Hindawi
Complexity
Volume 2021, Article ID 6683284, 21 pages
https://doi.org/10.1155/2021/6683284

Research Article

Chaos-Based Engineering Applications with a 6D Memristive Multistable Hyperchaotic System and a 2D SF-SIMM Hyperchaotic Map

Fei Yu,[1,2] **Shuai Qian**,[1] **Xi Chen**,[1] **Yuanyuan Huang**,[1] **Shuo Cai**,[1] **Jie Jin**,[3,4] **and Sichun Du**[5]

[1]*School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410114, China*
[2]*Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin 541004, China*
[3]*School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan 411201, China*
[4]*College of Information Science and Engineering, Jishou University, Jishou 416000, China*
[5]*College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China*

Correspondence should be addressed to Fei Yu; yufeiyfyf@csust.edu.cn

Received 14 December 2020; Revised 5 February 2021; Accepted 15 March 2021; Published 28 March 2021

Academic Editor: Chong Fu

In recent years, the research of chaos theory has developed from simple cognition and analysis to practical engineering application. In particular, hyperchaotic systems with more complex and changeable chaotic characteristics are more sensitive and unpredictable, so they are widely used in more fields. In this paper, two important engineering applications based on hyperchaos pseudorandom number generator (PRNG) and image encryption are studied. Firstly, the coupling 6D memristive hyperchaotic system and a 2D SF-SIMM discrete hyperchaotic mapping are used as the double entropy source structure. The double entropy source structure can realize a new PRNG that meets the security requirements, which can pass the NIST statistical test when the XOR postprocessing method is used. Secondly, based on the double entropy source structure, a new image encryption algorithm is proposed. The algorithm uses the diffusion-scrambling-diffusion encryption scheme to realize the conversion from the original plaintext image to the ciphertext image. Finally, we analyze the security of the proposed PRNG and image encryption mechanism, respectively. The results show that the proposed PRNG has good statistical output characteristics and the proposed image encryption scheme has high security, so they can be effectively applied in the field of information security and encryption system.

1. Introduction

With the rapid development of computer technology and communication technology, information has become an important resource in today's society, and the information security issues caused by it have become increasingly prominent [1–5]. In order to ensure the security of information, cryptographic technology is applied to information systems to achieve the confidentiality, integrity, availability, controllability, and nonrepudiation of information [6–9]. Random numbers (RNs) play an extremely important role in cryptography, such as generating parameters for public key cryptosystems (such as ECC and RSA), generating keys for symmetric cryptosystems (such as DES and 3DES), numerous cryptographic protocols, digital signatures, and identity authentication which need to use RNs. For this purpose, two types of generators are used: true random number generators (TRNGs) and pseudorandom number generators (PRNGs). The design method of TRNG is to obtain the natural physical random source in the integrated circuit by directly or indirectly sampling the noise on the resistor or MOS transistor or the phase jitter of the oscillator [10]. This kind of random number generator is often called a nondeterministic random number generator because the next number to be generated cannot be predetermined, and many TRNGs are relatively slow [11].

PRNs are widely used in image processing, the Internet of Things, and secure communications due to their

advantages such as fast generation, reusability, and relatively small storage memory required [12]. The rapid development of modern communication and information security technologies usually requires PRNs to have good randomness and complexity. Traditional PRNGs are mainly based on linear congruence theory, such as m-sequences and gold sequences. The complexity of these sequences is low, and there are security flaws in the application of information security, and the application in cryptographic design is limited by the speed of password generation. How to construct a good PRNG and generate high-performance and high-quality PRNs has always been a hot topic for scholars.

Chaos is widely used in complex networks [13–16], electronic circuits [17–20], image encryption [21–25], synchronous control [26–28], encryption system [29–32], and other fields because of its good random characteristics, extreme sensitivity to initial values and parameters, long-term unpredictability, and ergodicity of orbits. The PRNs based on the chaos system have the advantages of fast generation speed, high security performance, and good statistical characteristics. As a PRNG model with good performance, it has attracted more and more attention [33–37]. Chaotic systems are usually divided into continuous chaotic systems and discrete chaotic systems. Different chaotic systems exhibit different system dynamics, and the chaotic PRNs generated by them also have different random characteristics. The key to generating PRNs through chaos is often the selection of chaotic systems. The authors in [33, 34] designed PRNGs based on continuous chaotic systems: Chen chaotic system and generalized Lorenz chaotic system, respectively; The authors in [35, 36], respectively, designed PRNGs using discrete chaotic systems: Henon map and logistic map. It can be seen that the authors in [33–36] used such low-dimensional chaotic systems or one-dimensional chaotic maps to generate PRNs. These simple low-dimensional chaotic systems or one-dimensional chaotic maps can be attacked by using the nonlinear prediction method based on phase-space reconstruction [38, 39]. Therefore, a more ideal method is to generate PRNs directly by using high-dimensional chaotic systems. For chaotic systems, the more positive Lyapunov exponents, the better the randomness of chaotic systems and the higher the security of RNs generated based on such chaotic systems [38, 39]. Hyperchaotic systems have two or more positive Lyapunov exponents, and their orbits are separated in more directions, making them more difficult to predict and more complex in dynamic behavior than general chaotic systems [40–43]. Therefore, high-dimensional continuous or discrete hyperchaotic mapping system is the best choice for designing PRNGs. In order to overcome the disadvantage that the finite precision of the processor may lead to the degradation of the chaotic system into periodic functions or fixed points, the authors in [38] constructed a PRNG based on a hyperchaotic system with a large Lyapunov exponent. This method is better than other generators based on linear feedback shift register. In [44], a discrete hyperchaotic system was first designed by using piecewise linear state feedback. Then, a PRNG was designed by using three suchlike hyperchaotic systems with different feedback gain matrices. Through a threshold function, the three subsequences of the output of the piecewise linear function were transformed into 0-1 sequences. Then, by XOR operation, an unpredictable PRN was obtained. The analysis and simulation results show that the impulse response generated by the hyperchaotic system has good statistical characteristics.

In 1971, Chua [45] first proposed the memristor, which is a new type of two-port passive device. In 2008, HP developed the physical memristor based on titanium dioxide for the first time in the laboratory [46]. Since then, the memristor has been increasingly valued by academia and industry [47–51]. As a nonlinear part of chaotic systems, memristors can improve the randomness and complexity of signals in chaotic systems and reduce the physical size of systems. Many scholars have devoted themselves to the study of various memristor chaotic systems [26, 47]. Multistability is usually referred to as a coexistence of stable states or attractors, the stability of which depends on the speed at which the system returns to a certain state after perturbations that may be noise or even initial conditions. It has become a very hot research topic, and some significant research results have been achieved recently [52]. When the number of coexisting attractors generated by a chaotic system reaches infinity, the phenomenon of the coexistence of infinitely many attractors dependent on the initial conditions of state variables is called extreme multistability [41]. In fact, various systems have been proposed that exhibit extremely hidden multistability. However, a review of the literature revealed that no studies have examined this amazing behavior in autonomous systems with dimensions greater than five. In [53], a 6D memristive hyperchaotic system with hidden extreme multistability was constructed by introducing a flux-controlled memristor model into an existing 5D hyperchaotic autonomous system. Interestingly, for a particular set of parameters, an unusual metastable state showing the transition from chaos to periodic bursting dynamics was discovered. To our knowledge, PRNGs based on 6D memristive hyperchaotic autonomous systems with hidden extreme multistability are very rare in the literature. Therefore, it is of great significance to construct a PRNG based on 6D memristive hyperchaotic systems with hidden extreme multistability.

Based on Sine map and an iterative chaotic map with infinite collapse (ICMIC), a new high-dimensional hyperchaotic map called sinusoidal feedback Sine ICMIC modulation map (SF-SIMM) was proposed in [54]. The chaos performance of the 2D model of SF-SIMM was evaluated. The results shown that it had a complex phase-space trajectory, infinite equilibrium points, hyperchaotic behaviors, a fairly large maximum Lyapunov exponent, three typical bifurcations, and several coexisting attractors with odd symmetries. In addition, it had the advantages of complexity, distribution characteristics, and zero correlation and can produce two independent PRNs. Therefore, it has a good application prospect in PRNG.

Compared with text data, image data have the characteristics of large amount of data, strong data correlation, and large amount of redundant information [55–59]. This makes the traditional text-based cryptosystem no longer suitable

for the image encryption system [60–63]. The digital image encryption and decryption system based on the chaotic system can generate long enough key stream, which is very important for image pixel encryption. Recently, many image encryption algorithms based on chaotic systems have been proposed. Zhang [64] proposed a cipher block chain image encryption program based on AES, which was designed with C language. However, it is generally believed that AES is not suitable for image encryption. In [65], a new block image encryption scheme based on hybrid chaotic maps and dynamic random growth technique is proposed. Alawida et al. [66] used a hybrid system cascaded and combined with two chaotic maps as a new dual entropy source chaotic system. By disturbing the chaotic state and system parameters, pixel scrambling and substitution operations are carried out, respectively, to obtain the chaotic characteristics and diffusion characteristics of the chaotic system.

The purpose of this paper is to realize two important engineering applications based on hyperchaos—PRNG and image encryption. The rest of the paper is organized as follows. In the second section, the mathematical models and dynamic characteristics of a 6D memristive hyperchaotic autonomous system and a 2D SF-SIMM hyperchaotic mapping are listed, respectively. The third section introduces the postprocessing process of binary quantization of the chaotic system with double entropy sources and the statistical test results of NIST. In the fourth section, the security performance of the proposed PRNG algorithm is analyzed. In the fifth section, the diffusion-scrambling-diffusion image encryption scheme based on the double entropy source chaos is adopted. Finally, the conclusion is drawn in the sixth section.

2. System Description

2.1. 6D Memristive Hyperchaotic System. A 6D memristive hyperchaotic autonomous system [53] with complex and implicit extreme multistability has the following dynamic phenomena on a line or an equilibrium plane: hidden extreme multistability, transient chaos, bursting, and offset boosting phenomenon. This 6D memristive hyperchaotic system is the first high-order system to present all these rich dynamic behaviors:

$$\begin{cases} \dot{x}_1 = \alpha\left(1 - \beta|x_6|\right)x_2 - ax_1, \\ \dot{x}_2 = cx_1 + dx_2 - x_1x_3 + x_5, \\ \dot{x}_3 = -bx_1 + x_1^2, \\ \dot{x}_4 = ex_2 + fx_4, \\ \dot{x}_5 = -rx_1 - kx_5, \\ \dot{x}_6 = -x_2. \end{cases} \tag{1}$$

With the memristor model which is described as $\omega(\varphi) = 1 - \beta|\varphi|$, φ and β are flux variable and positive constant parameter, respectively.

In the 6D memristive hyperchaotic system given in [53], $x_1, x_2, x_3, x_4, x_5, x_6$ are the state variables and $\alpha, \beta, a, b, c, d, e, f, r, k$ are system parameters. When initial point $(x_1(0), x_2(0), x_3(0), x_4(0), x_5(0), x_6(0)) = (0.05, 0, 0.2, 1, 0, 0.5)$, the parameters are chosen as $\alpha = 16, \beta = 0.2, a = 6.5, b = 1.5, c = 5.5, e = 5, f = 0.01, r = 5$, and $k = 0.05$, especially, when $d = -1.122$ and $d = -1.964$, the signs of the Lyapunov exponents (LE1, LE2, LE3, LE4, LE5, LE6) are $(+, +, +, -, -, -)$, respectively. Thus, system (1) showed hyperchaotic states. When $f \neq 0$, it is calculated that system (1) has a line equilibrium $O(0, 0, 0, 0, 0, l)$, l can be any real constant and a plane of equilibrium as well.

Similarly, good research of hidden extreme multistability has been done in [53]. When the system parameters and some initial values are fixed and $x_1(0)$ and $x_6(0)$ are changed at the same time, it is easy to find that the system has hidden and multistable attractors. Some typical attractors obtained for different values of $x_6(0)$ are described in detail in Figure 1. When $x_6(0) = -1.80$ and $x_6(0) = -0.72$, the $x_1 - x_2$ plane of system (1) presents two cycle-2 limit; when $x_6(0) = -0.02$ and $x_6(0) = 0.36$, the $x_1 - x_2$ plane shows twin chaotic attractor. All of the above are discussed under
$\alpha = 15.47, \beta = 0.12, a = 4.8, b = 0.18, c = 8.5, d = -0.1, e = 1, f = -0.1, r = 0.1$, and $k = 0$, and $x_1(0) = 0.1$, $x_2(0) = 0$, $x_3(0) = 0.01$, $x_4(0) = 0.05$, and $x_5(0) = 0.07$.

2.2. 2D SF-SIMM Hyperchaotic Map. The existing chaotic maps can be divided into two categories: one-dimensional (1-D) and high-dimensional chaotic maps. Based on sinusoidal mapping and radio folding mapping, a new 2D hyperchaotic mapping sinusoidal feedback modulation mapping is proposed in [54] (based on closed-loop modulation coupling mode):

$$\begin{cases} y_1(n+1) = m\ \sin\left[\omega y_2(n)\right]\sin\left[\dfrac{n}{y_1(n)}\right], \\ y_2(n+1) = m\ \sin\left[\omega y_1(n+1)\right]\sin\left[\dfrac{n}{y_2(n)}\right], \end{cases} \tag{2}$$

where y_1 and y_2 are the state variables and m, n, and ω are system parameters, and $m, n, \omega \in (0, +\infty)$. m of the system is the amplitude. ω is the frequency. n is the internal perturbation frequency. In addition, in the case of $m = 1, n = 3$, and $\omega = \pi$, system (2) is hyperchaotic. Figures 2(a)–2(c) are the attractor phase diagrams for $m = 1$, $n = 3$; $m = 2$, $n = 5$; and $m = 3$, $n = 5$, respectively. When $m = 1, n = 3$, and $\omega = 1$, there is no equilibrium point in the system; when the value of ω increases to $\omega = 1.01$, the system has infinite equilibrium. According to the research in [54], when $m\omega \leq 1$, system (2) has no equilibrium point; when $m\omega > 1$, the system has infinite equilibrium point. Through dynamic analysis, 2D SF-SIMM has high complexity, uniform distribution, and zero correlation in the whole parameter range.

