International Journal of Innovative Computing, Information and Control
Volume **18**, Number **1**, February **2022**

 ISSN 1349-4198
pp. **73–92**

A NOVEL DIGITAL COLOR IMAGE ENCRYPTION ALGORITHM BASED ON A NEW 4-D HYPER-CHAOTIC SYSTEM AND AN IMPROVED S-BOX

Jinyuan Chen[1], Jianeng Tang[1,*], Feng Zhang[2], Hui Ni[2] and Yinghui Tang[1]

[1]College of Engineering
Huaqiao University
No. 269, Chenghua North Road, Quanzhou 362021, P. R. China
{ jinyuan_chen; yinghui_tang }@stu.hqu.edu.cn; *Corresponding author: jn_tang@hqu.edu.cn

[2]Fujian MM Electronics Co., Ltd.
No. 58, Taixin Street, Changtai Street, Licheng District, Quanzhou 362000, P. R. China
fzhang_mm@163.com; hni1234mm@88.com

Received July 2021; revised November 2021

Abstract. *With the rapid development of communication network and multimedia technology in recent years, a large number of digital images are generated and transmitted on the network. In order to make some sensitive information images safely transmitted on the Internet, this paper proposes a color image encryption algorithm based on a novel four-dimensional (4-D) hyper-chaotic system. Firstly, the key of the system is generated by the SHA-256 hash algorithm, which can effectively strengthen the algorithm's ability to resist known-plaintext attacks. Then we scramble the color plain image through Arnold map, and propose a new 4-D hyper-chaotic system and its circuit simulation. And an improved substitution box (S-box) is generated by using the chaotic pseudo-random sequence generated by the hyper-chaotic system. The scrambled image is replaced by the generated S-box, and then the replaced image is combined with the chaotic pseudo-random sequence for diffusion operation to generate ciphertext image. In the end, the experimental results and security analysis show that the algorithm has sufficient key space, high key sensitivity, and high security.*

Keywords: Color image encryption, Hyper-chaotic map, Random sequence, S-box

1. **Introduction.** In recent years, with the rapid development of communication network and multimedia technology, a large number of digital data are generated and transmitted on the Internet every day. These data carry various kinds of information. Among these data, digital images may contain some sensitive information, such as personal handwritten [1] signatures, medical files, commercial data, and military intelligence. When these images are intercepted or tampered with during transmission, the owners of these images will disclose their privacy information. Therefore, how to protect this information is a valuable research project. Image encryption usually means that the sender converts plaintext image information into meaningless information, such as noise-like images, and transmits it to the receiver for decryption to ensure the safe transmission of the image. Compared with text information, image data has the characteristics of large amount of data and strong correlation in spatial domain. Traditional general algorithms for text encryption, such as DES, AES and IDEA, are not suitable for image encryption [2-4]. Due to the special characteristics of chaotic system, such as sensitivity to initial conditions and internal randomness, image encryption algorithm based on chaos has become one of the ideal encryption methods [5]. Because the dynamic characteristics of chaotic systems are very

DOI: 10.24507/ijicic.18.01.73

similar to traditional cryptography, Fridrich [6] has proposed a general structure of chaotic image encryption systems of 'scrambling and diffusion'. And chaotic cryptography began to arouse scholars' research interest in image encryption.

The typical image encryption process based on chaotic systems is to generate pseudo-random sequences, and then use pseudo-random sequences for scrambling and diffusion [7]. According to the combination of simple chaotic maps and complex chaotic systems, an image encryption algorithm was proposed [8]. The image encryption algorithm can further enhance the sensitivity of the algorithm and enhance the security of the encryption system. The complexity of the encryption algorithm was increased by increasing the scrambling and diffusion rounds or by use of more complex chaotic maps to generate chaotic sequences [9].

According to the principle of cryptography, the security of encryption algorithm is closely related to its key space. Thus, the key space of an encryption algorithm must be large enough to resist brute force attacks under the existing computing power. The traditional one-dimensional chaotic map, such as Logistic map [10], has the disadvantage of uneven distribution, which leads to the poor encryption effect of the image encryption algorithm based on a one-dimensional chaotic map. The key space of the system is small, which makes the algorithm unable to resist all kinds of violent attacks. The common method to expand the key space of encryption system is to design a high-dimensional chaotic system or generate the key by combining hash functions.

Typically, confusion is the only nonlinear component in a cryptosystem preventing an attacker from estimating the propagation of information from input to output [11]. The S-box is a nonlinear component in cryptography [12], which is used to implement obtrusion operations in cryptography and is used in many famous block cipher systems such as DES [2,13] and AES [3,14]. Recent studies show that it is a novel and promising direction to design S-boxes by using the chaotic systems. In recent years, a large number of S-box construction algorithms based on chaotic systems have been proposed [15-18]. According to a general model about an S-box, Zhang and Xiao [15] proposed an image encryption scheme. The algorithm can not only resist chosen-plaintext attack, but also has low complexity. An adaptive color encryption scheme was designed based on autonomous chaotic system, SHA-512, and two S-boxes [17]. Hasanzadeh and Yaghoobi [18] proposed a novel color image encryption scheme based on fractals, S-box and hyper-chaotic system. The results and security analysis show that it has some advantages of a good encryption scheme, such as large key space and high sensitivity to key. In addition, the algorithm can resist a variety of typical attacks.

Based on the above discussion and analysis, in order to improve the sensitivity of image encryption algorithm to key and resist various typical attacks, such as improving the ability of the algorithm to resist chosen-plaintext attack and known-plaintext attack, we propose a novel digital color image encryption algorithm by using a new 4-D hyper-chaotic system and an improved S-box. The contributions and innovations of this article are as follows. 1) Use a new improved 4-D hyper-chaotic system, which has more complex chaotic behavior than general chaotic systems. At the same time, we design the corresponding circuit of the chaotic system. 2) By introducing the S-box used in the encryption system and confusion operation, an improved chaotic S-box is constructed based on the hyper-chaotic system. 3) By using SHA-256 to calculate the hash value of the plaintext file to generate the system key, we greatly improve the ability of the image encryption algorithm to resist chosen-plaintext attack and known-plaintext attack.

This paper proposes a color digital image encryption algorithm based on a 4-D hyper-chaotic system and a new S-box. The rest of this article is organized as follows. Section 2 introduces the basic theory of the chaotic systems we use. In Section 3, the detailed

steps of the encryption process and decryption process of the proposed algorithm are shown. In Section 4, the simulation experiment results and security analysis of the proposed encryption algorithm are given. Section 5 gives some conclusions of the proposed algorithm.

2. Introduction to Chaotic Systems.

This part introduces Arnold map for image scrambling, the basic theory of our improved 4-D hyper-chaotic system, the circuit simulation of the hyper-chaotic system, and the randomness of binary pseudo-random generated by chaotic system is tested by NIST SP800-22 test suite.

2.1. Arnold map.

The Arnold map was proposed by V. I. Arnold in the research of ergodic theory in 1960s [19]. As shown in the following Formula (1), this is a chaotic map that performs repeated folding and stretching transformations in a limited area.

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod (1) \tag{1}$$

where "$x \bmod (1)$" means the fractional part of x for any real number x.

We can see that the security of Arnold map only depends on these initial values. In order to overcome this shortcoming, a generalized Arnold map [20] with two parameters is given as shown in the following Formula (2):

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & b_1 \\ a_1 & a_1 b_1 + 1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod (1) \tag{2}$$

where a_1 and b_1 are real numbers called control parameters, and we could conclude that the largest Lyapunov exponent of generalized Arnold map is larger than that of origin Arnold map when $a_1 > 1$, $b_1 > 1$. This shows that generalized Arnold map is in a strong sense of chaos, so it can better perform in data shuffling.

2.2. A 4-D hyper-chaotic system.

There are two or more positive Lyapunov exponents in a hyper-chaotic system. At the same time, its dynamics extend in many different directions. This also means that hyper-chaotic systems have more complex dynamics. Due to its theoretical and practical applications in technical fields such as secure communications, lasers, nonlinear circuits, neural networks, generation, control and synchronization, the researches on hyper-chaotic systems have attracted more and more attention. Cai et al. [21] established a new three-dimensional autonomous chaotic system equation. The system description equation is as follows:

$$\begin{cases} \dot{x} = a(y - x) \\ \dot{y} = bx + cy - xz \\ \dot{z} = x^2 - hz \end{cases} \tag{3}$$

where a, b, c, h are constants and x, y, z are state variables of the system. When $a = 20$, $b = 14$, $c = 10.6$ and $h = 2.8$, the Lyapunov exponents of this chaotic system are $LE1 = 2.3554$, $LE2 = 0$ and $LE3 = -14.5561$. And the LE dimension is 2.1618.

