01	10.82	11.55	14.50	16.33
	32 × 32	—	—	—	30.14
Cropping	64 × 64	—	—	—	23.90
	128 × 128	12.76	15.34	7.18	21.08

6. Conclusions

In this paper, a four-dimensional chaotic system featuring multi-scroll hidden attractors is proposed, which is able to produce chaotic sequences with more complex dynamics. Leveraging the unique properties of this system, a new image encryption algorithm is developed. The new image encryption approach adopts a newly proposed scrambling algorithm that is based on parity coordinate transformation to scramble pixel positions in the plaintext image first, then use a newly proposed rotational diffusion algorithm to change the pixel value of the image to conceal information content of the original image, thus to achieve desired encryption effect. The

Table 11. Time performance results.

Image size	Image	Encryption time (s)
256 × 256	Peppers	0.1786
256 × 256	Clock	0.1645
256 × 256	Boat	0.1846
256 × 256	City	0.1748
256 × 256	Baboon	0.1683
512 × 512	Peppers	0.2358
512 × 512	Clock	0.2363
512 × 512	Boat	0.2465
512 × 512	City	0.2457
512 × 512	Baboon	0.2398

Table 12. Runtime comparison of various algorithms on the Baboon image.

Encryption algorithms	Proposed	[65]	[66]	[67]	[68]	[48]
Encryption time (s)	0.2398	1.6464	1.5069	18.7	2.8805	0.5211

effectiveness of the proposed algorithm is evaluated through extensive simulations and performance analyses using grayscale images. Results demonstrate that the algorithm possesses a large key space and exhibits strong robustness, which ensures strong resistance to external attacks, as a result the security of encrypted images is significantly improved. This research contributes to advancing encryption techniques through the application of chaotic systems with multi-scroll hidden attractors, thereby enhancing the security and confidentiality of digital image transmission and storage systems. The multi-scroll hidden chaotic attractors proposed in this paper is only in a single direction, and the research on the new chaotic system is limited to theoretical analyses and numerical simulations. Therefore, the future work will aim at the realisation of generating multi-scroll hidden attractors in multi-direction as well as the physical implementation of the proposed model.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Competing interests

The authors declare no conflict of interest.

Authors contributions

Pengfei Ding: Supervision, Methodology, Investigation, Validation, Writing-review & editing; **Jingge Zhu:** Methodology, Investigation, Formal analysis, Validation, Visualization, Writing-original draft; **Juan Zhang:** Investigation, Visualization,Writing-review & editing.

ORCID iDs

Pengfei Ding https://orcid.org/0000-0002-3297-4438
Jingge Zhu https://orcid.org/0009-0009-2932-9700
Juan Zhang https://orcid.org/0009-0003-2401-8619

References

[1] Atangana A and Araz S R 2020 Atangana-seda numerical scheme for labyrinth attractor with new differential and integral operators *Fractals* **28** 2040044
[2] A Boya B F B and Kengne J 2023 Adjustable symmetry on the dynamics of a new chaotic system with cyclic symmetry: theoretical study, control and experimental investigation *Phys. Scr.* **98** 095233
[3] Sun J, Li C, Lu T, Akgul A and Min F 2020 A memristive chaotic system with hypermultistability and its application in image encryption *IEEE Access* **8** 139289

