A Secure Chaotic Block Image Encryption Algorithm Using Generative Adversarial Networks and DNA Sequence Coding

Pengfei Fang, 1 Han Liu, 1 Chengmao Wu, 2 and Min Liu 3

1 School of Automation and Information Engineering, Xi’an University of Technology, Xi’an 710048, China
2 School of Electronic Engineering, Xi’an University of Posts & Telecommunications, Xi’an 710061, China
3 School of Engineering and Computer Science, Australian National University, Canberra Australian Capital Territory 0200, Australia

Correspondence should be addressed to Han Liu; liuhan@xaut.edu.cn

Received 16 November 2020; Revised 13 December 2020; Accepted 26 December 2020; Published 15 January 2021

Academic Editor: Heng Liu

Copyright © 2021 Pengfei Fang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a block encryption algorithm based on a new chaotic system that combines generative adversarial networks (GANs) and DNA sequence coding. First, the new one-dimensional chaotic system that combines GANs with DNA sequence coding generates two more complex keystream sequences. Then, the two different random sequences are combined with an improved Feistel network by utilizing the product of the block matrix to encrypt the image to scramble and diffuse the image. Finally, the security performance of this algorithm is quantitatively analysed. The simulation results show that the proposed chaotic system has a large key space, and the new algorithm yields adequate security and can resist exhaustive attacks and chosen-plaintext attacks. Therefore, this approach provides a new algorithm for secure transmission and protection of image information.

1. Introduction

With the rapid development of computer technology, when image information is transmitted, it is easy to leak or steal information, which could result in an unpredictable impact; thus, the security of image information transmission is becoming increasingly important. However, since the size of image datasets is much larger than the size of text datasets and the correlation between adjacent pixels is very strong, traditional encryption algorithms, such as the data encryption standard (DES) and advanced encryption standard (AES) [1, 2], with a low encryption security and the risk of being cracked, might not be suitable for image encryption. Therefore, image encryption and decryption technology has become one of the most important issues in the field of information security.

Many image encryption algorithms have been proposed by researchers. The common image encryption algorithms include the following: the pixel scrambling algorithm [3], which scrambles the position of each pixel in the image without changing the grey value of the pixel in the image. The histogram of the image does not change before and after the image changes, so it is easy to be decrypted; the pixel grey value diffusion algorithm [4], in which the grey value of each point in the image is used to change the grey value of each pixel in the original image by a certain transformation and the key can be easily analysed after the thief knows some part of the information in the plaintext; and the scrambling and diffusion combination algorithm [5], which has the advantage of not only changing the position of each pixel in the original image but also changing the grey value of the pixel, and the cryptosystem has a large key space and strong anti-interference ability.

The chaotic system has excellent aperiodicity, pseudorandomness, and highly sensitive initial values and is easy to reproduce, which is widely used in the field of image encryption [6]. Chaotic systems can be divided into low-dimensional and high-dimensional systems. The classical low-dimensional chaotic system includes a logistic chaotic system and a sine chaotic system, which has a simple structure and is easily implemented. High-dimensional chaotic systems include the Lorenz hyperchaotic system and Chen hyperchaotic...
system, which have complex nonlinear dynamic behaviour. The authors of [7] proposed the double-humped logistic map, which is used for pseudorandom number key generation (PRNG). The generalized parameter added to the map provides more control on the map chaotic range. The authors of [8] proposed a method of making a simple and effective chaotic system by using a difference of the output sequences of two identical existing one-dimensional (1D) chaotic maps. The authors of [9] proposed an improved classic quadratic chaotic map to enhance its chaotic properties. Compared with the classical quadratic map, the proposed quadratic map demonstrates a much larger maximal Lyapunov exponent. The authors of [10] proposed the new chaotic system, the tent-logistic system, which has a better chaotic performance and a wider chaotic range than the tent and logistic systems. The authors of [11] proposed a new one-dimensional chaotic map that can be used as a secure pseudorandom number generator in encryption systems. Nevertheless, low-dimensional chaotic systems have the shortcomings of a limited range of chaotic behaviours, a small space for generating chaotic sequences, and poor safety, which limit their application. Additionally, high-dimensional chaotic systems have multiple parameters, complex hardware design, large numbers of calculations, and other defects that are limiting factors. To solve problems such as poor performance of chaotic dynamics, small key space, and poor randomness of chaotic sequences, in this paper, we construct a new low-dimensional system. Analysis shows that the proposed system has a better chaotic performance and wider chaotic range than previously used systems. It can not only increase the randomness of the chaotic sequences but also expand the key space of cryptosystems, which is suitable for chaotic image encryption.