3. PRNG Algorithm and NIST Test

3.1. The Structure of PRNG Algorithm. In this section, we proposed and analyzed an algorithm for generation the pseudo-random binary sequence based on the 6D

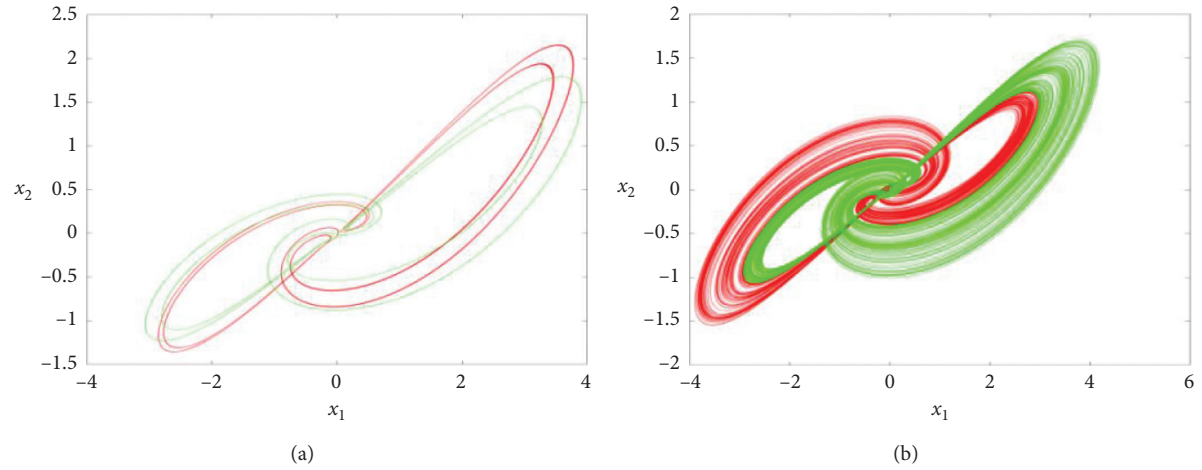

FIGURE 1: The typical attractors in the $x_1 - x_2$ plane with different values of $x_6(0)$: (a) cycle-2 limit attractor; (b) chaotic coexistence attractor.

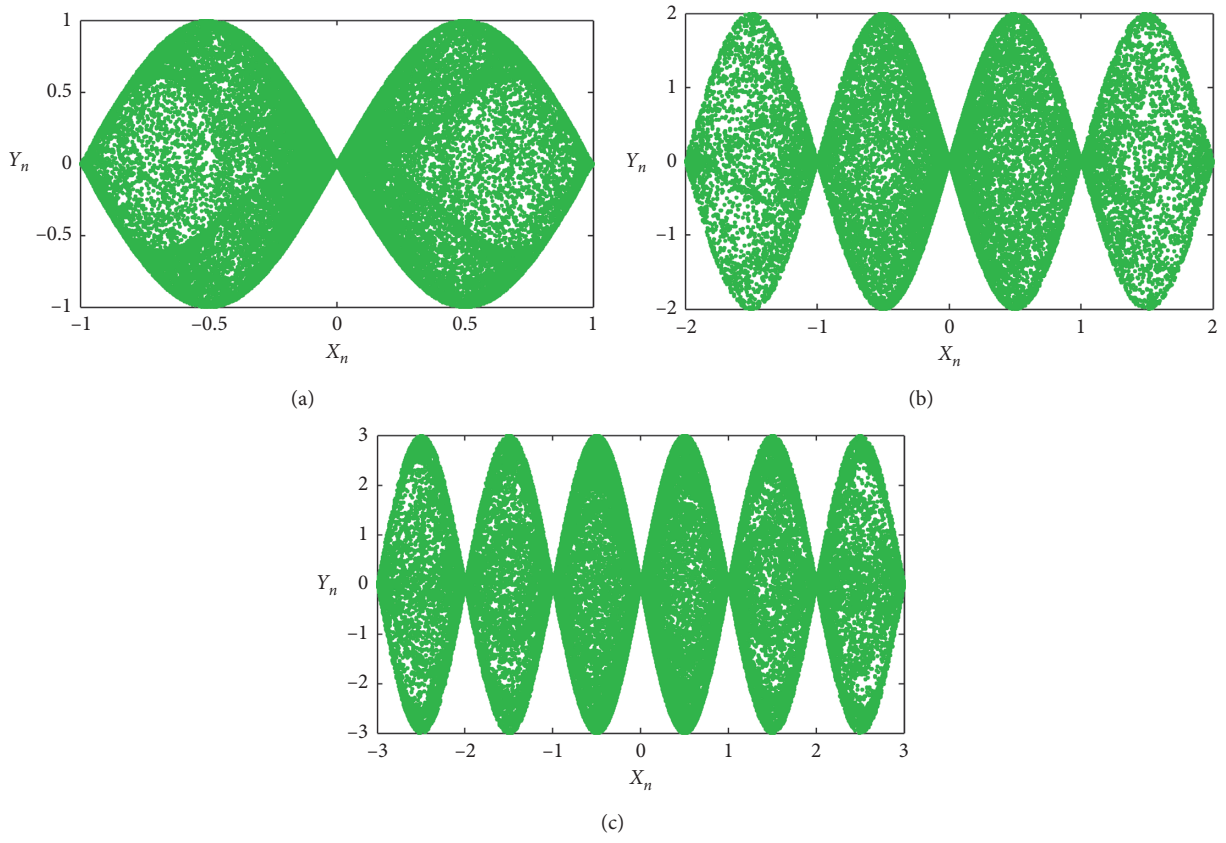

FIGURE 2: Chaotic attractors for 2D SF-SIMM systems with different parameters: (a)$m = 1, \omega = \pi, n = 3$; (b)$m = 2, \omega = \pi, n = 5$; (c).$m = 3, \omega = \pi, n = 5$.

memristive hyperchaotic system and the 2D SF-SIMM hyperchaotic map. In other words, we generated more random binary sequences on the system with the dual entropy cores chaotic system (DECHCS), which is more suitable for cryptography.

Generally, the algorithm of generating pseudorandom number is interesting because when we do not change the seed, we can guarantee the complete reproduction of pseudorandom sequence. This can be important in mock code.

The PRNG algorithm is divided into two stages: initialization and operation. The initialization phase is only used to set the initial state and system parameters of PRNG based on DECHCS. However, the attacker is not interested

in the initialization phase, so the description of the analyzed PRNG will only focus on the working phase. This phase will directly affect the complexity and randomness of the generated sequence and ultimately affect the security of its application.

The PRNG algorithm is performed by following steps.

Step 1. Initialization of initial value and system control parameters for DECHCS.

Step 2. The initial value and the control parameters of the system are substituted into the double entropy kernel system, and the required sequence length is obtained by many times of iteration.

Step 3. In order to eliminate the short periodicity and improve the random performance of the generated sequences, we proposed the continuous memristor hyperchaos system and SF-SIMM are mapped with equal probability distribution XOR. In order to obtain better random security, we remove the dimension sequence which does not conform to the security in the hyperchaotic system with high memory resistance.

Step 4. Repeat step 3 until sufficient sequence length is obtained.

The pseudorandom sequence generator designed according to the above steps has better pseudorandom characteristics compared with the sequences generated by the general binary quantization algorithm, and the sequences generated by the pseudorandom generator designed based on the DECHCS have higher security. The quantization algorithm is based on the DECHCS, that is, the combination of six chaotic orbit coordinates and two mapped orbits, which can ensure the security of the pseudorandom sequence generator. MATLAB is used to calculate the simulation.

3.2. Randomness Tests. The analysis of the randomness of pseudorandom sequences is an important content of cryptography security research. A large number of randomness testing algorithms and related standards can be used to evaluate the generated pseudorandom sequences, which can provide a lot of reference data for theoretical analysis. According to the standard SP 800-22 issued by National Institute of standards and technology (NIST 800.22) [67], 16 statistical testing methods of random testing are recommended. Later, Germany issued the BSI AIS-30 specification on the basis of NIST specification. Some commonly used ones include the Federal information processing standard (FIPS 140.1) and the Diehard test suite. NIST statistical test suite contains a sufficient number of almost independent statistical tests, which is the most stringent and current industry standard for random testing.

NIST suite testing requires a binary sequence of at least 10^6 bits to detect potential defects in the proposed pseudorandom sequence generator architecture. The significance level results of each test were shown as P value. When P value ≥ 0.01, we consider the sequence to be random with a confidence level of 0.99. In this experiment, we generate 100 different binary sequences with a length of 1000000 bits using the pseudorandom sequences generated by the hyperchaotic system based on double entropy sources memristor. The results are shown in Table 1. It can be seen that the binary sequence generated by our method has good randomness and statistical characteristics and has passed all test suites.

4. Security Analysis

4.1. Weak Key. The main security problem of chaotic cryptosystem is the relationship between the key and the control parameters. The security of encryption scheme depends on the confidentiality of key, not on the security of algorithm. The bifurcation diagram is used to represent the dynamic properties of the chaotic system, that is, the system characteristics under the key or control parameters. For the bifurcation diagram of a parameter, a few or no plotting points in some regions indicate the existence of fixed points or short period chaotic phenomena. In some part of the bifurcation graph, a large number of plotting points cover the black area, which shows the chaotic behavior. And this part of the bifurcation diagram is called the black area. In order to eliminate the weak key, we should select the key in the black area of the bifurcation diagram [11].

The bifurcation diagram of the 6D memristive hyperchaotic system with respect to the value range of d in $(-3, 0)$ is shown in Figure 3(a). The bifurcation scenarios mainly occur through abrupt changes from chaos/hyperchaos to period 2. When the parameter $d \in (-3, 0)$ of the 6D memristive hyperchaotic system increases, the black area of the bifurcation diagram decreases. When $d \in (-0.72, -0.2)$, period 2 appears, and most of the bifurcation diagrams are white. Similarly, in Figure 3(b), the Lyapunov exponent diagram of the 6D memristive hyperchaotic system is given. When the parameter d increases, the value of Lyapunov index decreases.

There are many parameters in the high-dimensional chaotic system, so in [53], all the values of $c \in [2.5, 5.5]$ can be used as part of the key. However, most of $c \in (2.5, 3.3)$ and $c \in (5.2, 5.5)$will reduce the security of the analyzed PRNG, as shown in Figure 4. Therefore, the output value x_i^j of the analyzed PRNG (after the first iteration) depends partly on d and c. Therefore, if the precision of 10–15 is used, the security of using the analytical PRNG only reduces to $10^{15} \times 10^{15} \approx 2^{100}$ when only the parameters d and c are considered. These two security levels are far lower than the recommended 2128 to resist exhaustive attacks, so we can consider all keys containing the values of $d \in (-0.72, -0.2)$, $c \in (2.5, 3.3)$, and $c \in (5.2, 5.5)$to be weak keys.

4.2. Key Space Analysis. In order to protect the confidentiality of information and resist cryptanalysis, the size of key space is very important. The larger the key space, the higher the encryption strength and the more suitable for information encryption. 1D continuous space chaotic map has relatively small key space, which is not desirable in cryptography [68]. Otherwise, if the key space is too small, it will

Table 1: The results of NIST test suite.

Statistical test	Passing ratio	P value	Results
Frequency	0.991	0.982586	Success
Block frequency	0.995	0.176868	Success
Cumulative sums	0.991	0.587454	Success
Runs	0.987	0.457579	Success
Longest run	0.995	0.873851	Success
Rank	1	0.713552	Success
FFT	0.983	0.686776	Success
Nonover lapping template	1	0.774162	Success
Over lapping template	0.969	0.050798	Success
Universal	0.995	0.982586	Success
Approximate entropy	0.991	0.343492	Success
Random excursions	1	0.689019	Success
Random excursions variant	1	0.159401	Success
Serial	0.995	0.164583	Success
Linear complexity	0.995	0.659741	Success

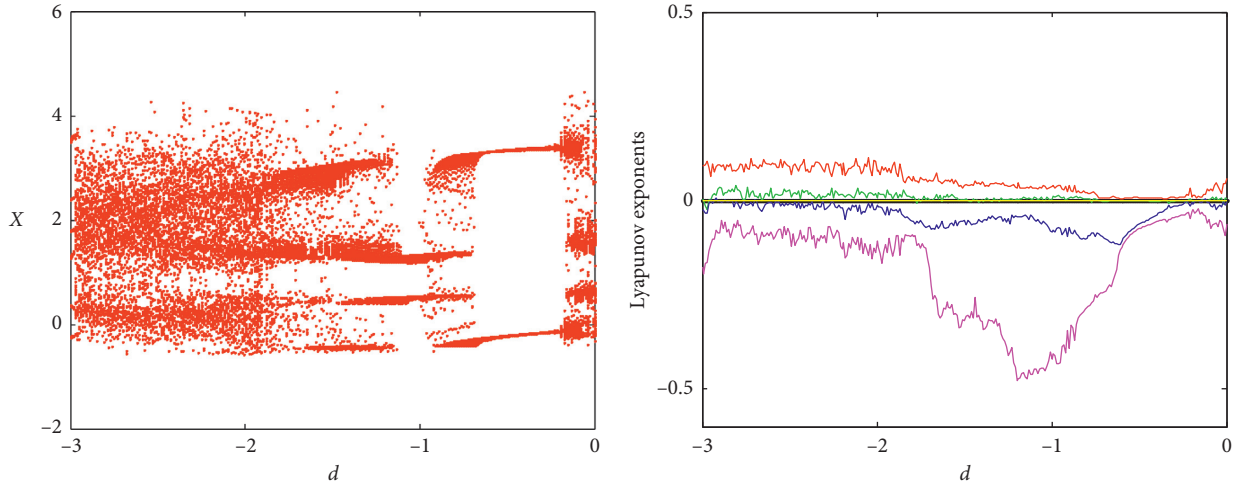

Figure 3: (a) Bifurcation diagram versus control parameter d showing the local maxima of the variable x_1 and (b) its corresponding Lyapunov exponent spectrum with $c = 4.25$ and $d \in [-3, 0]$. Other parameters are $(m, n, a, b, e, f, k, r) = (16, 0.2, 4.8, 0.28, 1, 0, 0.1, 1)$ and initial conditions are $(x_1(0), x_2(0), x_3(0), x_4(0), x_5(0), x_6(0)) = (0.05, 0, 0.2, 1, 0, 0.5)$.