In this paper, we propose a new hyper-chaotic system based on the above 3-D chaotic system, which is described as follows:

$$\begin{cases} \dot{x} = a(y - x) \\ \dot{y} = bx + cy - xz + w \\ \dot{z} = x^2 - hz \\ \dot{w} = -kx + ew \end{cases} \tag{4}$$

where x, y, z and w are state variables of the hyper-chaotic system, a, b, c, e, h and k are system parameters, and w is a state feedback controller. We have analyzed the system and find that when the typical parameters are fixed as $a = 20$, $b = 1$, $c = 10.6$, $e = 0.45$, $h = 2.8$ and $k = 3.7$, the system is hyperchaotic. Its Lyapunov exponents are $LE1 = 0.80492$, $LE2 = 0.39485$, $LE3 = 0.00290$ and $LE4 = -12.953$. And the LE dimension is 3.0929. The chaotic attractor diagram is shown in Figure 1.

In order to verify the feasibility of the chaotic system in hardware, we use Multisim14.0 to design and simulate the chaotic system circuit. The circuit diagram calculated is shown in Figure 2, and the phase diagrams of x-y, x-z, x-w simulated are shown in Figure 3.

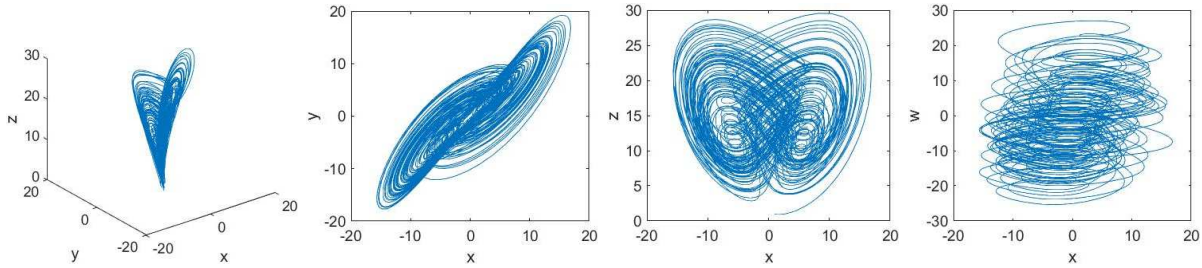


FIGURE 1. Simulation results: Projection on (a) x-y-z plane; (b) x-y plane; (c) x-z plane; (d) x-w plane

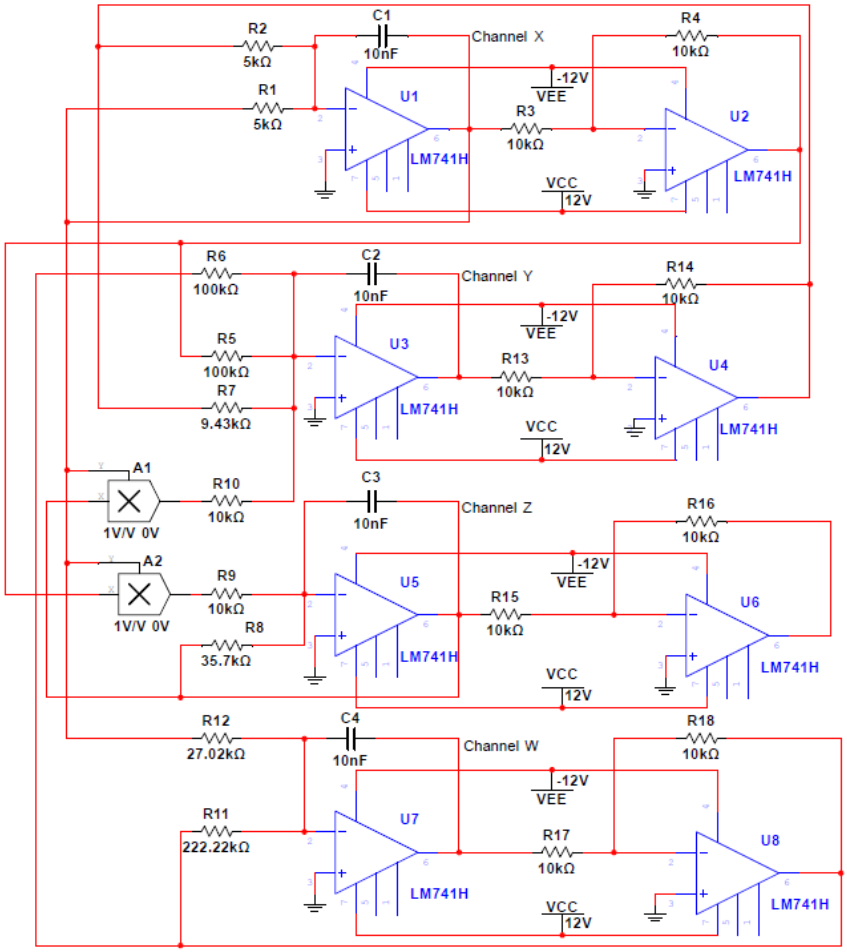
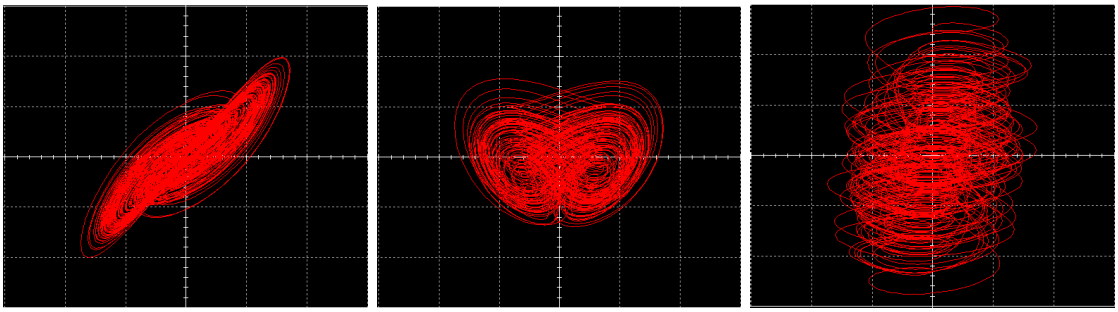


FIGURE 2. Simulation circuit of the hyper-chaotic system



(a) (b) (c)

FIGURE 3. Simulation results of Multisim virtual oscillograph: Projection on (a) x-y plane; (b) x-z plane; (c) x-w plane

2.3. **NIST SP800-22 test.** The NIST SP800-22 test is designed to evaluate the performance of the pseudo-random number generator (PRNG) [22]. It can judge whether chaotic binary sequence is suitable for image encryption algorithm. The NIST SP800-22 test consists of 15 tests, including approximate entropy test, block frequency test, random offset test, etc. The randomness of the test sequence can be measured by P-value. If $P \geq 0.01$, the sequence passes the test and is random. If $P < 0.01$, the sequence fails in the test and is not random. If $P = 1$, the sequence is completely random. If $P = 0$, the sequence is not random at all. According to the recommendation of [23], 100 binary streams with 100,000,000 bits are used as input data, and the generated P-values are expected to fall into the range of 0.01 and 1 to pass the test.

Table 1 shows the test results, which show that the binary stream generated by the hyper-chaotic system can pass all sub-tests, which shows that this chaotic system is suitable for image encryption.

TABLE 1. NIST SP800-22 test results

Sub-test items		P-value	Result
		0.01	
Approximate entropy ($m = 10$)		0.494392	Pass
Block frequency ($M = 128$)		0.319084	Pass
Cumulative sums	Forward	0.657933	Pass
	Reverse	0.055361	Pass
FFT		0.816537	Pass
Frequency		0.798139	Pass
Linear complexity ($M = 500$)		0.350485	Pass
Longest run		0.102526	Pass
Nonoverlapping template ($m = 9$)		0.508804	Pass
Overlapping template ($m = 9$)		0.595549	Pass
Random excursions		0.668811	Pass
Random excursions variant		0.571699	Pass
Rank		0.455937	Pass
Runs		0.075719	Pass
Serial ($m = 16$)	P-value1	0.554420	Pass
	P-value2	0.883171	Pass
Universal		0.616305	Pass

3. **The Encryption and Decryption Processes of the Proposed Algorithm.** This section presents the encryption and decryption processes of the proposed algorithm, including generating the key by SHA-256, the use of Arnold for pixel scrambling, the use of hyper-chaotic system to generate chaotic S-boxes, pixel byte substitution and image diffusion operations.

3.1. **Encryption process.** The encryption process is shown in Figure 4, and the operation steps are as follows.

Step 1: Get a color plaintext image I with the size of $M \times N$, and use the SHA-256 to calculate the hash value of the image file, and save it as the hash value K.

Step 2: Divide the 256-bit hash value K into 32 parts as $K = k_1, k_2, \ldots, k_{31}, k_{32}$, as shown in Figure 5. And we use every four parts to form a larger block, namely s1-s8. The eight blocks are used to generate the initial value of the chaotic system, which is mapped to X_0, Y_0, Z_0 and W_0 according to the following Formula (5), that is, as the initial value of the hyper-chaotic system, and the 32 parts of the hash value K are summed to calculate the block generation parameter P.

$$\begin{cases} X_0 = sum(K(1:4)/mean(K(5:8)))/4 \\ Y_0 = (sum(K(9:12)) - \max(K(13:16)))/4/256 \\ Z_0 = \max(bitxor(K(17:20), K(21:24)))/256 \\ W_0 = mean(bitxor(K(25:28), K(29:32)))/256 \\ P = sum(K(1:32)) \end{cases} \tag{5}$$

Step 3: Use the Arnold map according to the following Formula (6) to scramble all the pixels of R, G and B components of the plane image, and the scrambled image I_s obtained. The scrambling operation parameters are $a_1 = 3$, $b_1 = 5$, and the number of scrambling operations times $O = \mathrm{mod}(4 \times P, 64) + 50$.