From the structure of image encryption based on a chaotic system, it mainly uses the scrambling and diffusion mechanism. The common chaotic encryption algorithms include those based on bit-level, pixel-level, one-time keys, deoxyribonucleic acid (DNA) rules, dynamic DNA coding, mathematical models, S-boxes scrambling, and diffusion. The authors of [12] proposed transforming an image into a bitmap block to perform a logical XOR operation and obtain a ciphertext image that can resist common attacks. The authors of [13] proposed a chaotic system to construct a complex key, and the pixel values and pixel positions of the block image were changed to implement encryption. Some encryption algorithms based on the combination of the Feistel network and DNA coding have been proposed. The authors of [14] proposed the scrambling transformation of image pixel positions and the diffusion of pixel values with a Hill permutation, Feistel transformation, chaotic scrambling, and dynamic DNA encoding, in which the F function was used for the Feistel transformation. The DNA sequence operation reduced the number of encryption rounds by using multiple scrambling and DNA encoding and decoding steps to achieve the effect of multiround encryption, which resulted in good security performance. The authors of [15] proposed an encryption algorithm based on the Feistel network and DNA encoding scrambling and a diffusion structure for image encryption, which can effectively solve strong plaintext sensitivity. The authors of [16] proposed image encryption based on a double chaotic pseudorandom generator. It uses four secret keys for encryption, which is highly secure. The authors of [17] presented an image encryption algorithm based on dynamic S-boxes and random blocks, in which a plaintext image is scrambled and diffused to implement an encrypted image. The authors of [18] presented a new 5D continuous hyperchaotic system that is combined with DNA dynamic encoding, scrambling, and diffusion to encrypt an image, and the algorithm can resist chosen-plaintext attacks. The authors of [19] presented a novel colour image encryption algorithm based on dynamic DNA encoding and a chaotic system, where DNA encoding and diffusion are used to diffuse the image information. The authors of [20] presented quantum image compression and an encryption algorithm with a Daubechies quantum wavelet transform (DQWT) and hyperchaotic Henon map. The algorithm has high encryption efficiency and good security. The authors of [21] presented an approach based on the quaternion discrete fractional Hartley transform (QD-FrHT) and an improved pixel adaptive diffusion encryption algorithm, which can increase the encryption capacity and reduce the consumption of keys. The authors of [22] presented an approach based on hyperchaotic system and a quantum cross-exchange operation to realize bit-level image encryption. The algorithm has high complexity and good security. The authors of [23] presented an approach based on a chaotic system and a two-dimensional linear canonical transform (2D-LCT) encryption algorithm, and the algorithm has good reliability. However, the above encryption algorithms have limited key spaces and involve simple superposition; thus, these algorithms have difficulty in resisting brute-force attacks and chosen-plaintext attacks and have poor security, which leads to algorithm cracking. To solve the problems of low encryption efficiency, difficulties in resisting chosen-plaintext attacks, and low security, among others, in this paper, a new block image encryption algorithm is proposed that uses the one-time pad mechanism, double key stream mechanism, and improved Feistel structure to scramble and diffuse the pixel value and pixel position of the block image. Consequently, the block image parallel encryption is realized and the security performance and encryption efficiency of the algorithm are improved so that it can resist chosen-plaintext attacks and other encryption attacks.