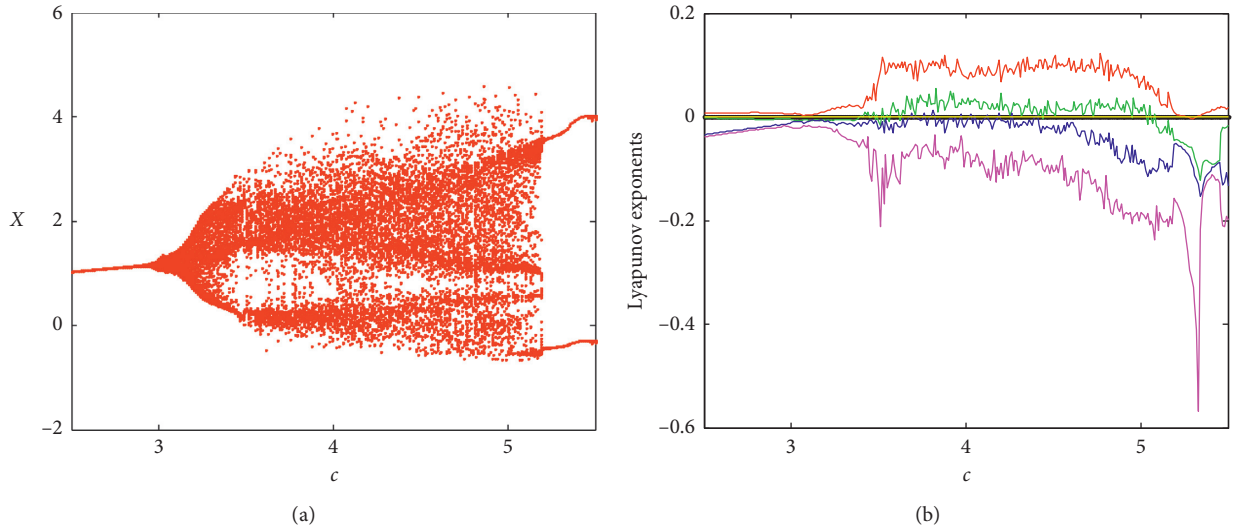

Figure 4: (a) Bifurcation diagram versus control parameter c showing the local maxima of the variable x and (b) its corresponding Lyapunov exponent spectrum with $d = -2.44$ and $c \in [2.5, 5.5]$. Other parameters are $(m, n, a, b, e, f, k, r) = (16, 0.2, 4.8, 0.28, 1, 0, 0.1, 1)$ and initial conditions are $(x_1(0), x_2(0), x_3(0), x_4(0), x_5(0), x_6(0)) = (0.05, 0, 0.2, 1, 0, 0.5)$.

be vulnerable to brute-force attack, and the key password will be cracked.

As a standard attack, brute-force attack can be used for any bloc password. The attack method usually enumerates all possible keys based on certain policies and rules until the correct key is found. In order to resist violent attacks, the size of the key space must be large. Generally speaking, it is not secure when the key space is less than 2^{128}.

In [69], the authors presented a PRNG algorithm based on two chaotic maps which enables to produce about 2_{213} pseudorandom sequences. Then, in 2015, Stoyanov and Kordov [70] proposed to construct a PRNG based on two Tinkerbell maps and obtained that the key space of the system is 2^{183}. In the same year, García-Martínez and Campos-Cantón [71] proposed a cryptographic secure pseudorandom number generator (CSPR) based on the multimodal discrete system, which is called k-modal mapping (based on logistic mapping), and through analysis, the key space is 2^{159}. In [72], a new pseudorandom enhanced logic map (PLEM) is proposed. The key space of the system is 2^{128}. Based on [36], the authors proposed a PRNG algorithm based on piecewise logistic map (PLM), which is an optimized version of logistic map. The security and efficiency of PRNG are analyzed. The key space of PLM is 9.14×10^{40}.

In this paper, a 6D continuous memristive hyperchaotic system and 2D discrete mapping are used to construct PRNG to increase the required key space. The key space is a collection of all possible keys that can be used for the initial seed of a pseudorandom scheme. The high-dimensional chaotic system has many parameters and is sensitive to boundary conditions and initial values of the system, and the relative key space is also large.

In most PRNGs using continuous space chaotic maps, the key space depends on the precision of floating-point numbers. However, in discrete mapping, the situation is very different. According to IEEE floating-point operation standard, the key consists of 16 double precision floating-point numbers of initial condition $\{x_1, x_2, x_3, x_4, x_5, x_6\}$ and system parameter $\{a, b, c, d, e, f, r, k, m, n\}$ of the 6D continuous memristive hyperchaotic system and five double precision floating-point numbers of initial condition $\{y_1(n+1), y_2(n+1)\}$ and mapping parameter m, n, and ω of discrete 2D SF-SIMM hyperchaotic map. In other words, the key space of this method is 2^{315} (for precision of 10^{-15}), which is much larger than 2^{128}, so it can effectively resist to make brute-force attack/exhaustive attack.

4.3. Key Sensitivity Analysis. For the sensitivity analysis of the binary sequence based on the double entropy kernel hyperchaotic system, the bit change rate of the two generated sequences can be calculated by changing the initial key of the system slightly. The greater the change, the better the sensitivity. In this test, we give an original key as the benchmark key to generate a pseudorandom sequence with a length of 120000 bits. We change the initial condition $x_1(0)$ and parameter α_1 of the chaotic system slightly to get a very close initial value, namely, $x_2(0) = x_1(0) + \Delta(0)$ and $\alpha_2 = \alpha_1 + \Delta(0)$. If $|\Delta(0)| \longrightarrow 0$, the initial value of iteration changes exponentially, which reflects the dependence of the chaotic system on initial value.

(1) The original sequence T_1 is generated when the initial condition is $\alpha = 16$, and then a new sequence T_1' is generated by slightly modifying the initial condition $\alpha' = 16 + 10^{-12}$

(2) The original sequence T_2 is generated when the initial condition is $x(0) = -1$, and then a new sequence T_2' is generated by slightly modifying the initial condition $x'(0) = 0.05 + 10^{-12}$

In this test, the length of sequences generated by each system is $N = 1000000$. The bit change rate can be used to measure the sensitivity of the PRNG to the key, that is, to observe the different degrees of the number of bits in the sequence generated by the PRNG when the key is changed slightly. The ideal bit rate of change is 50%. The closer the bit change rate is to 50%, the better the sensitivity of the PRNG is to the initial value. Let the length of sum of two pseudorandom sequences T_i and T_i' with different initial values be N, $i = 1, 2, 3, \ldots, N$. Then, the corresponding bit rate of change is defined as follows:

$$P = \frac{\sum_{i=1}^{n} (T_i - T_i')}{N} \times 100\%. \tag{3}$$

Respectively, the variation of bit rate P with initial value and parameter variation $\Delta i\,(i = x, \alpha)$ is shown in Table 2. It can be seen that the bit change rate of pseudorandom sequence is very close to the ideal 50% when the initial value and parameters of the system change only 10^{-12}, which shows that the system is extremely sensitive to the initial conditions and system parameters.

Figures 5(a) and 5(b) show the time-domain waveforms of the variable parameter a and the initial value $x(0)$ of the dual entropy kernel chaotic system before and after minor changes. The red track represents the original output of the system, and the blue track represents the output after minor changes to the initial state. It is shown in both figures that after about 200 iterations, the trajectory of the system is separated, and the degree of separation becomes more and more obvious with the increase in the number of iterations. Therefore, we can see that the sensitivity of initial value becomes more and more obvious with the increase in iteration times.

4.4. Correlation Analysis. The autocorrelation and cross-correlation of sequences are an important means to measure the randomness of two sequences generated by adjacent keys. Good correlation is one of the guarantees for the reliable operation of the system. To further verify the sensitivity of the design method to the initial key, the correlation between the two sequences generated by the similar key is observed. If the generated sequence is ideal and random, its autocorrelation graph is δ function, and the cross-correlation graph should be all zero. Therefore, using the proposed algorithm (checking the generated sequences with the neighbor key), the

Table 2: Correlation coefficient of four cases.

Case		Correlation coefficient
$S_1 \longrightarrow s_{01} = 16$	$S_1' \longrightarrow s_{01}' + \delta = 16.000000000001$	4.0954×10^{-7}
$S_2 \longrightarrow s_{02} = 0.05$	$S_2' \longrightarrow s_{02}' + \delta = 0.050000000001$	7.7323×10^{-4}
$S_3 \longrightarrow s_{03} = 4.8$	$S_3' \longrightarrow s_{03}' + \delta = 4.800000000001$	-1.2085×10^{-4}
$S_4 \longrightarrow s_{04} = 0.2$	$S_4' \longrightarrow s_{04}' + \delta = 0.200000000001$	-9.0471×10^{-7}

Table 3: Initial value sensitivity analysis of PRNG.

	$\Delta\alpha$	$\Delta x(0)$	P (%)
T_1'	10^{-12}		49.999
T_2'		10^{-12}	50.024

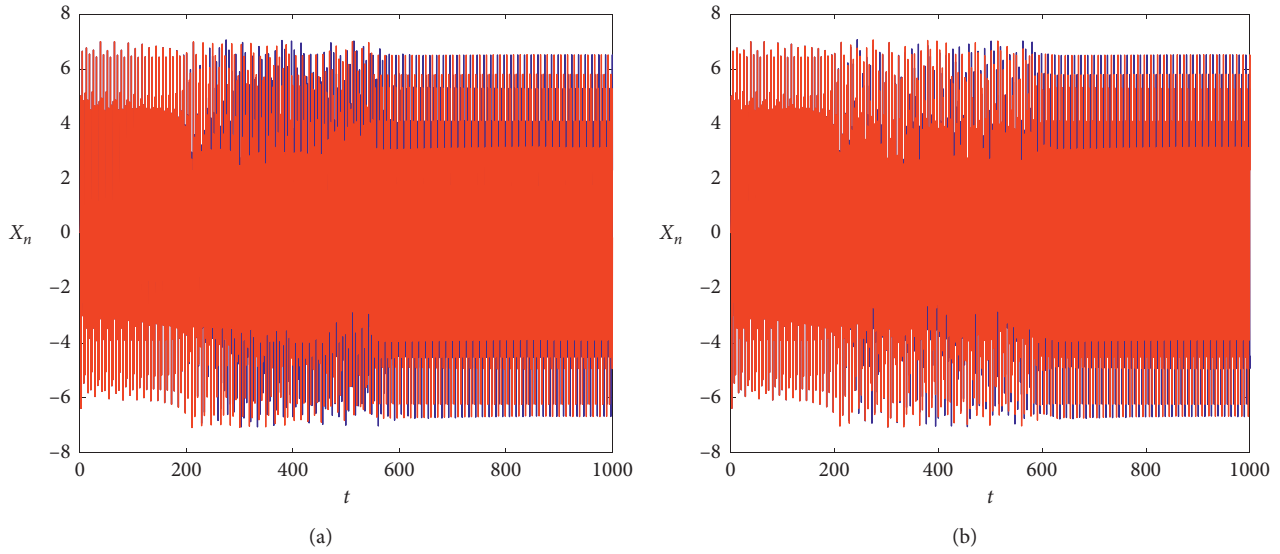

Figure 5: Time-domain waveform: (a) the change of variable parameter a; (b) the change of variable parameter $x(0)$.

correlation coefficients of $S = \{s_0, s_1, \ldots, s_{n-1}\}$ and $S' = \{s_0', s_1', \ldots, s_{n-1}'\}$ for each pair of sequences can be expressed as follows:

$$r_{SS'} = \frac{\sum_{i=0}^{N} (s_i - E(S)) \cdot (s_i' - E(S'))}{\sqrt{\sum_{i=0}^{N} (s_i - E(S))^2} \cdot \sqrt{\sum_{i=0}^{N} (s_i' - E(S'))^2}}, \quad (4)$$

where $E(S) = (1/N) \sum_{i=0}^{N} s_i, E(S') = (1/N) \sum_{i=0}^{N} s_i', N$ is the length of the sequence $r_{SS'} \in (-1, 1)$. If it keeps $r_{SS'} = 0$, then we can assume that there is zero correlation between S and S'. Therefore, the sensitivity of the proposed hyperchaotic system with double entropy sources to the small changes of initial values and system parameters is very high. In Figure 6, the autocorrelation graph and cross-correlation graph between the pseudorandom sequence generated by the original key and the four pseudorandom sequences generated by four randomly selected keys are given.

Based on the above analysis, the chaotic sequence with double entropy sources has the similar property of δ – like, that is, the autocorrelation function has sharp correlation peak, and the cross-correlation peak value is close to 0. It can be seen from Table 3 that the correlation coefficient between 120000 pseudorandom number sequences generated by the four experiments is very small, and the uniform results close to 0 verify that there is almost no correlation between the pseudorandom sequences generated by this method.