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & b_1 \\ a_1 & a_1 b_1 + 1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod (O) \tag{6}$$

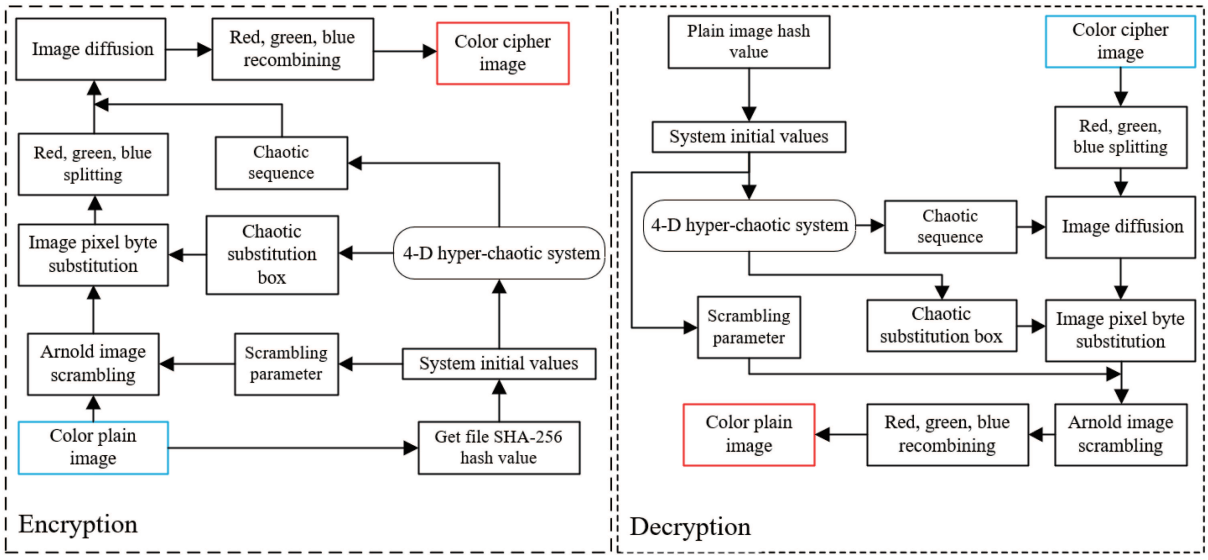


FIGURE 4. The encryption and decryption processes of the proposed algorithm

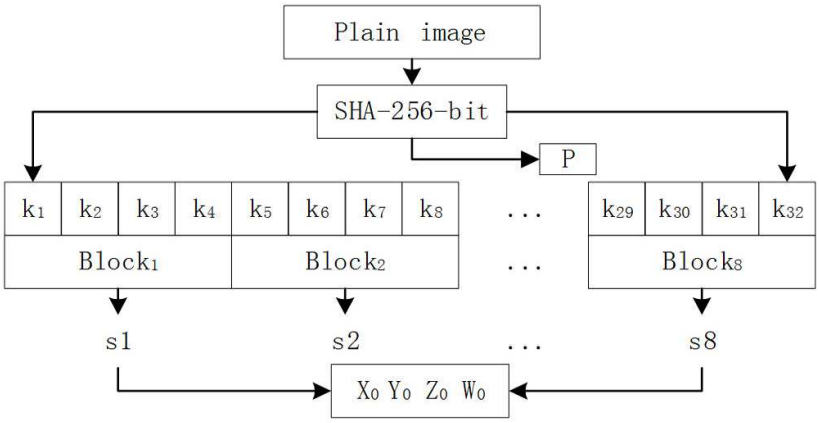


FIGURE 5. Schematic diagram of generating initial value and variable P

Step 4: The initial values of X_0, Y_0, Z_0 and W_0 are set in the hyper-chaotic system, and iterated $(10000 + M \times N)$ times to obtain the pseudo-random sequences X, Y, Z and W.

Step 5: Extract 256 points from the $3 \times P$ position from the sequence W to obtain the sequence W_z. After performing numerical amplification and modulo operations according to the following Formula (7), they are arranged in descending order to obtain the replacement index sequence W_{zb}. The standard S-box in the AES algorithm is used to perform byte substitution operation on the sequence W_{zb} to obtain the chaotic S-box for image encryption as shown in Figure 6.

$$\begin{cases} W_z = \mathrm{mod}\left(W_z \times 10^4, 256\right) \\ [W_z \;\; W_{zb}] = sort\left(W_z, \text{'}descend\text{'}\right) \\ s\_box = sub\_bytes\left(W_{zb}, aes\_s\_box\right) \end{cases} \tag{7}$$

where $sort()$ refers to a sorting function, '$sub\_bytes()$' represents a byte substitution function, 's_box' represents the generated chaotic S-box, and 'aes_s_box' represents the S-box of the AES algorithm.

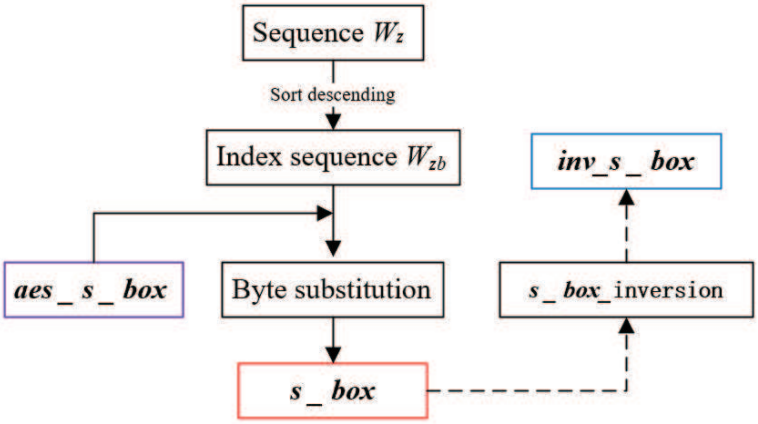


FIGURE 6. Schematic diagram of generating chaotic S-box and inverse box

Step 6: Use the generated chaotic S-box on the image I_s to do substitution operation after Arnold scrambling, and then obtain the substitutionary image I_{su}. From the iterated sequence X, Y, and Z, extract the data which are starting from the parameter P with length $M \times N$ respectively, to form the sequence X_z, Y_z, Z_z, and then expand the data

by 10^8 and retain the decimal part to obtain the sequence X_{zb}, Y_{zb}, Z_{zb}, according to the following Formula (8).

$$\begin{cases} X_{zb} = 10^8 \times X_z - round\,(10^8 \times X_z) \\ Y_{zb} = 10^8 \times Y_z - round\,(10^8 \times Y_z) \\ Z_{zb} = 10^8 \times Z_z - round\,(10^8 \times Z_z) \end{cases} \tag{8}$$

Step 7: The three sequences X_{zb}, Y_{zb} and Z_{zb} are mapped to integers in the ranges from 0 to 255. We can obtain the sequences ep_x, ep_y and ep_z for the next image diffusion operation in the following Formula (9).

$$\begin{cases} ep_x = uint8\,(\mathrm{mod}\,(floor\,(10^5 \times abs(X_{zb}))\,, 256)) \\ ep_y = uint8\,(\mathrm{mod}\,(floor\,(10^5 \times abs(Y_{zb}))\,, 256)) \\ ep_z = uint8\,(\mathrm{mod}\,(floor\,(10^5 \times abs(Z_{zb}))\,, 256)) \end{cases} \tag{9}$$

Step 8: The substitutionary image I_{su} is split into three components, namely, I_{sur}, I_{sug} and I_{sub}. According to Formula (10), we can get three sequences, that is ec_r, ec_g and ec_b. Then the three sequences are merged into the final ciphertext I_{enc}.

$$\begin{cases} ec_r = ep_x \oplus I_{sur} \\ ec_g = ep_y \oplus I_{sug} \\ ec_b = ep_z \oplus I_{sub} \end{cases} \tag{10}$$

where the symbol $\oplus$ refers to XOR operation.

3.2. **Decryption process.** The step of decrypting the cipher image is the reverse process of image encryption, as shown in the flowchart in Figure 4.

Step 1: Input the encrypted color image I_{enc} with the size of $M \times N$, and the hash value K of the original plaintext image file.

Step 2: Similar to Step 2 of encryption process, the initial value and parameter P of chaotic system are obtained by hash value K.

Step 3: The sequence ep_x, ep_y and ep_z are obtained in the same way as the encryption Steps 6 and 7.

Step 4: In the same way as Step 8 of encryption, the encrypted color image I_{enc} is divided into RGB components and XOR operation is performed respectively to obtain the image I_{su} after having done original substitution operation.