Nonlinear complex networks and deep learning are new research directions and have been widely used, especially in the field of image processing, for almost two years. A nonlinear complex network has the characteristics of a large number of weight parameters, nonlinearity, and a complicated network structure, which provides ample key space and resistance for brute-force attacks and chosen-plaintext attacks. The authors of [24] presented a two-dimensional chaotic system along with a Boolean network. Matrix semitensor product theory and random position transformation are used to scramble and diffuse an image to realize encryption, providing high encryption efficiency and security. The authors of [25] presented an image block scrambling algorithm followed by the generation of the key stream by combining a
Boolean network with a mixed linear-nonlinear coupled map; the image is encrypted by a matrix semitensor and key stream. The algorithm is secure, effective, and suitable for colour image encryption. The authors of [26] presented a new dispersion-keeping evaluation mechanism and two biobjective memetic genetic programming algorithms, which have good regression characteristics and can be applied to image encryption. The authors of [27] developed a Lorenz chaotic generation key stream matrix for image scrambling, and matrix semitensor product theory is applied to diffuse the image to generate a ciphertext image with improved encryption security. The authors of [28] developed PWLCM to generate a key stream and combined it with the nonlinear characteristics of the McCulloch–Pitts model to encrypt an image, which has a large key space and can resist common attacks. In the above described research, from the perspective of encryption algorithm security, the matrix semitensor product theory, Boolean network theory, and fractal sorting matrix are applied to image encryption, which has high security. In addition to the choice of algorithm, the structure of the key stream determines the performance of the encryption system. Therefore, in this paper, from the point of view of key stream construction and the combination of generative adversarial networks (GANs), a new one-dimensional chaotic system, and DNA sequence coding, a double key stream with good randomness and complexity is constructed. At the same time, the nonlinear characteristics of the GANs and the parallel computing characteristics of DNA coding are used to solve the periodic problem caused by the limited precision of chaotic sequences, and the plaintext image is encrypted with the encryption algorithm. Moreover, it also provides a new idea for the combination of deep learning, GANs, DNA sequence coding, and image information security technology.

To summarize the above, a secure chaotic block image encryption algorithm using generative adversarial networks and DNA sequence coding is proposed. A one-dimensional, two-parameter chaotic system with a large key space and a better chaotic dynamic performance is proposed, in which GANs are combined with DNA sequence coding to generate better randomness and complexity in the fusion random sequence and DNA coding sequence as a double key stream. Then, the matrix product is used to encrypt the plaintext block image pixel-level scrambling and diffusion as a ciphertext image based on an improved Feistel network, one-time pad mechanism, and double key stream. Moreover, the security of the algorithm is also analysed. The experimental results show that the proposed encryption algorithm based on the new chaotic system, GANs, and DNA sequence coding has a large secret key space, and it can resist common attacks with good security.

The remainder of this paper is structured as follows. The new chaotic system and fusion random sequence are detailed in Section 2. The encryption algorithm is detailed in Section 3. In Section 4, the experimental results and a security analysis are presented. Finally, Section 5 presents some conclusions.

2. New Chaotic System and Fusion Random Sequence

2.1. New Chaotic System. In nonlinear dynamic systems, a logistic system is a deterministic system of aperiodic non-convergence, which is similar to a stochastic process and is extremely sensitive to the initial values of parameters. Compared with the pseudorandom sequences produced by the traditional linear system, the logistic system sequences have secure confidentiality [29]. The logistic system can be expressed as follows:

\[ x_{i+1} = a x_i (1 - x_i), \quad (0 < x_i < 1, i = 1, 2, \ldots), \]

where \( a \in [3.569946, 4] \) is the control parameter and \( i \) is the number of iterations. However, the logistic chaotic system has the problems of a small chaotic interval and discontinuous distribution of the chaotic sequence. Ramadan et al. [9] proposed that the improved quadratic chaotic system can be used, as shown in the following equation:

\[ x_{i+1} = (a + (1 - 8x_i)^2) \mod 1, \quad (0 < x_i < 1, i = 1, 2, \ldots), \]

where the control parameter \( a \in [0, \infty) \) except for the values 0.11, 1.11, 2.11, \ldots to infinity; \( \mod(\cdot) \) is the modulus operator, of which the chaotic interval is large and the chaotic sequence is evenly distributed, but the chaotic system control parameters are few. To obtain better chaotic dynamic performance, the chaotic interval and control parameters are used to make it more suitable for chaotic image encryption. A new chaotic system that combines logistical chaotic with improved quadratic chaotic systems is proposed, as shown in the following equation:

\[ x_{i+1} = (a - bx_i (1 - 8x_i)^2) \mod 1, \quad (0 < x_i < 1, i = 1, 2, \ldots), \]

where the control parameter \( a \in [0, \infty) \) for the values 0.11, 1.11, 2.11, \ldots to infinity; \( b \in [0, 4] \) and \( \mod(\cdot) \) is the modulus operator, which has a wider range of values and better chaotic behaviour.