[4] Munir N, Khan M, Wei Z, Akgul A, Amin M and Hussain I 2020 Circuit implementation of 3d chaotic self-exciting single-disk homopolar dynamo and its application in digital image confidentiality *Wireless Networks* **9**
[5] Nazari M and Mehrabian M 2021 A novel chaotic IWT-LSB blind watermarking approach with flexible capacity for secure transmission of authenticated medical images *Multimed. Tools Appl.* **80** 1–41
[6] Hemdan E D 2020 An efficient and robust watermarking approach based on single value decompression, multi-level DWT, and wavelet fusion with scrambled medical images *Multimed. Tools Appl.* **19** 1–29
[7] Zambrano-Serrano E and Anzo-Hernandez A 2021 A novel antimonotic hyperjerk system: analysis, synchronization and circuit design *Phys. D Nonlinear phenom.* **424** 132927
[8] Boya B F B A, Ramakrishnan B, Effa J Y, Kengne J and Rajagopal K 2023 Effects of bias current and control of multistability in 3D hopfield neural network *Heliyon* **9** e13034
[9] Lin H R, Deng X H, Yu F and Sun Y C 2024 Grid multi-butterfly memristive neural network with three memristive systems: modeling, dynamic analysis, and application in police IoT *IEEE Internet of Things Journal* 3409373
[10] Said S, Tanougast C, Azzaz M S and Dandache A 2013 Design and fpga implementation of a wireless hyperchaotic communication system for secure real-time image transmission *Eurasip. J. Image Vide.* **43** 2013
[11] Filali R L, Benrejeb M and Borne P 2014 On observer-based secure communication design using discrete-time hyperchaotic systems *Commun. Nonlinear Sci. Numer. Simul.* **19** 1424–32
[12] Hassan and Mohamed F 2014 A new approach for secure communication using constrained hyperchaotic systems *Appl. Math. Comput.* **246** 711–30
[13] Wu X, Fu Z, Kurths and Jürgen 2015 A secure communication scheme based generalized function projective synchronization of a new 5D hyperchaotic system *Phys. Scri.* **90** 045210
[14] Rajagopal K, Panahi S, Karthikeyan A, Alsaedi A, Pham V T and Hayat T 2018 Some new dissipative chaotic systems with cyclic symmetry *Int. J. Bifurcat. Chaos* **28** 13850164
[15] Zhang C and Yu S 2013 On constructing complex grid multi-wing hyperchaotic system: theoretical design and circuit implementation *Int. J. Circ. Theor. App.* **41** 221–37
[16] Lu J, Chen G and Yu X 2004 Design and analysis of multiscroll chaotic attractors from saturated function series *IEEE T CIRCUITS-I* **51** 2476–90
[17] Wang C H, Hao X U and Fei Y U 2013 A novel approach for constructing high-order chua's circuit with multi-directional multi-scroll chaotic attractors *Int. J. Bifurcat. Chaos* **23** 50022
[18] Kuate P D K, Tchendjeu A E T and Fotsin H 2020 A modified Rössler prototype-4 system based on Chua's diode nonlinearity : dynamics, multistability, multiscroll generation and FPGA implementation *Chaos Soliton. Fract.* **140** 110213
[19] Wang F and Xiao Y 2020 A multiscroll chaotic attractors with arrangement of saddle-shapes and its field programmable gate array (FPGA) implementation *Complexity* **4** 1–8
[20] He P, Liu H W, Li G D, Xu X L and Gu Y J 2023 A general method for generating multi-scroll and multi-wing chaotic systems and its implementation of attractor reproduction *Phys. Scr.* **98** 085232
[21] Lin H R, Wang C H and Sun Y C 2024 A universal variable extension method for designing multiscroll/wing chaotic systems *IEEE T. Ind. Electron.* **71** 7806–18