Step 5: First, the chaotic S-box is obtained similar to the encryption Step 5, and then the inverse chaotic S-box is by taking the values of the elements of the S-box as indices. The original scrambled image I_s is obtained by pixel value substitution of the original image I_{su} with the chaotic inverse S-box.

Step 6: The original scrambled image I_{su} is restored by using the following inverse Arnold transform according to Formula (11), and other parameters are consistent with the encryption Step 3. The scrambling operation parameters are $a_1 = 3$, $b_1 = 5$, and the number of scrambling operations times $O = \mathrm{mod}(4 \times P, 64) + 50$. Finally, after reconstructing the restored RGB color components, the decrypted image I_{dec} can be obtained.

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} a_1 b_1 + 1 & -b_1 \\ -a_1 & 1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \mathrm{mod}\,(O) \tag{11}$$

4. **Experimental Results and Safety Analysis.** In order to verify the security performance of our proposed algorithm, we used Matlab R2018b to carry out experiments and analysis. All the experiments and analyses were carried out on the same computer. Three standard color images of Lena, Baboon, and Sailboat on lake with a size of 512×512

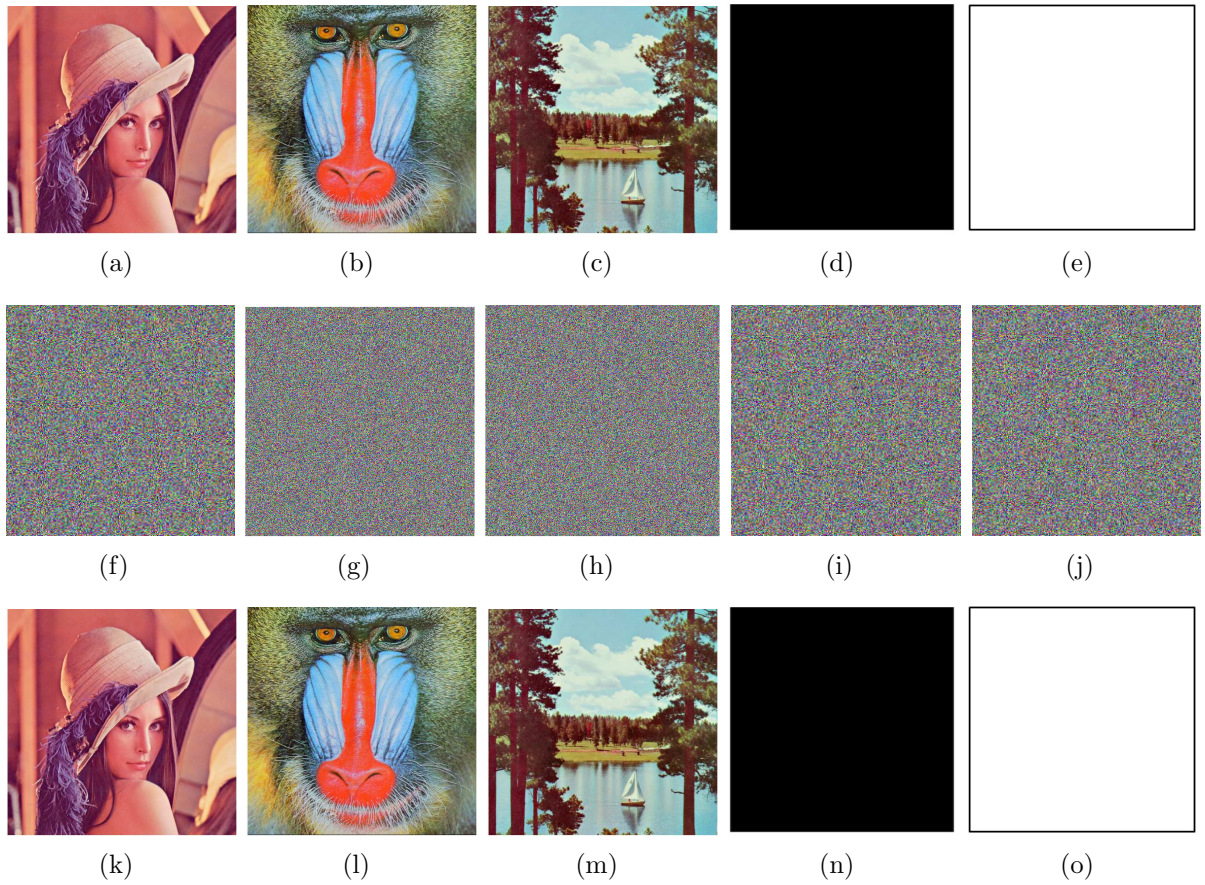


FIGURE 7. Plain image, cipher image, and decrypted image of the test images: (a) Lena; (b) Baboon; (c) Sailboat on lake; (d) All-black; (e) All-white; (f) Lena encrypted image; (g) Baboon encrypted image; (h) Sailboat on lake encrypted image; (i) All-black encrypted image; (j) All-white encrypted image; (k) Lena decrypted image; (l) Baboon decrypted image; (m) Sailboat on lake decrypted image; (n) All-black decrypted image; (o) All-white decrypted image

were selected from the USC-SIPI [10] image library as test images. And the same size images, All-black and All-white, are generated by Matlab. The simulation results are shown in Figure 7, which shows that the encryption algorithm is not limited to specific image information, and can effectively hide the information on the image intuitively.

4.1. **Key space.** A good encryption algorithm needs to have a large enough key space to resist brute-force attack. The key space needs to be at least 2^{100} [24]. The algorithm uses the SHA-256 to generate the initial key. SHA-256 can provide the key space of the 2^{128} anti-collision attack. At the same time, the algorithm has four main variable initial values. Assuming that the computer accuracy is 10^{15}, the effective key space of the algorithm is $2^{128} + 10^{15} \times 4 \approx 2^{237}$. Obviously, this algorithm has enough key space to resist violent attacks.

4.2. **Key sensitivity.** Chaotic maps are sensitive to initial values and are very suitable for encryption algorithms. The key of the algorithm is generated by SHA-256, and finally mapped to the initial values of chaos system iteration X_0, Y_0, Z_0, W_0 and parameter P. In order to test the key sensitivity of the algorithm, here we select several parameters of

the key control: X_0, Y_0, Z_0, W_0 to add a 10^{-16} perturbation, and a 1-bit change to the original hash value K. Finally, we try to decrypt encrypted Lena images using these keys and the correct key.

K = c056da23302d2fb0d946e7ffa11e0d94618224193ff6e2f78ef8097bb8a3569b
$K + 1$ = c056da23302d2fb0d946e7ffa11e0d94618224193ff6e2f78ef8097bb8a3569c
Key 1: $K + 1$, X_0, Y_0, Z_0, W_0
Key 2: K, $X_0 + 10^{-16}$, Y_0, Z_0, W_0
Key 3: K, X_0, $Y_0 + 10^{-16}$, Z_0, W_0
Key 4: K, X_0, Y_0, $Z_0 + 10^{-16}$, W_0
Key 5: K, X_0, Y_0, Z_0, $W_0 + 10^{-16}$

The simulation results are shown in Figure 8. We cannot decrypt the cipher image successfully as long as one of control parameters changes slightly. The results show that the algorithm has high key sensitivity.

FIGURE 8. Decrypted images based on different keys: (a) decrypted by key 1; (b) decrypted by key 2; (c) decrypted by key 3; (d) decrypted by key 4; (e) decrypted by key 5; (f) decrypted by correct key

4.3. **Histogram analysis.** The image histogram represents the intensity distribution of pixels in the image, and an encrypted image histogram showing a flat distribution can resist potential statistical analysis [25]. Figure 9 shows the histogram of the RGB color channels of the original image Lena and the encrypted image. It can be seen from Figure 9 that the pixel values of the R, G, and B channels of the original image are concentrated on some values, while the histograms of the three RGB channels of the encrypted image are very flat. Obviously, the encrypted image hides the histogram statistical characteristics of the original image, and it is difficult for an attacker to extract useful information from the encrypted image.