2.1.1. Bifurcation Diagram. Bifurcation diagrams are widely used in chaotic behaviour analysis [30]. The iteration step size is set to 0.5, and the number of iterations is 150. Additionally, the proposed chaotic system parameters are selected as \( a, b \in [0, 2] \), and the bifurcation diagrams of the logistic chaotic system, improved quadratic chaotic system, and new chaotic system with different control parameters of \( a, b \) are shown in Figure 1.

We can see from Figure 1 that the bifurcation diagram of two different control parameters of the new chaotic system is uniformly distributed so that the new chaotic system adapts to the requirements of image cryptography.

2.1.2. Lyapunov Exponent. The Lyapunov exponent is a mathematical tool used to characterize the chaotic behaviour of a nonlinear dynamic system [31]. When the exponent is
positive, the system exhibits chaotic behaviour, and the proposed chaotic system control parameters are set to $a, b \in [0, 4]$. The Lyapunov exponent spectrum of the logistic chaotic system, improved quadratic chaotic system, and new chaotic system is shown in Figure 2. The Lyapunov exponents of the logistics chaotic system, improved quadratic chaotic system, and new chaotic system and exponents from references [7–11] are listed in Table 1.

Figure 2 and Table 1 show that all kinds of chaotic systems listed above have different positive Lyapunov exponents. However, the new chaotic system has the largest Lyapunov exponent, which means that it yields more chaotic behaviour than the other systems.

2.1.3. NIST Test of New Chaotic Sequence. To test the complexity and randomness of chaotic sequences generated by new chaotic systems, the National Institute of Standards and Technology (NIST) is used to test chaotic sequences [32]. The measure of sequence randomness is the $P$ value algorithm, which provides the probability that the randomness of the sequence is better than that of a truly random sequence. All test results were determined by $P$ value; if $P < 0.01$, then the sequence is not random and complex. If $P \geq 0.01$, the sequence is random and complex. According to the requirements of the NIST test software, 10 groups of $10^6$ bits of new system (3) sequences are selected for testing. The test results are shown in Table 2.

Table 2 shows that the new chaotic system sequence has passed all NIST tests; therefore, it has good complexity and randomness characteristics.

2.1.4. Complexity Analysis of the Chaotic System. To better verify the randomness and complexity of the chaotic sequence generated by the new chaotic system compared with the relevant literature, spectral entropy, 0-1 sequencing, and approximate entropy [32] are used for analysis, as shown in Table 3.

Table 3 shows that the value of the new chaotic system sequence proposed is greater than those in other studies,
showing better randomness and complexity. Therefore, it is suitable for image encryption algorithms.

2.2. Fusion Sequence. In recent years, deep learning technology has made breakthrough developments, showing obvious advantages in various fields. Generative adversarial networks (GANs) are based on a generation model framework and were proposed by Goodfellow, who was inspired by the zero-sum game of using two people to train a network by confrontation [33]. GANs are used in image segmentation and image reconstruction and have good nonlinear and data reconstruction effects. At present, the GANs are one of the most important models in the field of computers. The GANs generate the required data samples through the competition between the generation model and the discriminant model. Compared with the traditional model, the combination of GANs and a chaotic system has unique advantages: (1) the training speed of the model is improved and then improves the efficiency of the training model; (2) compared with other models, it can produce clearer and more real samples; (3) more flexibility improves the expansion ability of the model; and (4) the complex nonlinear characteristics, which can effectively eliminate the problem of periodic chaotic sequences and increase the key space, improve the resistance to brute-force attacks of the encryption system and key. Therefore, the fusion pseudo-random sequence with a larger key space and stronger randomness can be generated by combining the generative adversarial networks with a chaotic system.
Additionally, the hidden layer uses the tanh activation function as the real value of the model, and the rectified linear unit (ReLU) activation function is used in the generative model. GANs, both the generator and the discriminator use a fully connected layer, for which the generator is a single-layer neural network, and the discriminator is a three-layer fully connected network. The rectified linear unit (ReLU) activation function is used in the generative model. The chaotic sequence is set as the real value of the model, and the normal distribution noise is set as the input value. Additionally, the hidden layer uses the tanh activation function, and the output does not have an activation function. The generation of fusion random sequences is characterized by high randomness and complexity, which is used to construct the encryption key stream and improves the security of the encryption system. A block diagram of the fusion sequences is shown in Figure 3.