4.5. Spectral Entropy Complexity. There are many algorithms to calculate the behavior complexity of chaotic pseudorandom sequences. In this paper, the spectral entropy complexity algorithm (S-E complexity) is used to analyze the complexity of chaotic pseudorandom sequences. In this paper, we discuss the system parameters and complexity of the hyperchaotic system with dual entropy sources. The complexity of the parameters m and a of the hyperchaos system with double entropy source is discussed. As shown in Figure 7, the variation range of m is $[-10, 0]$, and the step value is 0.01; the variation range of a is $[-1, 1]$, and the step value is 0.01. As can be seen from Figure 6, more than 98% of the displayed interval is in a chaotic state. When $m \in [0.5, -10]$ and $a \in [-1, 1]$, the hyperchaotic system with double entropy sources is in a high complexity region where the maximum Lyapunov exponent is larger. This shows that the fluctuation range of the system is relatively small when

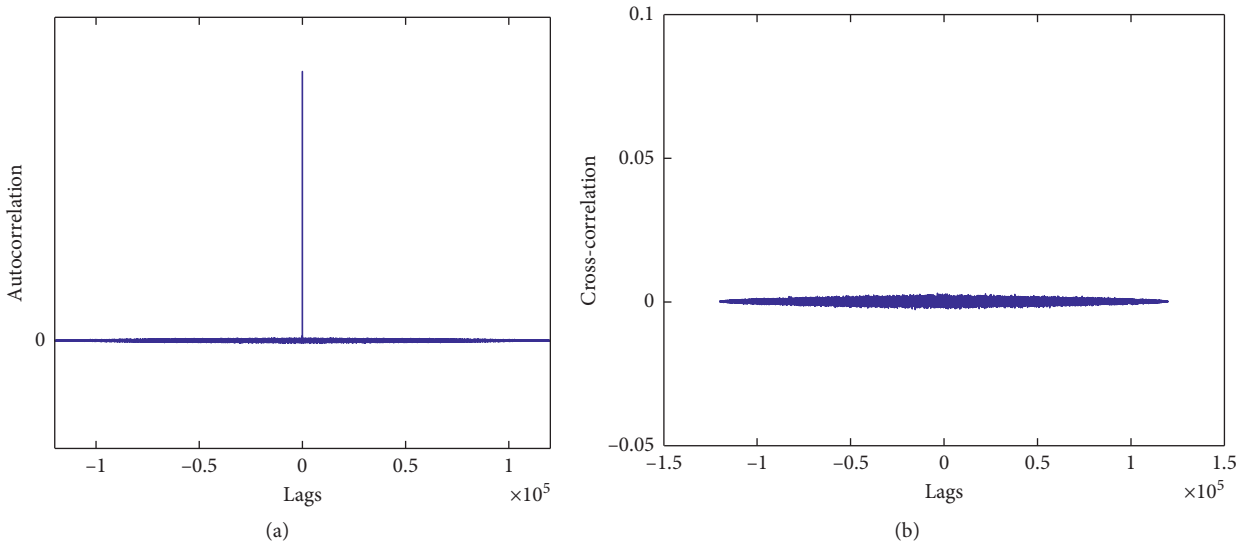

Figure 6: Correlation analysis of 4 experiments: (a) autocorrelation; (b) cross-correlation.

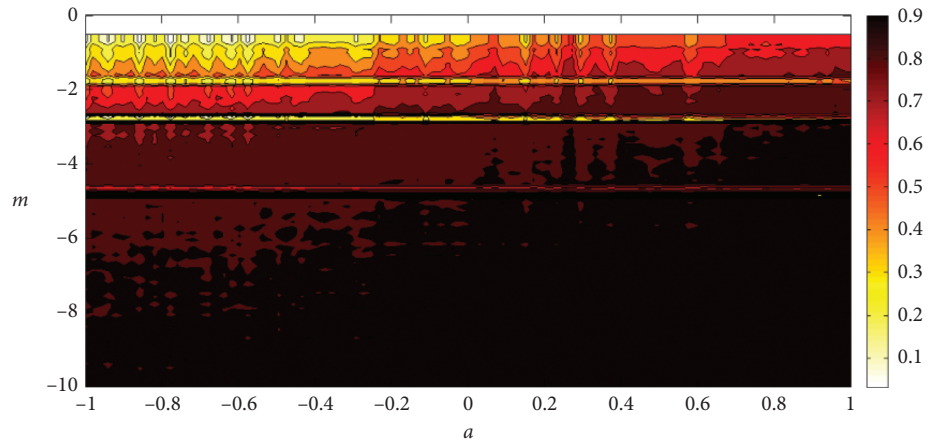

Figure 7: S-E complexity in the $m - a$ parameter plane.

the initial conditions fluctuate in a certain range, which also shows the rich dynamic characteristics of the proposed system.

4.6. Comparative Analysis. As researchers are more and more interested in the chaotic system, PRNG research based on the chaotic system is more and more extensive. In order to evaluate the superiority of our proposed method, we will focus on the security and compare with the latest schemes. The comparison results are shown in Table 4.

5. Image Encryption

For some special fields, such as military, commercial, and medical, the sender and receiver are required to communicate according to high security standards in the transmission process of digital images, so as to ensure the integrity, reliability, and security of digital images. The purpose of image encryption is to change the position or value of pixels. Fridrich applied chaos theory to image encryption for the first time, which has gradually become a hot research topic. In this section, we give a simulation based on our proposed encrypted color image. The scrambling-diffusion-scrambling structure will be adopted to change the spatial and gray distribution of image pixels and improve the robustness of image encryption technology.

The specific operation of encryption are performed in the following five steps:

Step 1. The double entropy source hyperchaotic system is iterated. In order to ensure the randomness of the sequence, the previous $r_1 + r_2$ iterations are omitted and the transition states of the hyperchaotic system with double entropy kernel are skipped. Continue to iterate $m \times n$ times to get 8 pseudorandom sequences, which are denoted $\{x_{1,i}\}, \{x_{2,i}\}, \{x_{3,i}\}, \{x_{4,i}\}, \{x_{5,i}\}, \{x_{6,i}\}, \{y_{1,i}\}, \{y_{2,i}\}$, $i = 1, 2, \ldots, M * N$ in turn.

Table 4: Comparative analysis of PRNG based on the DECHCS and the latest articles.

Comparison item	Proposed PRNG	Ref. [11]	Ref. [12]	Ref. [36]	Ref. [39]
Key space	2^{315}	2^{262}	4.29×10^{76}	9.14×10^{40}	2^{256}
Key sensitivity	Yes	Yes	Yes	Yes	Yes
Correlation coefficient	-9.0471×10^{-7}	0.009	0.025	0.025	1.98×10^{-4}
Entropy analysis	Yes	NA	NA	NA	NA
NIST test	Pass	Pass	Pass	Pass	Pass

Step 2. Take six pseudorandom sequences and generate matrices X, Y, Z, W, U, and V in turn with the following formula:

$$\begin{aligned}
X(k,l) &= \text{floor}\left(\left(x_{1,(k-1)\times N+l} + 500\text{mod}1\right) \times 10^{13}\right)\text{mod}2^{L},\\
Y(k,l) &= \text{floor}\left(\left(x_{2,(k-1)\times N+l} + 500\text{mod}1\right) \times 10^{13}\right)\text{mod}2^{L},\\
Z(k,l) &= \left(\text{floor}\left(x_{3,(k-1)\times N+l} \times 10^{13}\right)\text{mod}M\right) + 1,\\
W(k,l) &= \left(\text{floor}\left(\left(x_{4,(k-1)\times N+l} + 500\text{mod}1\right) \times 10^{12}\right)\text{mod}N\right) + 1,\\
U(k,l) &= \left(\text{floor}\left(\left(x_{5,(k-1)\times N+l} + 500\text{mod}1\right) \times 10^{12}\right)\text{mod}M\right) + 1,\\
V(k,l) &= \left(\text{floor}\left(\left(y_{1,(k-1)\times N+l} + y_{2,(k-1)\times N+l} + 500\text{mod}1\right) \times 10^{12}\right)\text{mod}N\right) + 1,
\end{aligned} \tag{5}$$

$k = 1, 2, \ldots, M$, $l = 1, 2, \ldots, N$. The positive integer matrix is obtained.

Step 3. By increasing j, the plaintext image P (size is $M \times N$) is transformed into matrix A according to the integer pseudorandom matrix X. The specific diffusion method is shown in Figure 8.

$$A(i,j) = P(i,j) + X(i,j) + r_1 \text{mod}2^{L}, \tag{6}$$

$$A(i,j) = P(i,j) + A(i,j-1) + X(i,j)\text{mod}2^{L}, \tag{7}$$

$$A(i,j) = P(i,j) + \text{sum}\left(A(i-1, 1 \text{ to } N)\right) + X(i,j)\text{mod}2^{L}. \tag{8}$$

Step 4. Image A is scrambled to generate image B by disturbing the correlation of adjacent pixels in pixels. For the coordinates (i, j) of any pixel in image A, the values of (m, n) are obtained by using the following formulas:

$$m = \left(U(i,j) + \text{sum}\left(A(Z(i,j), 1 \text{ to } N)\right)\text{mod}M\right) + 1, \tag{9}$$

$$n = \left(V(i,j) + \text{sum}\left(A(1 \text{ to } M, W(i,j))\right)\text{mod}N\right) + 1. \tag{10}$$

When $m = i$ or $Z(i, j)$, or $n = j$ or $W(i, j)$, or $Z(i, j) = i$, or $W(i, j) = j$, the position of $A(i, j)$ remains unchanged; otherwise, $A(i, j)$ and $A(m, n)$are interchanged. In this way, all the pixels in image A are traversed from left to right, from top to bottom, and converted to image B.

Step 5. In order to make the spatial distribution and gray distribution of image pixels more uniform, we will transform image B into matrix C with the help of integer pseudorandom matrix Y, and matrix C is the ciphertext image obtained. The difference between the diffusion algorithm and diffusion algorithm I is that it spreads forward from the last pixel of the image. The specific diffusion method is shown in Figure 9.

$$C(i,j) = B(i,j) + Y(i,j) + r_2 \text{mod}2^{L}, \tag{11}$$

$$C(i,j) = B(i,j) + C(i,j+1) + Y(i,j)\text{mod}2^{L}, \tag{12}$$

$$C(i,j) = B(i,j) + \text{sum}\left(C(i+1, 1 \text{ to } N) + Y(i,j)\right)\text{mod}2^{L}. \tag{13}$$

The decryption process is similar to the encryption process, which is the reverse process. The specific encryption and decryption process is shown in Figure 10.

In this simulation experiment, we give the correct key: $(x_1(0), x_2(0), x_3(0), x_4(0), x_5(0), x_6(0), r_1, r_2, y(0), y_2(0)) = (0.05, 0, 0.2, 1, 0, 0.5, 35, 201, 1, 1)$ and wrong key: $(x_1(0), x_2(0), x_3(0), x_4(0), x_5(0), x_6(0), r_1, r_2, y_1(0), y_2(0)) = (0.050000000001, 0, 0.2, 1, 0, 0.5, 35, 201, 1, 1)$, which means that there is a small difference between the correct key and the wrong key. Figure 11 shows the relevant results, where Figures 11(a) and 11(b) show the original plaintext image (Lena has a small amount of detail and Baboon has a medium level of detail). Figures 11(c) and 11(d) show a ciphertext image encrypted by a correct key. Figures 11(e) and 11(f) show the decrypted image with the correct key and

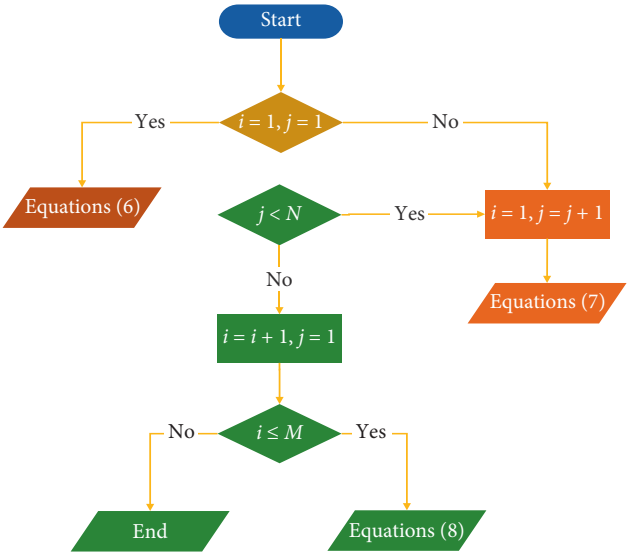

Figure 8: Block diagram of diffusion algorithm I.

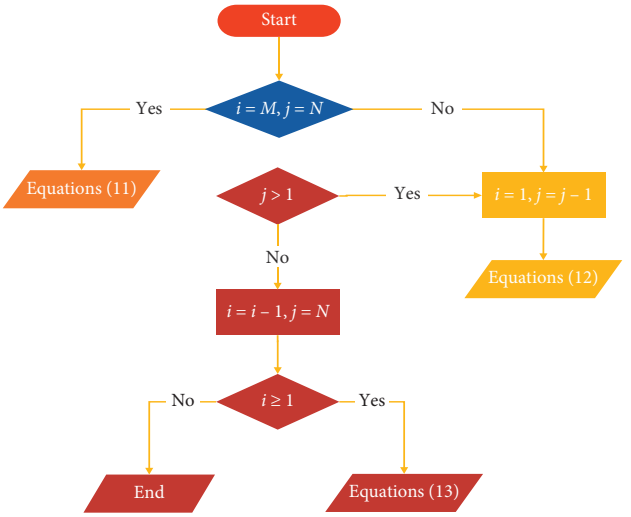

Figure 9: Block diagram of diffusion algorithm II.