[22] Jafari S and Pham V T 2016 Multiscroll chaotic sea obtained from a simple 3d system without equilibrium *Int. J. Bifurcat. Chaos* **26** 1650031
[23] Hu X, Liu C, Liu L, Ni J and Li S 2016 Multi-scroll hidden attractors in improved Sprott A system *Nonlinear Dyn.* **86** 1725–34
[24] Escalante-González R J, Campos-Cantón E and Nicol M 2017 Generation of multi-scroll attractors without equilibria via piecewise linear systems *Chaos* **27** 053109
[25] Zhang X and Wang C 2019 Multiscroll Hyperchaotic System with Hidden Attractors and Its Circuit Implementation *Int. J. Bifurcat. Chaos* **29** 1157–71
[26] Dolvis L G, Vaidyanathan S, Jacques K, Sambas A and Mamat M 2019 A new 4-D hyperchaotic system with four-scroll hidden attractor, its properties and bifurcation Analysis *Mat. Sci. Eng.* **621** 012014
[27] Tang L, He Z, Yao Y and C Y 2024 A generalised incomplete no-equilibria transformation method to construct a hidden multi-scroll system with no-equilibrium *Int. J. Comput. Sci. Eng.* **27** 57–67
[28] Zhang L, Li Z and Peng Y 2024 A hidden grid multi-scroll chaotic system coined with two multi-stable memristors *Chaos Soliton Fract* **185** 115109
[29] Zhu H, Zhao Y and Song Y 2019 2D logistic-modulated-sine-coupling-logistic chaotic map for image encryption *IEEE Access* **7** 14081–98
[30] Zhu H, Qi W, Ge J and Liu Y 2018 Analyzing devaney chaos of a sine-cosine compound function system *Int. J. Bifurcat. Chaos* **28** 1850176
[31] Zhu H, Zhang X, Yu H, Zhao C and Zhu Z 2017 An image encryption algorithm based on compound homogeneous hyper-chaotic system *Nonlinear Dyn.* **89** 61–79
[32] Cao W, Zhou Y, Chen C L and Xia L 2017 Medical image encryption using edge maps *Signal Process.* **132** 96–109
[33] Zhou Y, Cao W and Chen C L P 2014 Image encryption using binary bitplane *Signal Process.* **100** 197–207
[34] Wang X and Luan D 2013 A novel image encryption algorithm using chaos and reversible cellular automata *Commun. Nonlinear Sci. Numer. Simul.* **18** 3075–85
[35] Cao W, Mao Y and Zhou Y 2020 Designing a 2d infinite collapse map for image encryption *Signal Process.* **171** 107457
[36] Li Q F and Chen L 2024 An image encryption algorithm based on 6-dimensional hyper chaotic system and DNA encoding *Multimed. Tools Appl.* **83** 5351–68
[37] Wang X Y, Zhao M C, Feng S J and Chen X 2023 An image encryption scheme using bit-plane cross-diffusion and spatiotemporal chaos system with nonlinear perturbation *Soft Comput.* **27** 1223–40
[38] Wang X and Gao S 2020 Image encryption algorithm based on the matrix semi-tensor product with a compound secret key produced by a Boolean network *Inf. Sci.* **539** 195–214
[39] Ding P F, Wang Z X and Li K 2024 Design and analysis of image encryption based on memristor chaotic systems with hidden attractors *Phys. Scr.* **99** 075252
[40] Hu X Y, Liu C X, Liu L, Ni J K and Li S L 2016 Multi-scroll hidden attractors in improved Sprott A system *Nonlinear Dyn.* **86** 1725–34
[41] Zhang S, Zeng Y C and Li J Z 2018 A novel four-dimensional no-equilibrium hyper-chaotic system with grid multiwing hyper-chaotic hidden attractors *J. Comput. Nonlinear Dynam.* **13** 090908
[42] Alanazi A S, Munir N and Khan M 2021 Cryptanalysis of novel image encryption scheme based on multiple chaotic substitution boxes *IEEE Access* **9** 93795–802