The sequence generation process consists of the following steps:

Step 1: the secure SHA-512 is selected in this paper to convert the plaintext information into 512-bit hash values and then generate initial values and control parameters for the chaotic system; this approach increases the correlation between the plaintext and key, which can increase resistance to plaintext attacks. We input the image (size of \( M \times N \)) and select the generated 512-bit hash value as an array of \( H = [h_1, h_2, \ldots, h_{512}] \), which is divided into 64 blocks. Each block contains a total of 8 bits \( k = [k_1, k_2, \ldots, k_8] \), which are integers in the interval \( [0, 255] \), before being converted into decimal digits. Then, the initial values and control parameters of the new chaotic system are obtained as shown in (10).

\[
ha(i) = (k(j + 3) \oplus (k(j + 1) \oplus k(j + 2))) + k(j + 4) + k(j + 5) + k(j + 6) \mod 256,
\]

\( i = 1, 2, \ldots, 10, \)

\[ j = 6 \times (i - 1), \]

\[ x_1(1) = \frac{(ha(1) + ha(10))}{4}, \]

\[ x_2(1) = \frac{(ha(2) + ha(9))}{6}, \]

\[ x_3(1) = \frac{(ha(3) + ha(8))}{8}, \]

\[ cp(i) = \left( \mod\left(\sum_{i=1}^{10} k(i), 256\right) \right) \cdot \left( \mod(2^{3} \times 10) \right), \quad i = 1, 2, \ldots, 6, \]

\[ a_1 = cp(1), \]

\[ b_1 = cp(2), \]

\[ a_2 = cp(3), \]

\[ b_2 = cp(4), \]

\[ a_3 = cp(5), \]

\[ b_3 = cp(6), \]

where \( ha(i) \) and \( cp(i) \) are variables; \( a_1, b_1, a_2, b_2, a_3, b_3 \) are chaotic control parameters; \( x_1(1), x_2(1), \) and \( x_3(1) \) are the initial chaotic values; \( \oplus \) is the XOR operation; and \( \mod(\cdot) \) is the modulus operator.

Step 2: the proposed chaotic system iteratively generates the three chaotic sequences \( f_1, f_2, \) and \( f_3 \) for three sets of initial values: \( f_1 = (x_1, a_1, b_1), f_2 = (x_2, a_2, b_2), \) and \( f_3 = (x_3, a_3, b_3) \).

Step 3: \( f_1, f_2, \) and \( f_3 \) are sent to the discriminator as real data. A random noise signal in the range \([0, 1]\) has a normal distribution as input to the generator. The generator and discriminator are trained to generate random sequences \( y_1, y_2, \) and \( y_3 \).

Step 4: \( y_1, y_2, \) and \( y_3 \) are applied to generate fusion random sequences fusion \((i)\) and \( i = 1, 2, \ldots, M \times N \). The process is shown as follows:

\[ s_1(i) = \mod(y_1(i)y_2(i) - y_3(i), 256), \]

\[ s_2(i) = \mod(y_1(i) + s_1^2(i), 256), \]

\[ s_3(i) = \mod(s_2^2(i) - s_1(i), 256), \]

Table 2: NIST tests.

| Test                              | Sequence |
|----------------------------------|----------|
| Frequency                        | 0.350485 |
| Frequency test within a block    | 0.350485 |
| Cumulative sums                  | 0.739918 |
| Runs                             | 0.350485 |
| Test for the longest run of ones in a block | 0.739918 |
| Binary matrix rank test          | 0.544146 |
| Discrete Fourier transform (spectral) | 0.021141 |
| Nonoverlapping template matching | 0.534146 |
| Overlapping template matching    | 0.350485 |
| Maurer’s “universal statistical” test | 0.534146 |
| Approximate entropy              | 0.213309 |
| Random excursions                | 0.794590 |
| Random excursions variant        | 0.802851 |
| Linear complexity                | 0.534100 |
| Serial                           | 0.213309 |