4.4. **Differential attack analysis.** Differential attack is a kind of chosen-plaintext attacks. Differential attack used by the attacker tries to find the connection between plaintext and ciphertext by tracing the influence of subtle changes of plaintext on ciphertext. The established connection can be used to recover ciphertext without a key. So, an effective algorithm needs to be highly sensitive to the input data. If subtle changes in the

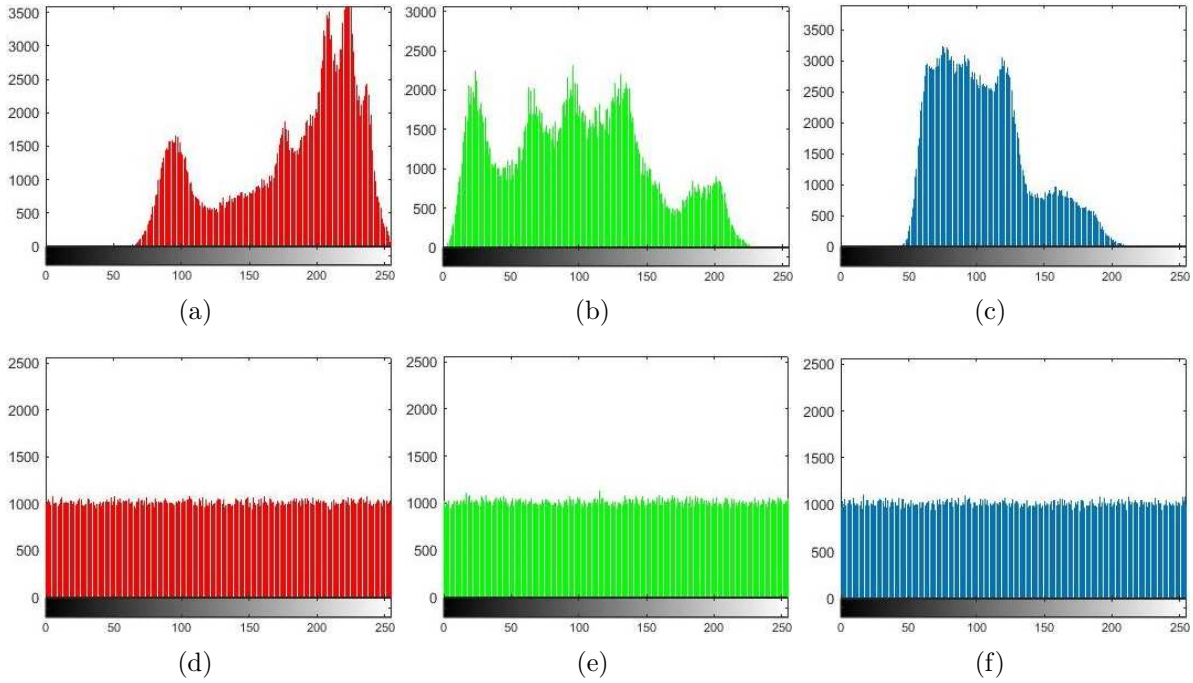


FIGURE 9. Histograms analysis for Lena image: (a) R component before encryption; (d) R component after encryption; (b) G component before encryption; (e) G component after encryption; (c) B component before encryption; (f) B component after encryption

input data result in a huge change in the output ciphertext, then this differential attack is usually meaningless. In order to test and qualitatively analyze the resistance of our proposed encryption scheme to differential attacks, we adopted UACI (unified average change intensity) and NPCR (number of pixels change rate) standards [26], which are defined as Formula (13) and Formula (14).

$$D(i,j) = \begin{cases} 0, & \text{if } C_1(i,j) = C_2(i,j) \\ 1, & \text{if } C_1(i,j) \neq C_2(i,j) \end{cases} \tag{12}$$

$$UACI(C_1, C_2) = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{|C_1(i,j) - C_2(i,j)|}{255} \times 100\% \tag{13}$$

$$NPCR(C_1, C_2) = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} D(i,j) \times 100\% \tag{14}$$

where C_1 and C_2 are two encrypted images, which are obtained by encrypting two images with only one bit difference.

In each test, we randomly change one-bit of the original image to get a slightly different plaintext image, then encrypt the changed ciphertext image, and calculate the NPCR and UACI of all the color components. We conduct 100 tests on each test image, and finally get 100 groups of NPCR and UACI data. These NPCR and UACI data are averaged, and classified according to the R, G, and B color components, as shown in Table 2. The NPCR and UACI values are very close to the expected values of 99.6094% and 33.4635% [27] and thus the proposed image encryption technique shows good sensitivity to plaintext and is invulnerable to differential attacks.

TABLE 2. The NPCR% and UACI% of ciphered image with random change one bit

Test image	Average NPCR (%)			Average UACI (%)		
	Red	Green	Blue	Red	Green	Blue
Lena	99.6090	99.6110	99.6083	33.4613	33.4666	33.4827
Baboon	99.6107	99.6107	99.6090	33.4731	33.4731	33.4875
Sailboat on lake	99.6075	99.6100	99.6104	33.4498	33.4498	33.4800
All-black	99.6095	99.6097	99.6083	33.4747	33.4591	33.4724
All-white	99.6077	99.6111	99.6091	33.4503	33.4555	33.4691

In 2011, Wu et al. [28] proposed a stricter criterion of NPCR and UACI. As for a significance level β, the critical NPCR score N^*_β can be expressed by

$$N^*_\beta = \left(F - \varphi^{-1}(\beta)\sqrt{F/MN}\right)/(F+1) \tag{15}$$

where $\varphi^{-1}(\cdot)$ is the inverse cumulative density function (CDF) of the standard normal distribution, and F represents the gray level of the image. For example, F is equal to 1 when the test image is a binary image. Similarly, F is equal to 255 when the test image is an 8-bit image. If the obtained values of an image encryption algorithm are larger than N^*_β, it can be considered to pass this test. The critical UACI interval $\left(U^{*-}_\beta, U^{*+}_\beta\right)$ can be calculated by

$$\begin{cases} U^{*-}_\beta = \mu_u - \varphi^{-1}(\beta/2)\sigma_u \\ U^{*+}_\beta = \mu_u + \varphi^{-1}(\beta/2)\sigma_u \end{cases} \tag{16}$$

where $\mu_u = (F+2)/(3F+3)$, $\sigma_u^2 = (F+2)\left(F^2+2F+3\right)/\left(18(F+1)^2MNF\right)$. If the calculations of an image encryption algorithm fall into the interval $\left(U^{*-}_\beta, U^{*+}_\beta\right)$, it can be considered to pass the test. The theoretical values of critical NPCR and UACI for images with different sizes and significance levels respectively are shown in Table 3.

TABLE 3. The theoretical values of critical NPCR and UACI

Image size	NPCR			UACI		
	$N^*_{0.05}$	$N^*_{0.01}$	$N^*_{0.001}$	$U^{*-}_{0.05}, U^{*+}_{0.05}$	$U^{*-}_{0.01}, U^{*+}_{0.01}$	$U^{*-}_{0.001}, U^{*+}_{0.001}$
$256 * 256$	99.5693	99.5527	99.5341	(33.2824, 33.6447)	(33.2255, 33.7016)	(33.1594, 33.7677)
$512 * 512$	99.5893	99.5810	99.5717	(33.3730, 33.5541)	(33.3445, 33.5826)	(33.3115, 33.6156)
$1024 * 1024$	99.5994	99.5952	99.5906	(33.4183, 33.5088)	(33.4040, 33.5231)	(33.3875, 33.5396)

TABLE 4. The critical NPCR (%) of ciphered image with random change one bit

Test image	Average NPCR (%)			Critical NPCR (%)		
	Red	Green	Blue	$N^*_{0.05}$	$N^*_{0.01}$	$N^*_{0.001}$
Lena	99.6090	99.6110	99.6083	Passed	Passed	Passed
Baboon	99.6107	99.6107	99.6090	Passed	Passed	Passed
Sailboat on lake	99.6075	99.6100	99.6104	Passed	Passed	Passed
All-black	99.6095	99.6097	99.6083	Passed	Passed	Passed
All-white	99.6077	99.6111	99.6091	Passed	Passed	Passed

TABLE 5. The critical UACI (%) of ciphered image with random change one bit

Test image	Average UACI (%)			Critical UACI (%)		
	Red	Green	Blue	$U^{*-}_{0.05}, U^{*+}_{0.05}$	$U^{*-}_{0.01}, U^{*+}_{0.01}$	$U^{*-}_{0.001}, U^{*+}_{0.001}$
Lena	33.4613	33.4666	33.4827	Passed	Passed	Passed
Baboon	33.4731	33.4731	33.4875	Passed	Passed	Passed
Sailboat on lake	33.4498	33.4498	33.4800	Passed	Passed	Passed
All-black	33.4747	33.4591	33.4724	Passed	Passed	Passed
All-white	33.4503	33.4555	33.4691	Passed	Passed	Passed

TABLE 6. NPCRs and UACIs of the Lena image (512×512) for different methods

Index	Ideal value	Channel	Ref. [26]	Ref. [27]	Ref. [29]	Ref. [30]	Ref. [31]	Ours
NPCR (%)	99.6094	R	99.6086	99.6037	99.6052	99.62432	99.6078	99.6090
		G	99.6051	99.5983	99.6060	99.61850	99.6088	99.6110
		B	99.6116	99.6159	99.6113	99.62807	99.6081	99.6083
UACI (%)	33.4635	R	33.4629	33.4290	33.4280	33.42243	33.4291	33.4613
		G	33.4840	33.4306	33.4966	33.43615	33.4252	33.4666
		B	33.4880	33.3665	33.3779	33.46037	33.4219	33.4827

Combined with the data in Table 2, it can be seen from Table 4 that all the NPCR values are larger than N^*_β, and close to the theoretical value of 99.6094%, and UACI scores shown in Table 5 are all within the accepted intervals $\left(U^{*-}_\beta, U^{*+}_\beta\right)$. Thus, the algorithm meets the design requirements of expected resistance to differential attack.

Table 6 shows the performance comparison with other algorithms about NPCR and UACI. The comparison proves that our algorithm is very sensitive to the changes of pixels in the plain image, and the algorithm is closer to the ideal value while passing the critical test standards of NPCR and UACI.