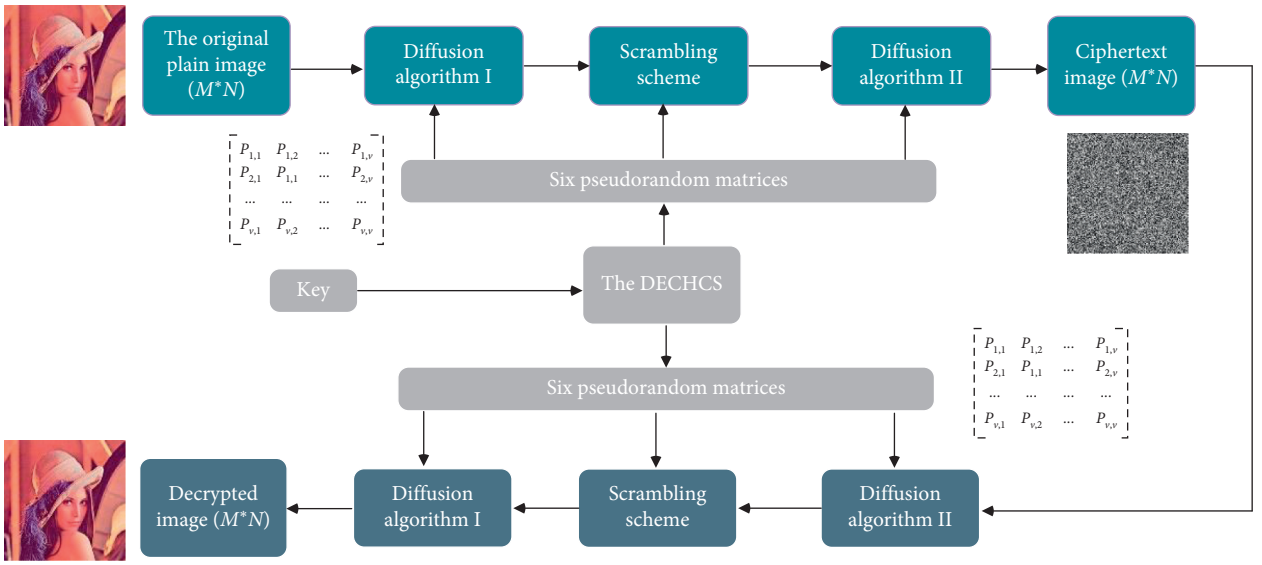

Figure 10: Image encryption-decryption block diagram.

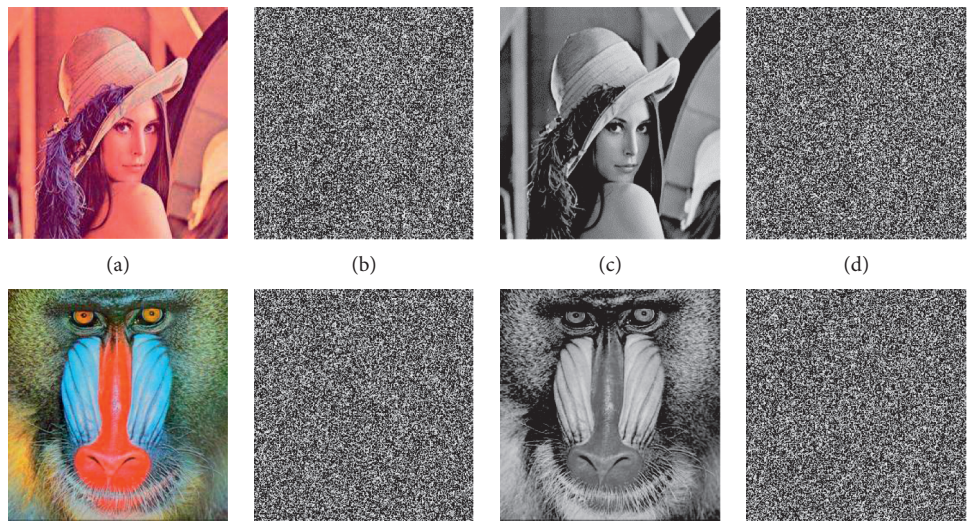

Figure 11: The algorithm of image encryption based on our proposed algorithm: (a) original plaintext image; (b) ciphertext image with correct key; (c) decryption image with correct key; (d) decryption image with wrong key.

successfully decrypted. Figures 11(g) and 11(h) show the operation of decryption using the wrong key. The results show that the algorithm can be used as a part of image encryption and has high sensitivity to its stream key. The results show that the algorithm is immune to differential attack.

5.1. Histogram Analysis. Histogram analysis provides information about the distribution of the number of pixels in the image for each value of pixel strength; that is, its distribution reflects the statistical characteristics of the image. If the probability of all the intensity pixels generated in the ciphertext image histogram is equal, the encryption symmetry is high and has good uniformity. Intuitively, Figures 12(a) and 12(b) show that the histogram of Lena and Baboon of the original plaintext images fluctuates, which means that the data can be extracted visually. However, compared with the original plaintext image, in Figures 12(c) and 12(d), the histogram of the ciphertext image is uniformly distributed, similar to a straight line. By comparing the histogram of plaintext image and its corresponding ciphertext image, it shows that each pixel strength in encrypted ciphertext image has almost equal generation probability. In Table 5, the variance of the histogram is obtained by the key. For the measurement of histogram difference of gray image, the variance values between different images have obvious changes in Table 5 which indicate that the smaller the variance is, the higher the uniformity of the ciphertext image is. Peak signal to noise ratio (PSNR) is an objective standard to evaluate image quality. It is often used as a measurement method of signal reconstruction quality in image compression and other fields. It can be simply defined by mean square error (MSE). Generally speaking, the higher the PSNR value, the higher the image quality. In this paper, PSNR = 40.1030 dB. These show that the encryption algorithm can well hide the statistical information of the image and ensure the security of the ciphertext image. Therefore, the proposed algorithm is efficient.

5.2. Image Correlation. In digital image, the ciphertext image is generated by a good encryption algorithm theoretically, the adjacent pixels in the plaintext image have a high degree of correlation, while the correlation of the adjacent pixels of the ciphertext image is close to 0, and there is no correlation. In this experiment, we randomly select 2000 pairs of adjacent pixels on horizontal, vertical, positive diagonal, and negative diagonal from plaintext image and ciphertext image, respectively, and calculate the corresponding correlation coefficient. The correlation coefficient is calculated according to equation (5).

As shown in Figures 13–16, the adjacent pixel pairs of plaintext image in all directions are dense on the $y = x$ line, while the adjacent pixel pairs of ciphertext image in each direction are evenly distributed in the matrix area, which shows that the plaintext image has strong correlation in all directions, while the ciphertext image has no correlation in all directions. In addition, Table 6 shows the correlation coefficient values of color plaintext images and their respective encrypted versions used in our experiments. From Table 6, we can see that the correlation coefficient of the ciphertext image is close to 0. In order to prove the effectiveness of the proposed image encryption mechanism in terms of correlation coefficient, in Table 7, we provide the average value of the correlation coefficient and various image encryption methods to verify the efficiency of the proposed method.

5.3. Image Differential Attack. Biham and Shamir introduced differential attack, a cryptanalysis technique, whose principle is to analyze and utilize the influence of small

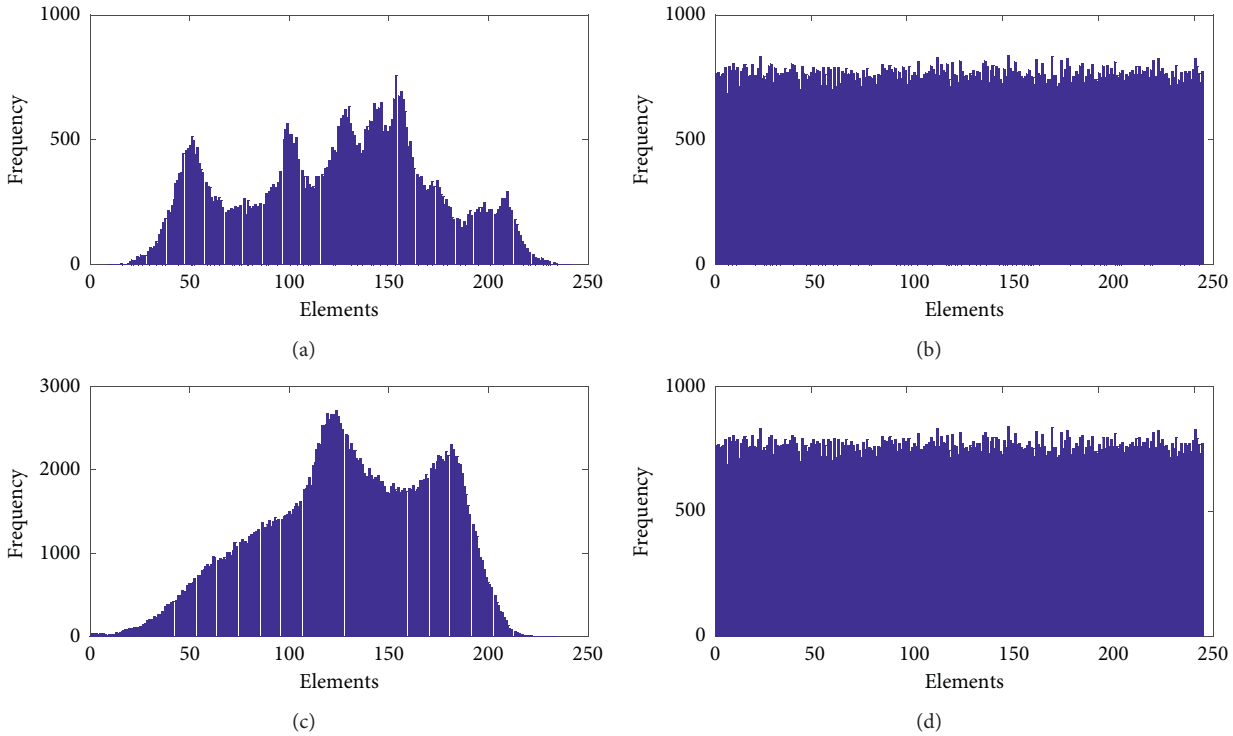

Figure 12: Histogram of plaintext image and ciphertext image.

Table 5: Variances of histograms comparing the plain images and ciphered images.

Image	Lena	Baboon
Plaintext	4048344.5312	3284085.5312
Ciphertext	16405.6454	16377.5066

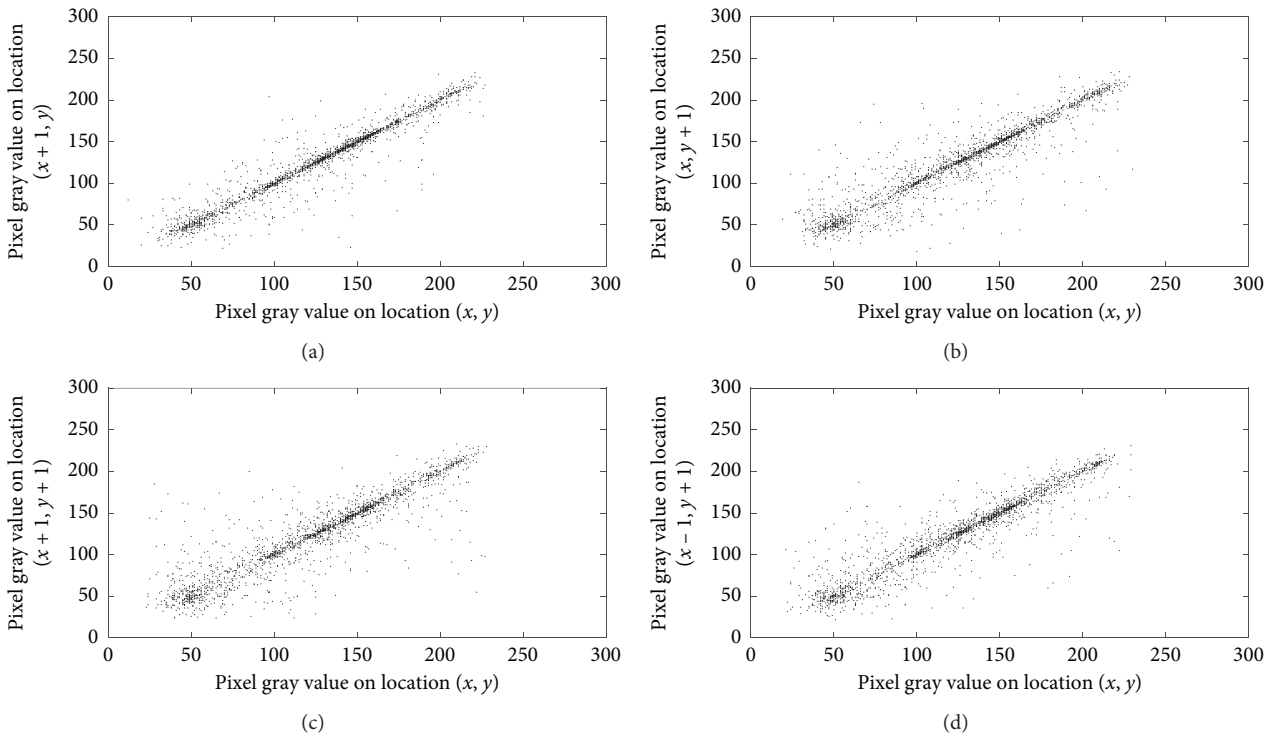

Figure 13: Lena plaintext image correlation: (a) horizontal direction; (b) vertical direction; (c) positive diagonal direction; (d) negative diagonal direction.