Table 3: Performance comparison.

| Chaotic system | Spectral entropy | 0-1 test | Approximate entropy |
|----------------|------------------|----------|---------------------|
| Reference [7]  | 0.9258           | 0.9983   | 0.6563              |
| Reference [8]  | 0.9304           | 0.9977   | 0.8021              |
| Reference [9]  | 0.9254           | 0.9965   | 1.3575              |
| Reference [10] | 0.9234           | 0.9986   | 0.6355              |
| Reference [11] | 0.9264           | 0.9981   | 0.6668              |
| Logistic       | 0.9317           | 0.9980   | 0.6514              |
| Proposed       | 0.9970           | 0.9991   | 1.5625              |
2.2.1. Test of Fusion Sequence. To better test the randomness and complexity of the fusion random sequence, we test the proposed fusion random sequence and the proposed chaotic sequences generated with different initial values and control parameters of $x_1, a_1$, and $b_1$ (represented by 1); $x_2, a_2$, and $b_2$ (represented by 2); and $x_3, a_3$, and $b_3$ (represented by 3) and the fusion random sequence (represented by “fusion”). The test results are compared with the approximate entropy values of the random sequences in Table 4.

Table 4 shows that the proposed fusion sequences have the largest value of the spectral entropy, 0-1 test, and approximate entropy. Therefore, the proposed approach can significantly improve the complexity of random sequences and is beneficial to applications such as encryption.

3. Encryption Algorithms

In this section, we introduce a new encryption structure. Many block encryption structures are essentially based on the Feistel network structure, but in the current Feistel network, the encryption and decryption function structures are not sufficiently complicated, and the security is imperfect. These algorithms use only a single key, have low functional complexity, struggle to resist chosen-plaintext attacks, and provide weak security, so they are not applied to image encryption. To solve these problems, an image encryption and decryption algorithm based on improved Feistel network image pixel-level scrambling and a diffusion encryption algorithm are proposed in this paper. In the new chaotic system, GANs and DNA sequence coding are used to construct the double key stream that uses the nonlinear characteristics of the generative adversarial networks and the parallel computing characteristics of DNA coding to solve the periodic problem caused by the limited precision of chaotic sequences. At the same time, the double key stream makes the encryption algorithm more resistant to attacks, which is combined with the improved Feistel network scrambling and diffusion mechanism encryption of the image to achieve local and overall encryption and obtain better encryption security. The process employed by the
image encryption and decryption algorithms is depicted in Figure 4.

The specific steps of the encryption algorithm are as follows:

- Input: the inputs are the plaintext image $P$ and the initial values of the parameters.
- Output: the output is the ciphertext image $C$.
- Step 1: the fusion pseudorandom sequence is generated as key stream $K_1(i)$, and $i = 1, 2, \ldots, M \times (N/2)$, according to Section 2.2.
- Step 2: generate key streams $K_{2(j)}(i)$, $i = 1, 2, \ldots, M \times N$, and $j = 1, 2, \ldots, 16$ encoded

\[
dnakey = \text{dnacoding convert decimal (GenBank database)},
\]

\[
f_1(i) = f_1(i) \times 10^4 \mod 256, \quad i = 1, 2, \ldots, M \times \left(\frac{N}{2}\right),
\]

\[
K_{2(1)}(i) = f_1(i), \quad i = 1, 2, \ldots, M \times \left(\frac{N}{2}\right),
\]

\[
K_{2(2)}(i) = K_{2(1)}(i) \oplus \text{dnakey}, \quad i = 1, 2, \ldots, M \times \left(\frac{N}{2}\right),
\]

\[
K_{2(j)}(i) = \text{dnacoding convert decimal (dnaadd(dnasub(K_{2(j-1)}(i), K_{2(j-2)}(i), \ldots, K_{2(j)}(i)))}, \quad i = 1, 2, \ldots, M
\]

\[
\times \left(\frac{N}{2}\right), 3 \leq j \leq 16,
\]

where \text{dnacoding convert decimal}(-) converts a DNA code to a decimal number, $\oplus$ is the XOR operation, \text{dnaadd}(-) is the nucleotide addition operation, and \text{dnasub}(-) is the nucleotide subtraction operation.