4.5. **Clipping attack analysis.** Clipping attack is a common attack on encrypted image. Encrypted image may be maliciously damaged at any time in the process of transmission, which will cause some permanent damage to the encrypted image information. Image encryption algorithm should be able to resist cropping attack. In this section, Lena is the test image. The encrypted ciphertext image is subjected to 1/16, 1/8, 1/4 and 1/2 clipping attacks respectively, and then the damaged ciphertext image is decrypted respectively. The simulation results are shown in Figure 10. Even if the encrypted image is attacked by 1/2 cropping, the decrypted image can still be roughly observed with the naked eye. The simulation results show that the algorithm can resist clipping attack well.

4.6. **Correlation analysis.** The correlation between adjacent pixels of an image reflects the degree of correlation between pixel values of adjacent positions of an image. A good image encryption algorithm should make the correlation between adjacent pixels of the encrypted image significantly reduced, and should try to achieve zero correlation, so as to protect the original information. This algorithm analyzes the correlation of all the adjacent pixels, and calculates the correlation coefficients of the adjacent pixels in the horizontal, vertical and diagonal directions. The formulas for calculating the correlation coefficient are as follows:

$$r_{xy} = \operatorname{cov}(x, y) \Big/ \sqrt{D(x)D(y)} \tag{17}$$

$$\operatorname{cov}(x, y) = \frac{1}{N} \sum_{i=0}^{N} (x_i - E(x))(y_i - E(y)) \tag{18}$$

$$D(x) = \frac{1}{N} \sum_{i=0}^{N} (x_i - E(x))^2, \quad E(x) = \frac{1}{N} \sum_{i=0}^{N} x_i \tag{19}$$

where x and y are the gray values of two adjacent pixels, $E(x)$ is the average, $D(x)$ is the variance, and $\operatorname{cov}(x, y)$ is the covariance. The correlation coefficient of the three directions of the plain image and the encrypted image is shown in Table 7, and the comparison of other encryption algorithms is listed in Table 9. The experimental results show that the encryption algorithm can protect the plain image information effectively, and so it is an eligible encryption algorithm. It can be seen that the correlation coefficient of the adjacent pixels in the three directions of the three channels of the original image is close to 1, reflecting the high correlation of the adjacent pixels in the plain image, while the correlation coefficient of the adjacent pixels in the three directions of the RGB channels of the encrypted image is close to zero.

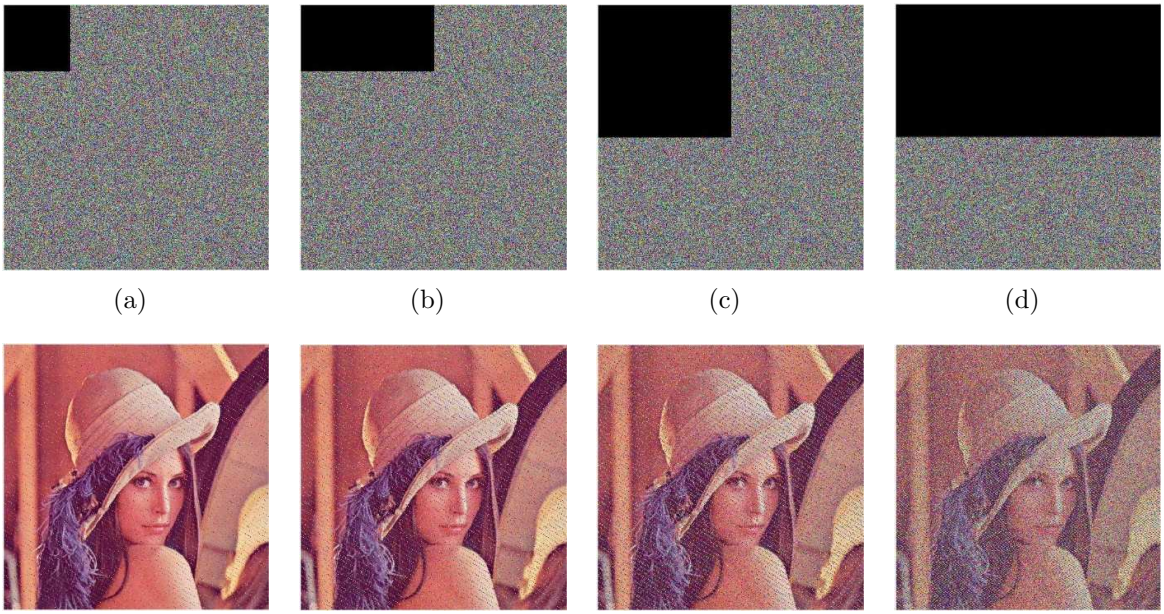


FIGURE 10. Simulation results of the tailoring attack: (a) cipher image (1/16 occlusion); (b) cipher image (1/8 occlusion); (c) cipher image (1/4 occlusion); (d) cipher image (1/2 occlusion); (e) recovered image (1/16 occlusion); (f) recovered image (1/8 occlusion); (g) recovered image (1/4 occlusion); (h) recovered image (1/2 occlusion)

The correlations between the original image of Lena and the corresponding cipher image are shown in Figure 11 and Figure 12. For convenience of illustration, the drawing selects 10000 random pixel points for illustration. It can be seen that the adjacent pixel pairs in three directions of RGB channels of the common image are densely distributed. The adjacent pixel pairs in three directions of RGB channels are distributed uniformly in the region after encryption. This shows that the encryption algorithm effectively reduces the correlation between adjacent pixels.

TABLE 7. Correlation coefficients of two adjacent pixels in the plain and cipher images

Images	Channel	Plain-image			Cipher-image		
		Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Lena	R	0.97748	0.98801	0.97371	−0.00215	0.00276	−0.00032
	G	0.96615	0.98170	0.96048	0.00179	0.00232	0.00102
	B	0.93041	0.95678	0.92187	0.00121	−0.00113	0.00069
Baboon	R	0.92176	0.86243	0.85310	−0.00270	−0.00099	0.00109
	G	0.86434	0.75914	0.72986	−0.00012	0.00247	0.00094
	B	0.90712	0.87823	0.84115	0.00002	0.00326	0.00025
Sailboat on lake	R	0.95438	0.95287	0.93962	0.00168	−0.00086	0.00147
	G	0.96916	0.96265	0.95200	0.00331	0.00082	−0.00012
	B	0.96895	0.96879	0.95206	0.00022	0.00374	0.00011
All-black	R	\	\	\	−0.00067	−0.00197	−0.00096
	G	\	\	\	0.00149	−0.00165	−0.00068
	B	\	\	\	0.00052	−0.00009	0.00172
All-white	R	\	\	\	−0.00152	0.00108	−0.00285
	G	\	\	\	0.00048	−0.00010	−0.00045
	B	\	\	\	0.00044	0.00044	0.00108

TABLE 8. Information entropy in the plain and cipher images

Image type	Channel	Lena	Baboon	Sailboat on lake	All-black	All-white
Plain-image	R	7.25310	7.70667	7.31239	0	0
	G	7.59403	7.47443	7.64285	0	0
	B	6.96843	7.75222	7.21364	0	0
Cipher-image	R	7.99942	7.99923	7.99929	7.99928	7.99929
	G	7.99929	7.99933	7.99926	7.99928	7.99925
	B	7.99929	7.99942	7.99926	7.99928	7.99936

TABLE 9. Performance of the proposed scheme and other methods

Algorithms	Channel	Correlation coefficient			Entropy
		Horizontal	Vertical	Diagonal	
Ours	R	−0.00215	0.00276	−0.00032	7.99942
	G	0.00179	0.00232	0.00102	7.99929
	B	0.00121	−0.00113	0.00069	7.99929
Ref. [27]	R	−0.001854	−0.005394	−0.029268	7.999281
	G	−0.021045	−0.050137	0.001236	7.999337
	B	0.007067	0.001908	−0.009406	7.999335
Ref. [30]	R	0.001365	0.004776	0.000232	7.99171
	G	0.003294	−0.000579	0.004807	7.99121
	B	0.002060	0.000194	−0.004043	7.99177
Ref. [31]	R	−0.0073	0.0010	−0.0013	7.9966
	G	0.0011	−0.0020	0.0078	7.9972
	B	−0.0061	0.0058	−0.0003	7.9967
Ref. [33]	R	−0.0001	0.0026	−0.0053	7.9974
	G	−0.0011	0.0009	0.0026	7.9969
	B	−0.0010	−0.0030	−0.0051	7.9979