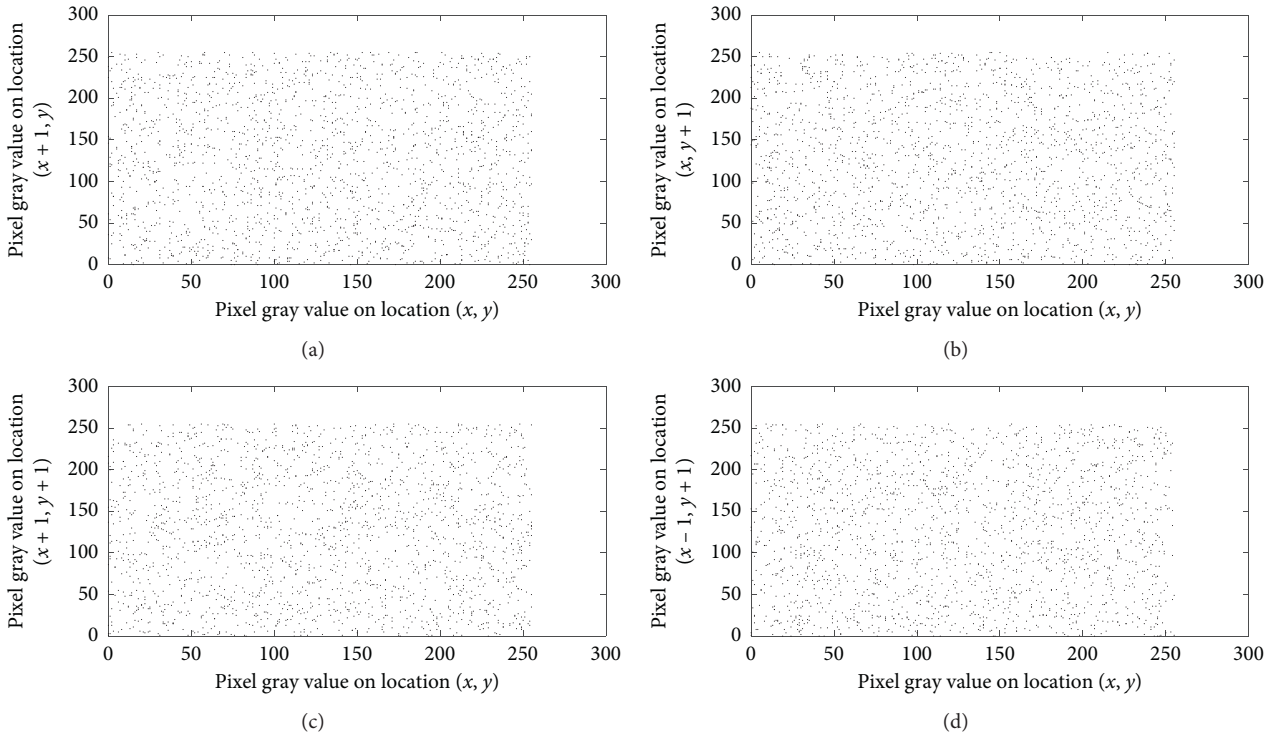

Figure 14: Lena ciphertext image correlation: (a) horizontal direction; (b) vertical direction; (c) positive diagonal direction; (d) negative diagonal direction.

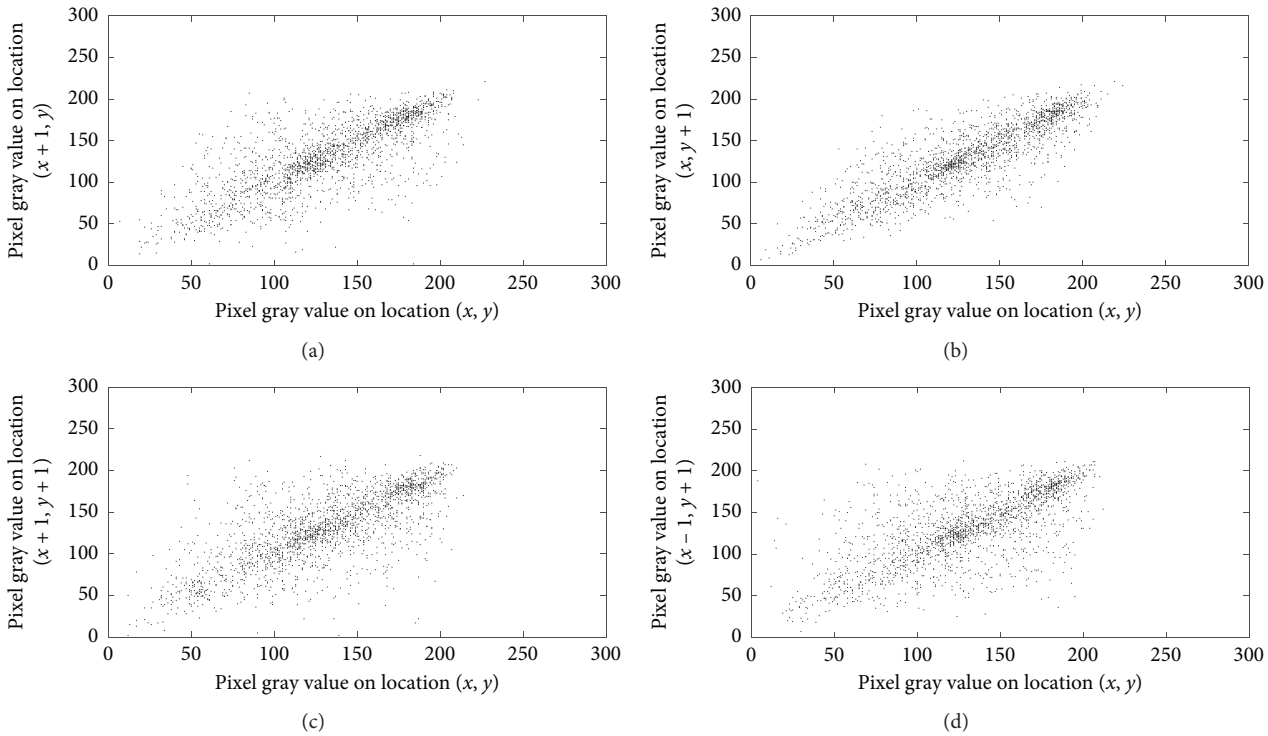

Figure 15: Baboon plaintext image correlation: (a) horizontal direction; (b) vertical direction; (c) positive diagonal direction; (d) negative diagonal direction.

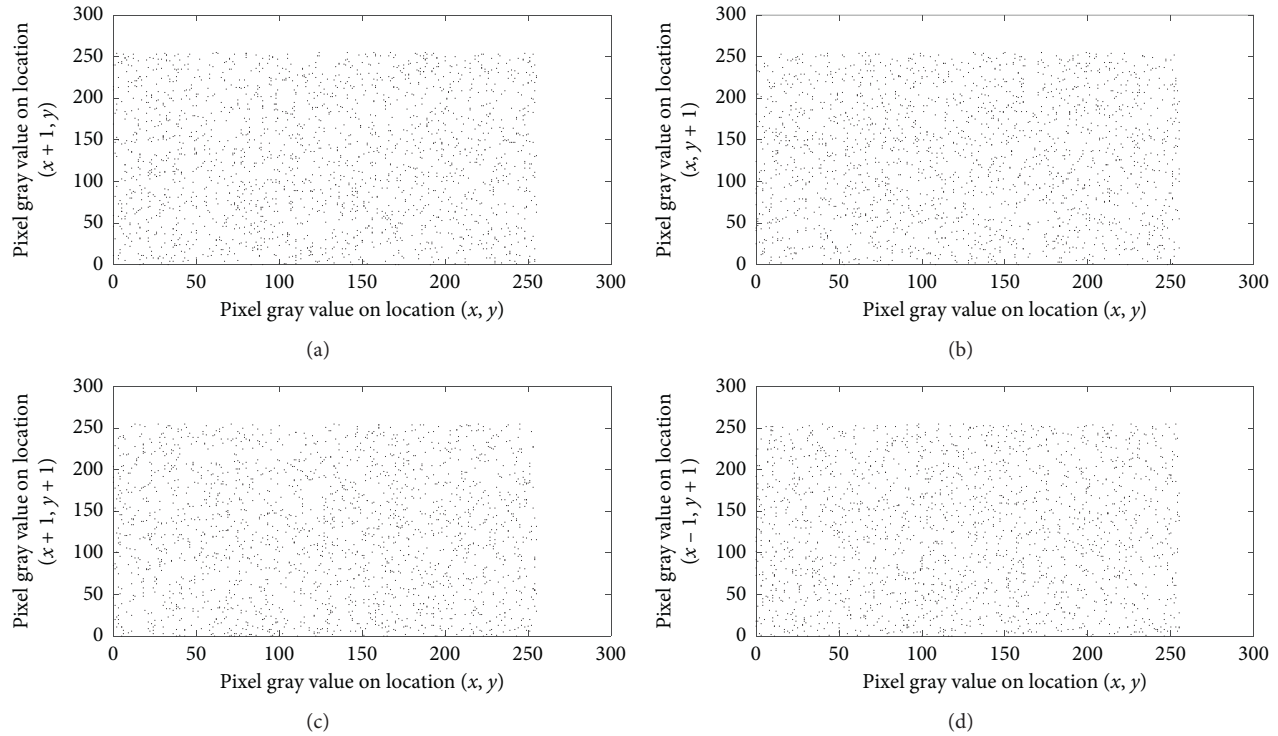

Figure 16: Baboon ciphertext image correlation: (a) horizontal direction; (b) vertical direction; (c) positive diagonal direction; (d) negative diagonal direction.

Table 6: Calculation results of correlation coefficient.

Image		Horizontal	Vertical	Positive diagonal	Negative diagonal
Lena	Plaintext	0.95868	0.92361	0.91376	0.93566
	Ciphertext	**0.00146**	−0.01890	0.04126	**−0.00334**
Baboon	Plaintext	0.73630	0.87027	0.72811	0.69000
	Ciphertext	−0.02581	−0.02097	**−0.00515**	−0.01158
Average	Ciphertext	0.01217	0.01993	0.01855	−0.00412

Table 7: Comparison of correlation coefficients of different encryption methods.

Method	Horizontal	Vertical	Diagonal
Proposed	0.01217	0.01993	−0.00412
Ref. [64] AES-S	0.0467	−0.0173	−0.0078
Ref. [64] AES-D	0.0307	0.0190	0.0102
Ref. [35]	−0.0519	−0.0385	0.0046
Ref. [73]	0.0018	0.0003	0.0027
Ref. [74]	0.0285	−0.0350	−0.0102
Ref. [75]	0.2725	−0.0256	−0.0661

differences in the input plaintext on the corresponding image ciphertext differences. The difference of seed values used by the same input vector is very small, and the sequence generated is completely different. However, when two adjacent input vectors use the same seed value, it may bring great convenience to the attacker, and the differential attack will be much simpler. Therefore, this situation should be considered in the algorithm design to generate secure output and have high resistance against differential attack. If there is a significant difference between the two ciphertext images, it indicates that the key sensitivity of the image encryption scheme is strong; otherwise, it is weak. A good image encryption scheme should have strong key sensitivity.

In this chapter, in order to verify the key sensitivity of the proposed image encryption algorithm, we can use two metrics commonly used in the cryptography system to measure the system's ability to resist differential attack. The pixel rate of change (NPCR) and normalized mean change

intensity (UACI) were used to quantify two adjacent initial seeds and generate two pseudorandom sequences. NPCR measures how many elements are different between two PRNG (P_i) and (P_i'), while UACI measures the average intensity of the difference between (P_i) and (P_i'), which makes up for the one sidedness of NPCR in measuring the difference between two images. The definition expression of NPCR is given by the following equation:

$$\mathrm{NPCR}(P_i, P_i') = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left|\mathrm{sign}\left(P_i(i,j) - P_i'(i,j)\right)\right| \times 100\%. \tag{14}$$

The probability that two images of the same size have different pixels at any position is (255/256)× 100% ≈ 99.6094%. That is, the theoretical NPCR value of the given image and random image is also 99.6094%.

The definition expression of UACI is given by the following equation:

$$\mathrm{UACI} = \frac{1}{MN}\sum_{i=0}^{M}\sum_{j=0}^{N}\frac{\left|P_i(i,j) - P_i'(i,j)\right|}{255-0} \times 100\%. \tag{15}$$

The average value of the ratio of the difference between a given image and a random image and the maximum difference (255) is called UACI.

The difference between two images of the same size is defined as $D = |P_i(i,j) - P_i'(i,j)|$. Then, the adjacent pixels of the difference image are divided into $(M-1)\times(N-1)$ small image blocks (the image size is $M \times N$) according to the matrix of 2×2. The average value of the absolute value of the difference between any two elements is as follows:

$$\Delta_i = \frac{1}{6}\left(\left|d_{i1} - d_{i2}\right| + \left|d_{i1} - d_{i3}\right| + \left|d_{i1} - d_{i2}\right| + \left|d_{i1} - d_{i4}\right| + \left|d_{i2} - d_{i4}\right| + \left|d_{i3} - d_{i4}\right|\right). \tag{16}$$

BACI (block average changing intensity) is to calculate the ratio of the maximum difference between Δ_i and pixels of all small image blocks, so as to solve the problem when the image visual effect is similar, but NPCR and UACI values are not ideal.

$$\mathrm{BACI} = \frac{1}{(M-1)(N-1)}\sum_{i=1}^{(M-1)(N-1)}\frac{\Delta_i}{255}. \tag{17}$$

In this experiment, 100 keys were randomly generated, and $x_1(0), x_2(0)$and $y_1(0), y_2(0)$ were changed slightly in turn. The two keys before and after the change were used to encrypt the plaintext image, respectively. Then, the NPCR, UACI, and BACI between the two ciphertext images corresponding to the same plaintext image were analyzed. The NPCR, UACI, and BACI results are calculated as the average of 100 trials as shown in Table 8. We can see that NPCR is over 99%, and UACI is over 33%, which is very close to the ideal value. This means that the proposed PRNG has a high ability to resist differential attack. The sensitivity results of other existing methods to small changes in the composition of the original image are shown in Table 8. This performance is further illustrated by comparing our mean with the average obtained from other methods (see Table 9), which shows minimal error with the ideal value.

5.4. Choose-Plaintext Attack. Choose-plaintext attack (CPA) means that the attacker can select any plaintext image and encrypt the ciphertext image with the corresponding image encryption scheme; that is, the attacker knows the selected plaintext image and the encrypted ciphertext image and then uses the generated ciphertext image to speculate the key. In the implementation of the CPA algorithm, this paper uses all-black and all-white plaintext images of the same size for encryption, which is shown in Figure 17. By measuring the statistical information of ciphertext images, the following results are obtained, as shown in Table 10.