- Step 3: the plaintext matrix $P$ has a size of $M \times N$. This matrix is divided into two matrices: the left subblock $L_i$ and right subblock $R_j$, where $i = 1, 2, \ldots, M \times (N/2)$.
- Step 4: we obtain the iterative pseudorandom sequences generated as key matrices $K_1(i)$ and $K_{2(j)}(i)$, where $i = 1, 2, \ldots, M \times (N/2)$ and $j = 1, 2, \ldots, 16$. To ensure the security and universality of the proposed algorithm, the unitary matrix $U_j$, $i = 1, 2, \ldots, M \times (N/2)$, is added for left subblock $L_i$ iterative encryption; the values of this subblock are all 1, and 16 rounds of iteration were performed.

The block diagram of the encryption algorithm for the improved Feistel network is shown in Figure 5.

### Table 4: Randomness and complexity of random sequences.

| Random sequence | Spectral entropy | 0-1 test | Approximate entropy |
|-----------------|------------------|---------|--------------------|
| 1               | 0.9987           | 0.9982  | 0.8056             |
| 2               | 0.9981           | 0.9968  | 1.6197             |
| 3               | 0.9496           | 0.9981  | 1.6135             |
| Fusion          | 0.9991           | 0.9989  | 1.7712             |

The one-round improved Feistel network encryption process is shown in Algorithm 1.

\[
C(1) = cs(i),
\]

\[
C(i) = \text{bit xor}(cs(i), C(i-1)),
\]

where $C$ is the final ciphertext image and bit xor (-) is XOR operation.

The decryption algorithm involves the inverse of the above process; notably, the inputs are the ciphertext image $C$, and $cs$ is obtained by inverse diffusion of ciphertext image $C$. The output is the plaintext image $P$. The $K_1(i)$ and $K_{2(j)}(i)$ are still combined to form the key stream with the improved Feistel network decryption algorithm to iteratively decrypt the ciphertext image for 16 rounds and obtain a plaintext image $P$. The block diagram of the decryption algorithm for the improved Feistel network is shown in Figure 6.
The one-round improved Feistel network decryption process is shown in Algorithm 2.

4. Experimental Results and Security Analysis

To verify the validity of the above encryption algorithm, a simulation experiment was conducted on an Intel i7-8565U.
Table 6: Nucleotide computing rules associated with the first coding rule.

| + | A | G | C | T | − | A | G | C | T |
|---|---|---|---|---|---|---|---|---|---|
| A | A | G | C | T | A | A | T | C | G |
| G | G | T | A | G | G | A | T | C |
| C | C | T | A | G | C | C | G | A | T |
| T | T | A | G | C | T | T | C | G | A |

(1) Read \( P, K_i(i), K_{2j}(i) \), \( i = 1, 2, \ldots, M \times (N/2), j = 1, 2, \ldots, 16 \)
(2) \( L_i \) = left half of \( P \)
(3) \( R_i \) = right half of \( P \)
(4) \( F(R_i, K_i(i), K_{2j}(i)) = \text{mod}(K_1 \times R_i \times K_{2j}(i), 256) \)
(5) \( R_{i+1} = L_i \oplus F(R_i, K_i, K_2) \)
(6) \( F(R_i, R_{i+1}, U_i) = \text{mod}(R_i + R_{i+1}^2 + U_i, 256) \)
(7) \( L_{i+1} = F(R_i, R_{i+1}, U_i) \)
(8) \( cs = L_{i+1} || R_{i+1} \)

Step 5: let \( C \) with size \( M \times N \) represent the ciphertext image corresponding to plaintext image \( P \).
Step 6: we transform the pixel sequence \( cs(i), i = 1, 2, \ldots, M \times N \) of \( cs \) image and carry out the diffusion operation to obtain the final ciphertext image \( C \).