[43] Gao X Y, Sun B, Cao Y H, Banerjee S and Mou J 2023 A color image encryption algorithm based on hyperchaotic map and DNA mutation *Chinese Phys.* B **3** 131–42
[44] Mondal B and Singh J P 2022 A lightweight image encryption scheme based on chaos and diffusion circuit *Multimed. Tools Appl.* **81** 34547–71
[45] Zhou W, Wang X, Wang M and Li D 2022 A new combination chaotic system and its application in a new bit-level image encryption scheme *Opt. Lasers Eng.* **149** 106782
[46] Ghazvini M, Mirzadi M and Parvar N 2020 A modified method for image encryption based on chaotic map and genetic algorithm *Multimed. Tools Appl.* **79** 26927–50
[47] Mozaffari S 2018 Parallel image encryption with bitplane decomposition and genetic algorithm *Multimed. Tools Appl.* **77** 25799–819
[48] Zefreh E Z 2020 An image encryption scheme based on a hybrid model of DNA computing, chaotic systems and hash functions *Multimed. Tools Appl.* **79** 24993–5022
[49] Islam Y, Li C B, Sun K H and He S B 2024 Enhancing image security through an advanced chaotic system with free control and zigzag scrambling encryption *Multimed. Tools Appl* **83** 67327–55
[50] Lu Q, Yu L and Zhu C 2022 Symmetric image encryption algorithm based on a new product trigonometric chaotic map *Symmetry* **14** 373
[51] Vidhya R and Brindha M 2022 A novel approach for chaotic image encryption based on block level permutation and bit-wise substitution *Multimed. Tools Appl.* **81** 3735–72
[52] Cheng G, Wang C and Chen H 2019 A novel color image encryption algorithm based on hyperchaotic system and permutation diffusion architecture *Int. J. Bifurc. Chaos* **29** 1950115
[53] Hosny K M, Kamal S T and Darwish M M 2022 A color image encryption technique using block scrambling and chaos *Multimed. Tools Appl.* **81** 505–25
[54] Imran H, Ali A, Shah D and Shah T 2021 Block cipher's nonlinear component design by elliptic curves: an image encryption application *Multimed. Tools Appl.* **80** 4693–718
[55] Ayubi P, Setayeshi S and Rahmani A M 2020 Deterministic chaos game: a new fractal based pseudo-random number generator and its cryptographic application *J. Inf .Secur. Appl* **52** 1–20
[56] Xu J, Zhao B and Wu Z 2022 Research on color image encryption algorithm based on bit-plane and chen chaotic system *Entropy.* **24** 186
[57] Mondal B and Singh J P 2022 A lightweight image encryption scheme based on chaos and diffusion circuit *Multimed. Tools Appl* **81** 34547–71
[58] Zhou W, Wang X and Wang M 2022 A new combination chaotic system and its application in a new Bit-level image encryption scheme *Opt. Lasers Eng.* **149** 106782
[59] Alvarez G and Li S 2006 Some basic cryptographic requirements for chaos-based cryptosystems *Int. J. Bifurcat. Chaos* **16** 2129–51
[60] Hosny K M, Kamal S T and Darwish M M 2022 Novel encryption for color images using fractional-order hyperchaotic system *J Ambient Intell Human Comput* **13** 973–88
[61] Huang H and Cheng D 2022 A secure image compression-encryption algorithm using DCT and hyperchaotic system *Multimed. Tools Appl.* **81** 31329–47
[62] Chong J, Xie S, Zhang J and Liu D 2021 Block color image encryption algorithm based on elementary cellular automata and dna sequence operations *J. Electron. Imaging* **30** 043025
[63] Chen Y, Xie S and Zhang J 2022 A novel double image encryption algorithm based on coupled chaotic system *Phys. Scr.* **97** 065207
[64] Chen Y, Xie S and Zhang J 2022 A hybrid domain image encryption algorithm based on improved henon map *Entropy* **24** 287
[65] Yousif S F, Abboud A J and Alhumaima R S 2022 A new image encryption based on bit replacing, chaos and DNA coding techniques *Multimed. Tools Appl.* **81** 27453–93
[66] Mondal B and Singh J P 2022 A lightweight image encryption scheme based on chaos and diffusion circuit *Multimed. Tools Appl.* **81** 34547–71
[67] Sang J, Akbar M A, Cai B, Xiang H and Hu H 2018 Joint image compression and encryption using IWT with SPIHT, Kd-tree and chaotic maps *Appl. Sci.* **8** 1963
[68] Liang Q and Zhu C 2023 A new one-dimensional chaotic map for image encryption scheme based on random DNA coding *Opt. Laser Technol.* **160** 109033