It can be seen that the ciphertext image has the characteristics of randomness and uniform distribution, and the correlation of adjacent pixels in positive diagonal, antidiagonal, horizontal, and vertical directions is close to 0. So, the data can prove that the encryption algorithm can effectively resist the chosen-plaintext attack.

5.5. Plaintext Sensitivity Analysis. Plaintext sensitivity analysis is to use the same given key to encrypt two original plaintext images with the help of the image encryption system to get two corresponding ciphertext images. The specific operation is as follows. Randomly select a pixel in the original plaintext image P_1, change the value of the selected pixel, and record the changed image as plaintext image P_2. Using the same given key to encrypt plaintext image P_1 and P_2, get the corresponding ciphertext image C_1 and C_2. Then, compare the difference between the two ciphertext images. If the difference between the two ciphertext images is very different, the image cryptosystem is said to have good plaintext sensitivity. It can be seen from Table 11 that the calculation results of NPCR, UACI, and BACI are close to the theoretical value and can resist the chosen-plaintext attack or known plaintext attack. In this paper, the change amount of P_1 in the (i, j) position is 1, and the value of the changed pixel is recorded as $P_2(i,j) = (P_1(i,j) + 1)\mathrm{mod}256$.

5.6. Information Entropy. Image information entropy is an important reference index, which reflects the uncertainty of image information and the richness of information contained in it. Generally speaking, the greater the information contained in the image (the greater the uncertainty), the less information can be directly observed and the greater the entropy value is. The calculation formula of information entropy is shown in the following equation:

$$E = -\sum_{i=0}^{L} p(i)\log_2\ p(i). \tag{18}$$

Table 8: NPCR, UACI, and BACI test results for the key $x_1(0), x_2(0)$, $y_1(0)$, and $y_2(0)$ with minor changes reported in the experiment.

Index		Lena	Baboon	Theoretical value
$x_1(0)$	NPCR (%)	99.6098	99.6095	99.6094
	UACI (%)	33.4603	33.4632	33.4635
	BACI (%)	26.7734	26.7716	26.7712
$x_2(0)$	NPCR (%)	99.6090	99.6082	99.6094
	UACI (%)	33.4647	33.4620	33.4635
	BACI (%)	26.7720	26.7677	26.7712
$y_1(0)$	NPCR (%)	99.6109	99.6100	99.6094
	UACI (%)	33.4653	33.4600	33.4635
	BACI (%)	26.7661	26.7679	26.7712
$y_2(0)$	NPCR (%)	99.6088	99.6101	99.6094
	UACI (%)	33.4639	33.4636	33.4635
	BACI (%)	26.7750	26.7686	26.7712
Average	NPCR (%)	99.6096	99.6094	99.6094
	UACI (%)	33.4635	33.4622	33.4635
	BACI (%)	26.7716	26.7689	26.7712

Table 9: Comparison of NPCR, UACI, and BACI values with different encryption methods.

Method	NPCR (%)	UACI (%)
Proposed	99.6095	33.4628
Ref. [65]	99.5511	33.3461
Ref. [76]	99.6114	33.4523
Ref. [77]	99.5800	30.5840
Ref. [78]	99.57	33.30

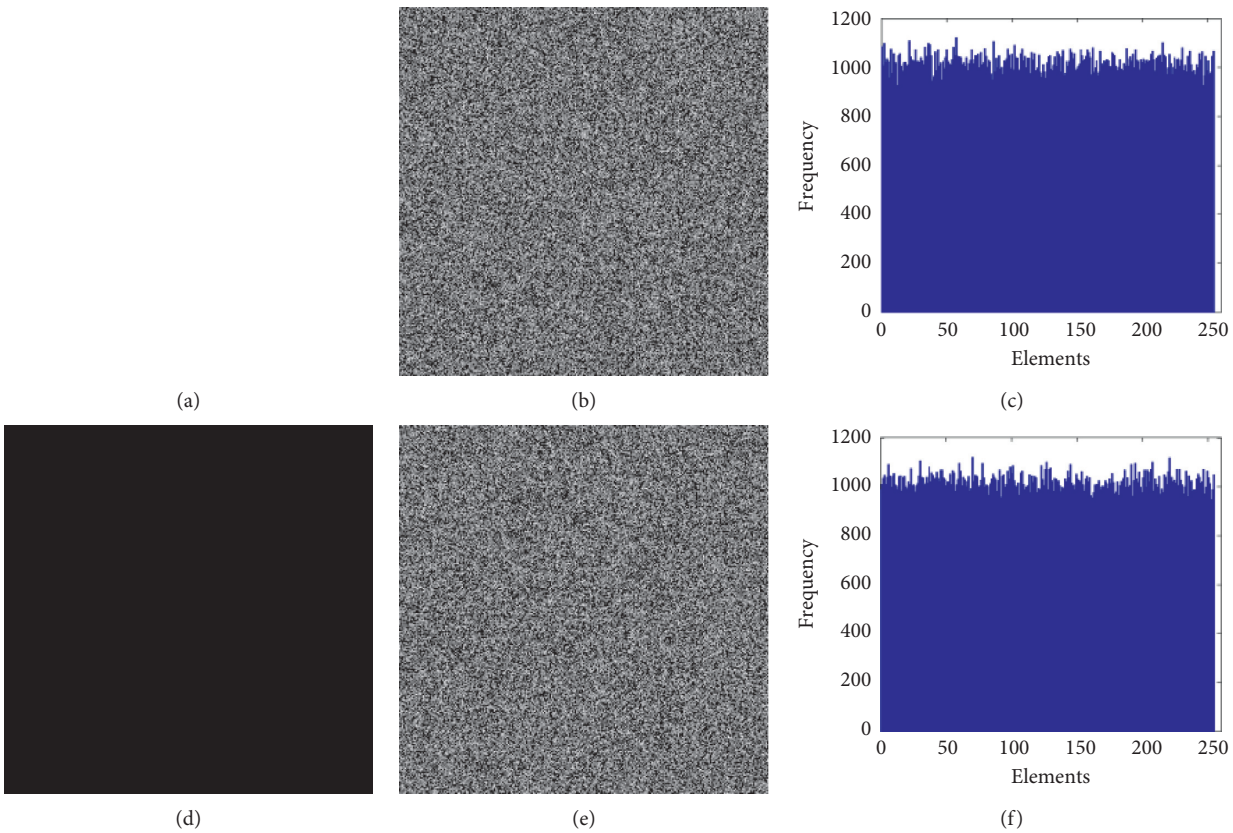

Figure 17: Encryption and decryption of white and black images and histogram of ciphertext images.

Table 10: Result of CPA analysis.

Image	Information entropy	NPCR	UACI	BACI	Correlation coefficients			
					Positive diagonal	Antidiagonal	Horizontal	Diagonal
All-white	7.99917	99.60479	33.45712	26.69787	−0.04363	0.03466	−0.00290	−0.03635
All-black	7.99917	99.60594	33.49045	26.74253	−0.01469	−0.03348	0.00022	−0.02505

Table 11: Results of sensitivity analysis.

Method	Lena	Baboon	Theoretical value
NPCR	99.6105	99.6111	99.6094
UACI	33.4624	33.4666	33.4635
BACI	26.7745	26.7786	26.7712

Table 12: Information entropy of the plaintext and ciphertext images.

Lena		Baboon	
Plaintext	Ciphertext	Plaintext	Ciphertext
7.47558	7.99914	7.36713	7.99905

Table 13: Comparisons of information entropy values for the proposed method alongside those from different encryption methods.

Method	Lena	Baboon
Proposed	7.9991	7.9990
Ref. [64] AES-S	7.9970	
Ref. [64] AES-D	7.9969	
Ref. [66]	7.9975	7.9971
Ref. [78]	7.9901	
Ref. [79]		7.9120
Ref. [80]	7.9975	

Here, L is the gray level number of the image and $p(i)$ is the probability of gray value. If the possible value of color image is $2^8 \times 3$, the ideal entropy value of each channel (R, G, and B) is equal to 8 bits, and the ideal entropy value of gray image is also equal to 8 bits. Therefore, in order to verify the effectiveness of the proposed encryption mechanism, the entropy value of the encrypted plaintext image should be close to 8. Table 12 lists the Shannon entropy of the original plaintext image and its corresponding ciphertext image. Obviously, the information entropy of each plaintext image is different from the theoretical value (i.e., 8), and the information entropy of all ciphertext images is very close to the theoretical value. Consequently, the proposed image encryption scheme is secure against entropy attack. In addition, in Table 13, we provide a comparison of information entropy between our proposed method and other existing image encryption methods. The results of two tables (i.e., Tables 12 and 13) verify the performance of our proposed method in terms of expected results.

6. Conclusion

In this paper, a new pseudorandom number generation method and image encryption are proposed by using a 6D continuous memristive hyperchaotic system and a 2D SF-SIMM discrete hyperchaotic mapping. Through the analysis of the weak key characteristics, key sensitivity, correlation, and spectral entropy of the PRNG, the generated pseudorandom number sequence can pass NIST test package. Then, an image encryption algorithm based on the double entropy source hyperchaotic system is proposed. The results of encryption and decryption, security analysis, and anti-differential attack analysis ensure the effectiveness of the algorithm. The performance comparison with existing encryption algorithms shows that the algorithm is superior.

Data Availability

The data used to support the findings of this study are available from the corresponding authors upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under grants 61504013 and 61702052, Natural Science Foundation of Hunan Province under grants 2019JJ50648, 2020JJ4622, and 2020JJ4221, Guangxi Key Laboratory of Cryptography and Information Security under grant GCIS201919, Postgraduate Training Innovation Base Construction Project of Hunan Province under grant 2020-172-48, Postgraduate Scientic Research Innovation Project of Hunan Province under grant CX20200884, Scientific Research Fund of Hunan Provincial Education Department under grant 18A137, Young Teacher Development Program Project of Changsha University of Science and Technology under grant 2019QJCZ013, and Special Funds for the Construction of Innovative Provinces in Hunan Province under grant 2020JK4046.

References

[1] J. Vijila and A. Raj, "Ameliorate security by introducing security server in software defined network," *Computers, Materials & Continua*, vol. 62, no. 3, pp. 1077–1096, 2020.

[2] Z. Fang, J. Cai, and L. Tian, "Security of chip bank card in remote payment based on risk feature," *Computer Systems Science and Engineering*, vol. 35, no. 4, pp. 299–305, 2020.

[3] Z. Baig and S. Zeadally, "Cyber-security risk assessment framework for critical infrastructures," *Intelligent Automation & Soft Computing*, vol. 25, no. 1, pp. 121–129, 2019.

[4] W. Wang, X. Wang, J. Wang, N. N. Xiong, S. Cai, and P. Liu, "Ensuring cryptography chips security by preventing scan-based side-channel attacks with improved DFT architecture," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, p. 1, 2020.

[5] Z. Yin, Y. Song, H. Chen, and Y. Cao, "A security sensitive function mining approach based on precondition pattern analysis," *Computers, Materials & Continua*, vol. 63, no. 2, pp. 1013–1029, 2020.

[6] A. Kelec and Z. Djuric, "A proposal for addressing security issues related to dynamic code loading on android platform," *Computer Systems Science and Engineering*, vol. 35, no. 4, pp. 271–282, 2020.

[7] P. Centonze, "Security and privacy frameworks for access control big data systems," *Computers, Materials & Continua*, vol. 59, no. 2, pp. 361–374, 2019.

[8] Y. Park, H. Choi, S. Cho, and Y.-G. Kim, "Security analysis of smart speaker: security attacks and mitigation," *Computers, Materials & Continua*, vol. 61, no. 3, pp. 1075–1090, 2019.

[9] W. M. Eid, S. Atawneh, and M. Al-Akhras, "Framework for cybersecurity centers to mass scan networks," *Intelligent Automation & Soft Computing*, vol. 26, no. 6, pp. 1319–1334, 2020.

[10] F. Yu, L. Li, Q. Tang, S. Cai, Y. Song, and Q. Xu, "A survey on true random number generators based on chaos," *Discrete Dynamics in Nature and Society*, vol. 2019, Article ID 2545123, 10 pages, 2019.

[11] A. Akhshani, A. Akhavan, A. Mobaraki, S.-C. Lim, and Z. Hassan, "Pseudo random number generator based on quantum chaotic map," *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 1, pp. 101–111, 2014.

[12] L. Palacios-Luengas, J. L. Pichardo-Méndez, J. A. Díaz-Méndez, F. Rodríguez-Santos, and R. Vázquez-Medina, "PRNG based on skew tent map," *Arabian Journal for Science and Engineering*, vol. 44, no. 4, pp. 3817–3830, 2018.

[13] H. Lin, C. Wang, W. Hao, and Y. Tan, "Chaotic dynamics in a neural network with different types of external stimuli," *Communications in Nonlinear Science and Numerical Simulation*, vol. 90, Article ID 105390, 2020.

[14] Y. M. Tan and C. H. Wang, "A simple locally active memristor and its application in HR neurons," *Chaos*, vol. 30, no. 5, Article ID 053118, 2020.

[15] L. Zhou, F. Tan, F. Yu, and W. Liu, "Cluster synchronization of two-layer nonlinearly coupled multiplex networks with multi-links and time-delays," *Neurocomputing*, vol. 359, pp. 264–275, 2019.

[16] W. Yao, C. H. Wang, Y. C. Sun et al., "Exponential multistability of memristive Cohen-Grossberg neural networks with stochastic parameter perturbations," *Applied Mathematics and Computation*, vol. 386, Article ID 125483, 2020.

[17] Q. Z. Wan, Z. T. Zhou, W. K. Ji, C. H. Wang, and F. Yu, "Dynamic analysis and circuit realization of a novel no-equilibrium 5D memristive hyperchaotic system with hidden extreme multistability," *Complexity*, vol. 2020, Article ID 7106861, 16 pages, 2020.

[18] Z. Wen, Z. Li, and X. Li, "Transient MMOs in memristive chaotic system via tiny perturbation," *Electronics Letters*, vol. 56, no. 2, pp. 78–80, 2020.