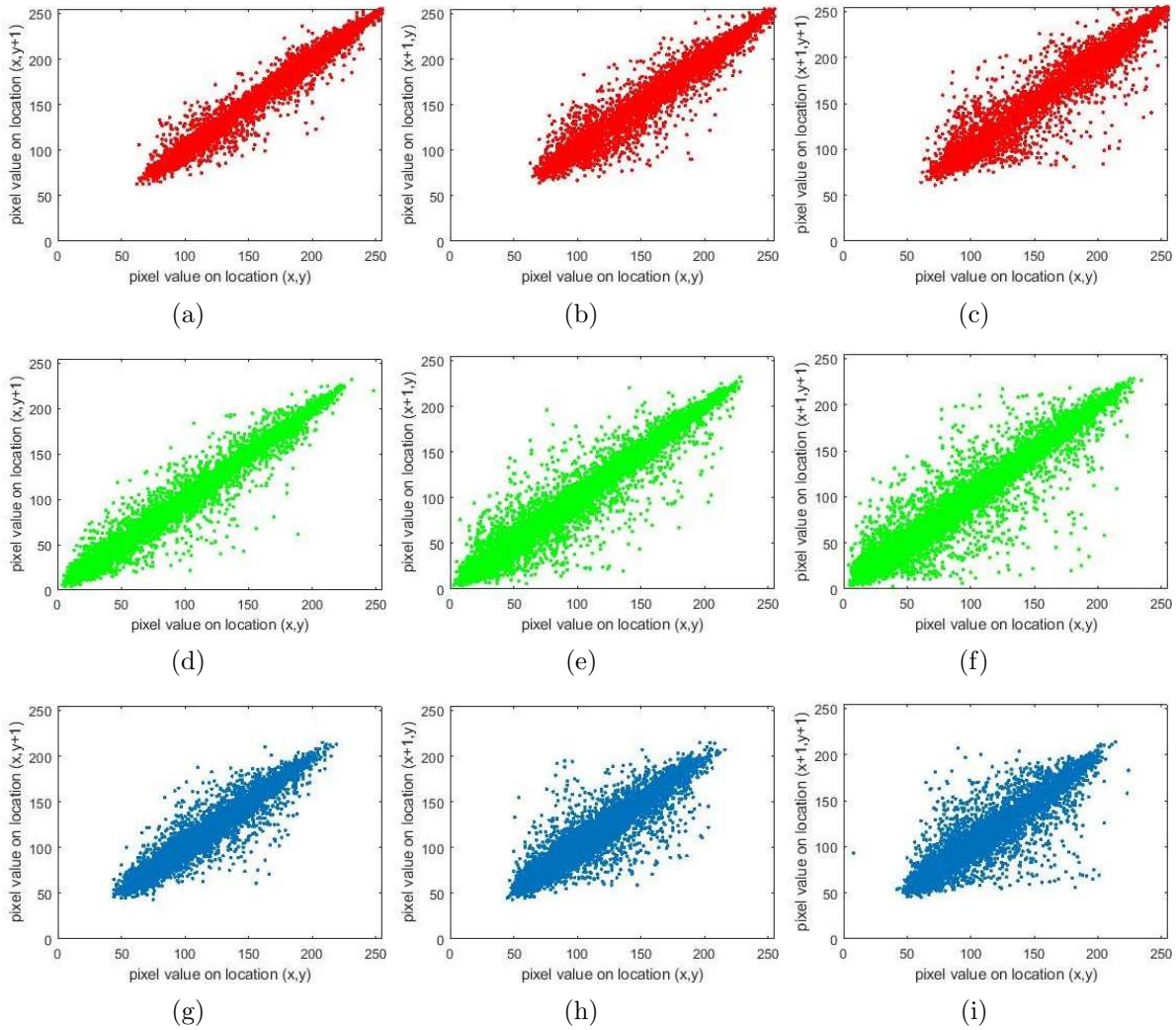


FIGURE 11. Correlation of adjacent pixels before Lena image encryption: (a)-(c) Horizontal direction, vertical direction and diagonal direction correlations of R component, (d)-(f) three directions of the G component and (g)-(i) three directions of the B component

4.7. **Information entropy analysis.** In order to resist various security attacks, the pixels of encrypted image need to be randomly distributed. Shannon's information theory [32] puts forward the concept of information entropy. The specific mathematical definition of information entropy is as follows:

$$H(q) = -\sum_{i=0}^{2^n-1} P(q_i) \log_2[P(q_i)] \tag{20}$$

where $P(q_i)$ is the probability of q_i and 2^n is the gray level of the image. Theoretically, 256 level gray image has 2^8 kinds of gray value possibilities. Therefore, the theoretical value of information entropy can be calculated. The ideal value of entropy is 8.

Table 8 shows the information entropy of the normal image and the encrypted image of the test image. As can be seen from Table 9, compared with other algorithms, the information entropy of this algorithm is closer to the theoretical value which indicates that the randomness of the generated ciphertext image is better. The results show that the algorithm can effectively resist the information entropy analysis.

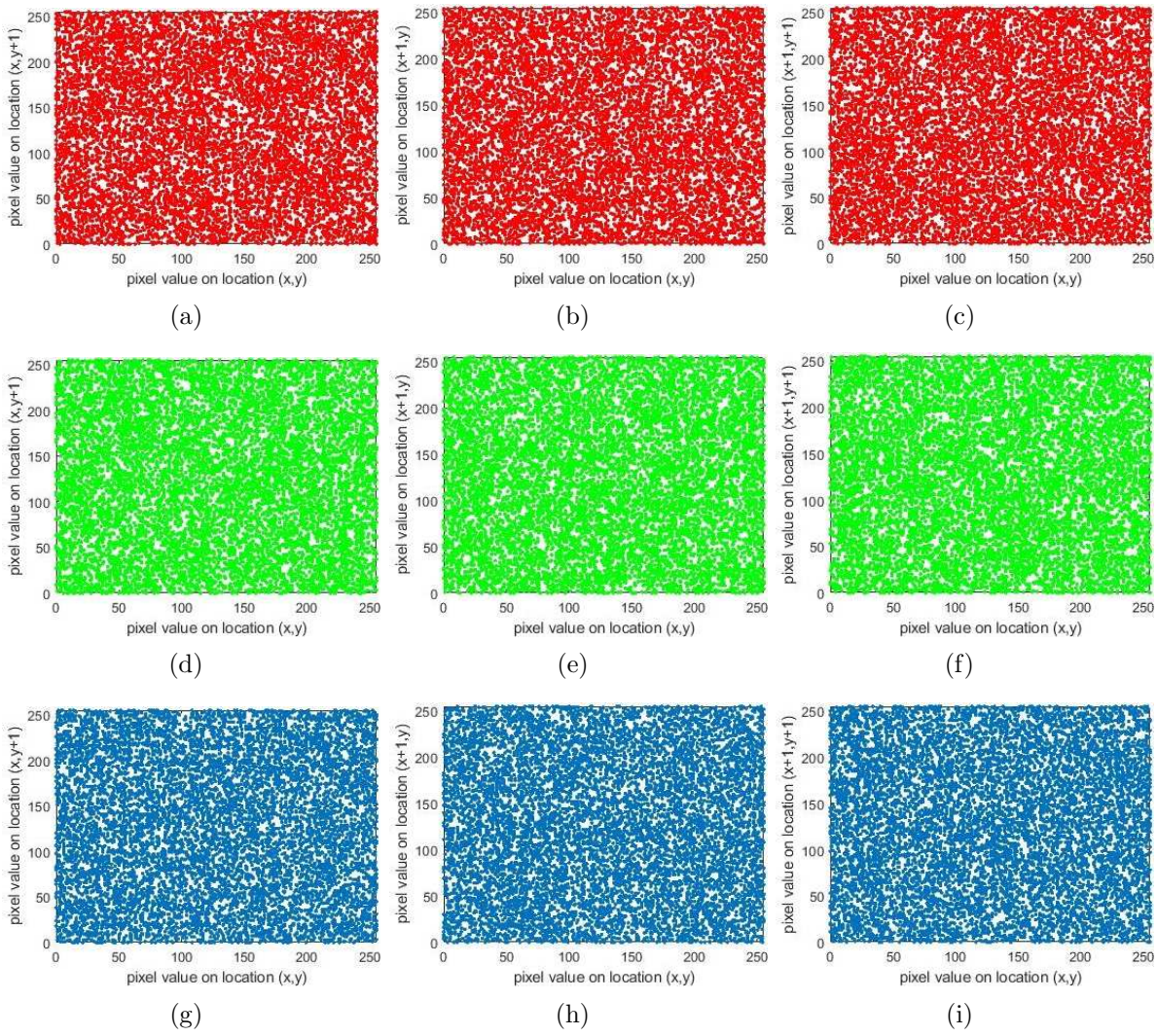


FIGURE 12. Correlation of adjacent pixels after Lena image encryption: (a)-(c) Horizontal direction, vertical direction and diagonal direction correlations of R component, (d)-(f) three directions of the G component and (g)-(i) three directions of the B component

5. **Conclusions.** In this paper, an improved 4-D hyper-chaotic system is proposed. The circuit simulation and NIST SP800-22 test show that the hyper-chaotic system can be used for image encryption. Then we propose a color digital image encryption algorithm. In order to improve the ability of the algorithm to resist plaintext attack, SHA-256 hash algorithm is used to calculate the hash value of plaintext image file and generate the initial key of the system. And then the initial key is generated, and the Arnold map is used to scramble the original image. The chaotic S-box is obtained by the pseudo-random sequence from the hyper-chaotic system, and the scrambled image is substituted. Finally, the replaced image and the pseudo-random sequence are diffused to generate the ciphertext image. According to the simulation results and some specific security performance analysis, the image encryption algorithm can effectively hide the plaintext information and resist the current common differential attack, clipping attack, selected plaintext attack and other attacks. To sum up, the designed algorithm has excellent encryption performance.

Acknowledgment. This research was funded by National Natural Science Foundation of China under Grant No. 61573004, and Quanzhou City Science & Technology Program of China under Grant No. 2018C106R.

REFERENCES

[1] T. Otsuki, T. Takahashi, S. Nakamura, H. Orii and H. Kawano, Image correction method to beautify Japanese handwritten characters using generative adversarial networks, *ICIC Express Letters, Part B: Applications*, vol.12, no.7, pp.645-650, 2021.

[2] D. Coppersmith, The data encryption standard (DES) and its strength against attacks, *IBM J. Res. Dev.*, vol.38, no.3, pp.243-250, 1994.

[3] S. Heron, Advanced encryption standard (AES), *Netw. Secur.*, vol.2009, no.12, pp.8-12, 2009.

[4] S. Toughi, M. H. Fathi and Y. A. Sekhavat, An image encryption scheme based on elliptic curve pseudo random and advanced encryption system, *Signal Processing*, vol.141, pp.217-227, 2017.

[5] Y. C. Zhou, Z. Y. Hua, C. M. Pun and C. L. P. Chen, Cascade chaotic system with applications, *IEEE Trans. Cybernetics*, vol.45, no.9, pp.2001-2012, 2015.

[6] J. Fridrich, Symmetric ciphers based on two-dimensional chaotic maps, *Int. J. Bifurcation Chaos Appl. Sci. Eng.*, vol.8, no.6, pp.1259-1284, 1998.

[7] Z. Y. Hua, F. Jin, B. Xu and H. Huang, 2D logistic-sine-coupling map for image encryption, *Signal Processing*, vol.149, pp.148-161, 2018.

[8] W. C. Qiu and S. J. Yan, An image encryption algorithm based on the combination of low-Dimensional chaos and high-dimensional chaos, *IEEE Int. Conf. Electron. Inf. Technol. Comput. Eng. (EITCE)*, pp.684-687, 2019.

[9] H. Liu, Y. Zhang, A. Kadir and Y. Xu, Image encryption using complex hyper-chaotic system by injecting impulse into parameters, *Appl. Math. Comput.*, vol.360, no.3, pp.83-93, 2019.

[10] N. K. Pareek, V. Patidar and K. K. Sud, Image encryption using chaotic logistic map, *Image Vis. Comput.*, vol.24, no.9, pp.926-934, 2006.

[11] Z. B. Faheem, A. Ali, M. A. Khan, M. E. Ul-Haq and W. Ahmad, Highly dispersive substitution box (S-box) design using chaos, *ETRI J.*, vol.42, no.4, 2020.

[12] X. Zhang, Z. Zhao and J. Wang, Chaotic image encryption based on circular substitution box and key stream buffer, *Signal Processing Image Communication*, vol.29, no.8, pp.902-913, 2014.

[13] E. F. Brickell, J. H. Moore and M. R. Purtill, Structure in the S-boxes of the DES, *Advances in Cryptology – CRYPTO'86*, Santa Barbara, CA, USA, 1986.

[14] J. Daemen and V. Rijmen, The design Rijndael: AES – The advanced encryption standard, *Information Security and Cryptography*, 2002.

[15] Y. Zhang and D. Xiao, Cryptanalysis of S-box-only chaotic image ciphers against chosen plaintext attack, *Nonlinear Dyn.*, vol.72, no.4, pp.751-756, 2013.

[16] A. Belazi, M. Khan, A. A. El-Latif and S. Belghith, Efficient cryptosystem approaches: S-boxes and permutation – Substitution-based encryption, *Nonlinear Dyn.*, vol.87, no.1, pp.337-361, 2016.

[17] H. Liu, A. Kadir, X. Sun and Y. Li, Chaos based adaptive double-image encryption scheme using hash function and S-boxes, *Multimed. Tools Appl.*, vol.77, no.2, pp.1391-1407, 2018.

[18] E. Hasanzadeh and M. Yaghoobi, A novel color image encryption algorithm based on substitution box and hyper-chaotic system with fractal keys, *Multimed. Tools Appl.*, vol.79, no.4, pp.7279-7297, 2020.

[19] H. Zhu, C. Zhao, X. Zhang and L. Yang, An image encryption scheme using generalized Arnold map and affine cipher, *Optik*, vol.125, no.22, pp.6672-6677, 2014.

[20] R. S. Ye, A novel chaos-based image encryption scheme with an efficient permutation-diffusion mechanism, *Opt. Commun.*, vol.284, no.22, pp.5290-5298, 2011.

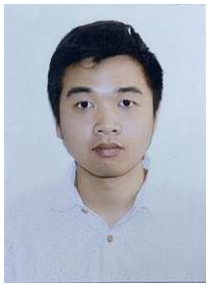
[21] G. L. Cai, Z. M. Tan, W. H. Zhou and W. T. Tu, Dynamical analysis of a new chaotic system and its chaotic control, *Acta Phys. Sin.*, vol.56, no.11, pp.6230-6237, 2007.

[22] C. Fan, Q. Ding and C. K. Tse, Counteracting the dynamical degradation of digital chaos by applying stochastic jump of chaotic orbits, *Int. J. Bifurcation Chaos*, vol.29, DOI: 10.1142/S0218127419300234, 2019.

[23] L. Bassham, A. Rukhin, J. Soto, J. Nechvatal, M. Smid, S. Leigh, M. Levenson, M. Vangel, N. Heckert and D. Banks, *SP800-22 Rev. 1a. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*, Tech. Rep., SP800-22, National Institute of Standards & Technology, 2010.

[24] C. H. Li, G. C. Luo, K. Qin and C. Li, An image encryption scheme based on chaotic tent map, *Nonlinear Dyn.*, vol.87, no.1, pp.127-133, 2016.
[25] Y. C. Zhou, W. J. Cao and C. L. P. Chen, Image encryption using binary bitplane, *Signal Processing*, vol.100, pp.197-207, 2014.
[26] X. Y. Wang and X. Chen, An image encryption algorithm based on dynamic row scrambling and Zigzag transformation, *Chaos, Solitons & Fractals*, vol.147, DOI: 10.1016/j.chaos.2021.110962, 2021.
[27] X. Y. Wang, X. M. Qin and C. M. Liu, Color image encryption algorithm based on customized globally coupled map lattices, *Multimed. Tools Appl.*, vol.78, pp.6191-6209, 2019.
[28] Y. Wu, J. P. Noonan and S. Agaian, NPCR and UACI randomness tests for image encryption, *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT)*, vol.4, pp.31-38, 2011.
[29] X. J. Wu, J. Kurths and H. B. Kan, A robust and lossless DNA encryption scheme for color images, *Multimed. Tools Appl.*, vol.77, no.10, pp.12349-12376, 2018.
[30] Y.-Q. Zhang, Y. He, P. Li and X.-Y. Wang, A new color image encryption scheme based on 2DNLC-ML system and genetic operations, *Opt. Lasers Eng.*, vol.128, DOI: 10.1016/j.optlaseng.2020.106040, 2020.
[31] A. U. Rehman, X. F. Liao, R. Ashraf, S. Ullah and H. Wang, A color image encryption technique using exclusive-OR with DNA complementary rules based on chaos theory and SHA-2, *Optik*, vol.159, pp.348-367, 2018.
[32] C. E. Shannon, A mathematical theory of communication, *Bell System Technical Journal*, vol.27, no.3, pp.379-423, 1948.
[33] A. Girdhar and V. Kumar, A RGB image encryption technique using Lorenz and Rossler chaotic system on DNA sequences, *Multimed. Tools Appl.* vol.77, pp.27017-27039, 2018.

Author Biography



Jinyuan Chen is currently pursuing the M.S. degree with the College of Engineering, Huaqiao University, Quanzhou, China. His research interests include information security, chaotic synchronization and control, and the application of chaotic cryptography in image processing.

Jianeng Tang received the B.Sc. degree in electronic information science and technology from Xijiang Normal University, China, 2006; the M.Sc. degree in circuits and systems from Ningxia University, China, 2009; the Ph.D. degree in information and communication engineering from Southeast University, China, 2012.

Jianeng Tang is currently an associate professor at College of Engineering, Huaqiao University, China. His research interests include image encryption, RF circuit design, complex network synchronization, and chaos synchronization and control. He has published over 30 papers in journals and conferences.

Feng Zhang received the B.Sc. degree in applied physics from University of Electronic Science and Technology of China, China, 2007. He is currently a deputy general manager of Fujian MM Electronics Co., Ltd. His research interests include image encryption and RF circuit design.

Hui Ni received the B.Sc. degree in project management from Fuzhou University, China, 2014. He is currently a deputy general manager of Fujian MM Electronics Co., Ltd. His research interests include image encryption and RF circuit design.

Yinghui Tang received the M.S. degree in the College of Engineering, Huaqiao University, Quanzhou, China, in 2021. Her research interests include chaos theory and its applications in information security, chaos-based cryptography, image processing.