[19] Z. Wen, Z. Li, and X. Li, "Bursting dynamics in parametrically driven memristive Jerk system," *Chinese Journal of Physics*, vol. 66, pp. 327–334, 2020.

[20] Q. L. Deng, C. H. Wang, and L. M. Yang, "Four-wing hidden attractors with one stable equilibrium point," *International Journal of Bifurcation and Chaos*, vol. 30, no. 6, Article ID 2050086, 2020.

[21] J. R. Sun, M. Peng, F. Liu, and C. Tang, "Protecting compressive ghost imaging with hyper-chaotic system and DNA encoding," *Complexity*, vol. 2020, Article ID 8815315, 13 pages, 2020.

[22] J. Liu, J. Li, J. Cheng et al., "A novel robust watermarking algorithm for encrypted medical image based on DTCWT-DCT and chaotic map," *Computers, Materials & Continua*, vol. 61, no. 2, pp. 889–910, 2019.

[23] J. Chen, Y. Zhang, L. Qi, C. Fu, and L. Xu, "Exploiting chaos-based compressed sensing and cryptographic algorithm for image encryption and compression," *Optics & Laser Technology*, vol. 99, pp. 238–248, 2018.

[24] C. Xu, J. R. Sun, and C. H. Wang, "An image encryption algorithm based on random walk and hyperchaotic systems," *International Journal of Bifurcation and Chaos*, vol. 30, no. 4, Article ID 2050060, 2020.

[25] H. Lin, C. Wang, F. Yu et al., "An extremely simple multi-wing chaotic system: dynamics analysis, encryption application and hardware implementation," *IEEE Transactions on Industrial Electronics*, p. 1, 2021.

[26] F. Yu, L. Liu, H. Shen et al., "Dynamic analysis, circuit design and synchronization of a novel 6D memristive four-wing hyperchaotic system with multiple coexisting attractors," *Complexity*, vol. 2020, Article ID 5904607, 17 pages, 2020.

[27] C. Zhou, C. H. Wang, Y. C. Sun, and W. Yao, "Weighted sum synchronization of memristive coupled neural networks," *Neurocomputing*, vol. 403, pp. 225–232, 2020.

[28] W. Yao, C. Wang, Y. Sun, C. Zhou, and H. Lin, "Synchronization of inertial memristive neural networks with time-varying delays via static or dynamic event-triggered control," *Neurocomputing*, vol. 404, pp. 367–380, 2020.

[29] Y. Li, Z. Li, M. Ma, and M. Wang, "Generation of grid multi-wing chaotic attractors and its application in video secure communication system," *Multimedia Tools and Applications*, vol. 79, no. 39-40, pp. 29161–29177, 2020.

[30] B. Lu, F. Liu, X. Ge, and Z. Li, "Cryptanalysis and improvement of a chaotic map-control-based and the plain image-related cryptosystem," *Computers, Materials & Continua*, vol. 61, no. 2, pp. 687–699, 2019.

[31] Q. Li, X. Wang, X. Wang, B. Ma, C. Wang, and Y. Shi, "An encrypted coverless information hiding method based on generative models," *Information Sciences*, vol. 553, pp. 19–30, 2021.

[32] L. Zhou, F. Tan, and F. Yu, "A robust synchronization-based chaotic secure communication scheme with double-layered and multiple hybrid networks," *IEEE Systems Journal*, vol. 14, no. 2, pp. 2508–2519, 2020.

[33] H. Hu, L. Liu, and N. Ding, "Pseudorandom sequence generator based on the Chen chaotic system," *Computer Physics Communications*, vol. 184, no. 3, pp. 765–768, 2013.

[34] V. Lynnyk, N. Sakamoto, and S. Čelikovský, "Pseudo random number generator based on the generalized Lorenz chaotic system," *IFAC-PapersOnLine*, vol. 48, no. 18, pp. 257–261, 2015.

[35] M. O. Meranza-Castillón, M. A. Murillo-Escobar, R. M. López-Gutiérrez, and C. Cruz-Hernández, "Pseudo-random number generator based on enhanced Hénon map and its implementation," *AEU—International Journal of Electronics and Communications*, vol. 107, pp. 239–251, 2015.

[36] Y. Wang, Z. Liu, J. Ma et al., "A pseudorandom number generator based on piecewise logistic map," *Nonlinear Dynamics*, vol. 83, no. 4, pp. 2373–2391, 2015.

[37] X. Chen, S. Qian, F. Yu et al., "Pseudorandom number generator based on three kinds of four-wing memristive hyperchaotic system and its application in image encryption," *Complexity*, vol. 2020, Article ID 8274685, 17 pages, 2020.

[38] X. Tong and Y. Liu, "Hyperchaotic system-based pseudo-random number generator," *IET Information Security*, vol. 10, no. 6, pp. 433–441, 2016.

[39] F. Yu, Q. Wan, J. Jin et al., "Design and FPGA implementation of a pseudorandom number generator based on a four-wing memristive hyperchaotic system and Bernoulli map," *IEEE Access*, vol. 7, pp. 181884–181898, 2019.

[40] J. Zeng and C. H. Wang, "A novel hyper-chaotic image encryption system based on particle swarm optimization algorithm and cellular automata," *Security and Communication Networks*, vol. 2021, Article ID 6675565, 15 pages, 2021.

[41] H. Lin, C. Wang, and Y. Tan, "Hidden extreme multistability with hyperchaos and transient chaos in a Hopfield neural network affected by electromagnetic radiation," *Nonlinear Dynamics*, vol. 99, no. 3, pp. 2369–2386, 2020.

[42] G. Cheng, C. Wang, and C. Xu, "A novel hyper-chaotic image encryption scheme based on quantum genetic algorithm and compressive sensing," *Multimedia Tools and Applications*, vol. 79, pp. 29243–29263, 2020.

[43] X. Ye, J. Mou, C. Luo, and Z. Wang, "Dynamics analysis of Wien-bridge hyperchaotic memristive circuit system," *Nonlinear Dynamics*, vol. 92, no. 3, pp. 923–933, 2018.

[44] C. G. Li and Z. Z. Han, "Pseudo-random sequences generator based on discrete hyperchaotic systems," *Journal of Systems Engineering and Electronics*, vol. 4, pp. 86–91, 2003.

[45] L. Chua, "Memristor-the missing circuit element," *IEEE Transactions on Circuit Theory*, vol. 18, no. 5, pp. 507–519, 1971.

[46] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing memristor found," *Nature*, vol. 453, no. 7191, pp. 80–83, 2008.

[47] C. Wang, H. Xia, and L. Zhou, "A memristive hyperchaotic multiscroll jerk system with controllable scroll numbers," *International Journal of Bifurcation and Chaos*, vol. 27, no. 6, Article ID 1750091, 2017.

[48] S. Zhong, "Heterogeneous memristive models design and its application in information security," *Computers, Materials & Continua*, vol. 60, no. 2, pp. 465–479, 2019.

[49] H. Lin, C. Wang, Y. Sun, and W. Yao, "Firing multistability in a locally active memristive neuron model," *Nonlinear Dynamics*, vol. 100, no. 4, pp. 3667–3683, 2020.

[50] M. Zhu, C. Wang, Q. Deng, and Q. Hong, "Locally active memristor with three coexisting pinched hysteresis loops and its emulator circuit," *International Journal of Bifurcation and Chaos*, vol. 30, no. 13, Article ID 2050184, 2020.

[51] H. R. Lin, C. H. Wang, Q. H. Hong, and Y. C. Sun, "A multistable memristor and its application in a neural network," *IEEE Transactions on Circuits and Systems-II: Brief Papers*, vol. 67, no. 12, pp. 3472–3476, 2020.

[52] F. Yu, L. Liu, H. Shen et al., "Multistability analysis, coexisting multiple attractors and FPGA implementation of Yu-Wang four-wing chaotic system," *Mathematical Problems in Engineering*, vol. 2020, Article ID 7530976, 16 pages, 2020.

[53] B. A. Mezatio, M. T. Motchongom, B. R. W. Tekam, R. Kengne, R. Tchitnga, and A. Fomethe, "A novel memristive 6D hyperchaotic autonomous system with hidden extreme multistability," *Chaos, Solitons & Fractals*, vol. 120, pp. 100–115, 2019.

[54] W. Liu, K. Sun, and S. He, "SF-SIMM high-dimensional hyperchaotic map and its performance analysis," *Nonlinear Dynamics*, vol. 89, no. 4, pp. 2521–2532, 2017.

[55] G. Ravikanth, K. V. N. Sunitha, and B. Eswara Reddy, "Location related signals with satellite image fusion method using visual image integration method," *Computer Systems Science and Engineering*, vol. 35, no. 5, pp. 385–393, 2020.

[56] Q. Mo, H. Yao, F. Cao, Z. Chang, and C. Qin, "Reversible data hiding in encrypted image based on block classification permutation," *Computers, Materials & Continua*, vol. 59, no. 1, pp. 119–133, 2019.

[57] X. Zhang, S. Zhou, J. Fang, and Y. Ni, "Pattern recognition of construction bidding system based on image processing," *Computer Systems Science and Engineering*, vol. 35, no. 4, pp. 247–256, 2020.

[58] L. Pan, J. Qin, H. Chen, X. Xiang, C. Li, and R. Chen, "Image augmentation-based food recognition with convolutional neural networks," *Computers, Materials & Continua*, vol. 59, no. 1, pp. 297–313, 2019.

[59] F. Zhang, H. Zhao, W. Ying, Q. Liu, A. Noel et al., "Human face sketch to RGB image with edge optimization and generative adversarial networks," *Intelligent Automation & Soft Computing*, vol. 26, no. 6, pp. 1391–1401, 2020.

[60] J. Chen, L. Chen, L. Y. Zhang, and Z.-l. Zhu, "Medical image cipher using hierarchical diffusion and non-sequential encryption," *Nonlinear Dynamics*, vol. 96, no. 1, pp. 301–322, 2019.

[61] J. Chen, L. Chen, and L. Y. Zhang, "Universal chosen-ciphertext attack for a family of image encryption schemes," *IEEE Transactions on Multimedia*, 2020.

[62] J. Chen, L. Chen, and Y. Zhou, "Cryptanalysis of image ciphers with permutation-substitution network and chaos," *IEEE Transactions on Circuits and Systems for Video Technology*, p. 1, 2020.

[63] J. Deng, M. Zhou, C. Wang, S. Wang, and C. Xu, "Image segmentation encryption algorithm with chaotic sequence generation participated by cipher and multi-feedback loops," *Multimedia Tools and Applications*, 2021.

[64] Y. Zhang, "Test and verification of aes used for image encryption," *3D Research*, vol. 9, no. 1, pp. 37–51, 2018.

[65] X. Wang, L. Liu, and Y. Zhang, "A novel chaotic block image encryption algorithm based on dynamic random growth technique," *Optics and Lasers in Engineering*, vol. 66, pp. 10–18, 2015.

[66] M. Alawida, A. Samsudin, J. S. Teh, and R. S. Alkhawaldeh, "A new hybrid digital chaotic system with applications in image encryption," *Signal Process*, vol. 160, pp. 45–58, 2019.

[67] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," Technical report, NIST, Gaithersburg, MD, USA, 2001.

[68] C. Grebogi, E. Ott, and J. A. Yorke, "Roundoff-induced periodicity and the correlation dimension of chaotic attractors," *Physical Review A*, vol. 38, no. 7, p. 3688, 1988.

[69] M. Francois, T. Grosges, D. Barchiesi, and R. Erra, "A new pseudo-random number generator based on two chaotic maps," *Informatica*, vol. 24, no. 2, pp. 181–197, 2013.

[70] B. Stoyanov and K. Kordov, "Novel secure pseudo-random number generation scheme based on two Tinkerbell maps," *Advanced Studies in Theoretical Physics*, vol. 9, pp. 411–421, 2015.

[71] M. García-Martínez and E. Campos-Cantón, "Pseudo-random bit generator based on multi-modal maps," *Nonlinear Dynamics*, vol. 82, no. 4, pp. 2119–2131, 2015.

[72] M. A. Murillo-Escobar, C. Cruz-Hernandez, and L. Cardoza Avendano, "A novel pseudorandom number generator based on pseudorandomly enhanced logistic map," *Nonlinear Dynamics*, vol. 87, pp. 407–425, 2016.

[73] X. Wang and S. Gao, "Image encryption algorithm for synchronously updating Boolean networks based on matrix semi-tensor product theory," *Information Sciences*, vol. 507, pp. 16–36, 2020.

[74] Y. Zhou, W. Cao, and C. L. Philip Chen, "Image encryption using binary bitplane," *Signal Processing*, vol. 100, pp. 197–207, 2014.

[75] Y. Zhou, K. Panetta, S. Agaian, and C. L. P. Chen, "(n, k, p)-gray code for image systems," *IEEE Transactions on Cybernetics*, vol. 43, pp. 515–529, 2013.

[76] M. Zhou and C. Wang, "A novel image encryption scheme based on conservative hyperchaotic system and closed-loop diffusion between blocks," *Signal Processing*, vol. 171, Article ID 107484, 2020.

[77] M. Bakiri, C. Guyeux, J.-F. Couchot, and A. K. Oudjida, "Survey on hardware implementation of random number generators on FPGA: theory and experimental analyses," *Computer Science Review*, vol. 27, pp. 135–153, 2018.

[78] A. Anees, A. M. Siddiqui, and F. Ahmed, "Chaotic substitution for highly autocorrelated data in encryption algorithm," *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 9, pp. 3106–3118, 2014.

[79] J. Ahmad, M. A. Khan, F. Ahmed, and J. S. Khan, "A novel image encryption scheme based on orthogonal matrix, skew tent map, and XOR operation," *Neural Computing and Applications*, vol. 30, no. 12, pp. 3847–3857, 2018.

[80] M. Asgari-Chenaghlu, M. A. Balafar, and M. R. Feizi-Derakhshi, "A novel image encryption algorithm based on polynomial combination of chaotic maps and dynamic function generation," *Signal Process*, vol. 157, pp. 1–13, 2019.