**Algorithm 1**: Image encryption process.

```
Algorithm 1: Image encryption process.

1. Read \( P, K_i(i), K_{2j}(i) \), \( i = 1, 2, \ldots, M \times (N/2), j = 1, 2, \ldots, 16 \)
2. \( L_1 \) = left half of \( P \)
3. \( R_1 \) = right half of \( P \)
4. \( F(L_1, R_1, U_1) = \text{mod}(L_1 \times R_1 \times U_1, 256) \)
5. \( R_1 = F(L_1, R_1, U_1) \)
6. \( F(R_1, R_{2j+1}, U_i) = \text{mod}(R_1 + R_{2j+1}^2 + U_i, 256) \)
7. \( L_{i+1} = F(R_1, R_{2j+1}, U_i) \)
8. \( P = L_{i+1} || R_{i+1} \)
```

**Algorithm 2**: Image decryption process.

```
Algorithm 2: Image decryption process.

1. Read \( cs, K_1(i), K_{2j}(i) \), \( i = 1, 2, \ldots, M \times (N/2), j = 1, 2, \ldots, 16 \)
2. \( L_1 \) = left half of \( C \)
3. \( R_1 \) = right half of \( C \)
4. \( F(L_{i+1}, R_{i+1}, U_i) = \text{mod}(L_{i+1} \times R_{i+1}^2 \times U_i, 256) \)
5. \( R_{i+1} = F(L_{i+1}, R_{i+1}, U_i) \)
6. \( F(R_{i+1}, K_1(i), K_2) = \text{mod}(K_1 \times R_{i+1} \times K_{2j}(i), 256) \)
7. \( L_{i+1} = R_{i+1} \oplus F(R_{i+1}, K_1(i), K_{2j}(i)) \)
8. \( P = L_{i+1} || R_{i+1} \)
```
CPU with 4 GB of memory and the Windows 10 Professional operating system; the software packages MATLAB 2018b and Python 3.5 were used for coding. Here, we input different plaintext images into the SHA-512 function and use the series of calculations introduced in Section 2.2 to obtain three sets of chaotic initial values and control parameters for the proposed chaotic system: $x_1 (1), x_2 (1), x_3 (1); a_1, b_1, a_2, b_2, a_3, b_3$. For example, if the plaintext image is Lena, then the values are $x_1 (1) = 0.7881, x_2 (1) = 0.7936, x_3 (1) = 0.5640, a_1 = 4.2188, b_1 = 2.0117, a_2 = 3.5547, b_2 = 3.3594, a_3 = 4.3555,$ and $b_3 = 4.7656$. The encryption and decryption results for the Lena $(256 \times 256)$, Cameraman $(256 \times 256)$, All zeroes $(256 \times 256)$, White $(256 \times 256)$, Skeleton $(256 \times 256)$, Remote sensing $(256 \times 256)$, and Scenery $(256 \times 256)$ are shown in Figure 7. In addition, we also give the encryption and decryption results for the colour images Couple $(256 \times 256)$ and Ship $(256 \times 256)$ in Figures 8 and 9.

As seen from Figures 7–9, the encrypted image is random noise, and no effective information can be obtained, which shows the effectiveness of the proposed algorithm.

### 4.1. Histogram Analysis

A histogram shows the distribution of pixel values in an image. The ideal histogram of a ciphertext image is uniformly distributed [6]. The histogram values of the plaintext and ciphertext Lena images are shown in Figure 10.

Figure 10 shows that the histogram distribution of the plaintext image is nonuniform, and the histogram distribution of the encrypted ciphertext image is uniform, so the proposed encryption algorithm is resistant to statistical attacks.

In addition, we use the variance of the histogram to quantitatively evaluate the uniformity of the ciphertext image. The smaller the variance is, the more uniform the histogram distribution is, which is defined as follows:

$$\text{var}(x) = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{(x_i - x_j)^2}{2},$$

where $X = \{x_0, x_1, \ldots, x_{255}\}$ is the set of frequency values, $x_i$ and $x_j$ are the number of pixels whose pixel values are equal to $i$ and $j$, respectively, and $n$ is the grey level of the image. Table 7 shows the histogram variances of the plaintext and ciphertext images.

As seen from Table 7, the variance in the plaintext image histogram is very large, while the variance in the ciphertext image histogram is smaller than that of the plaintext image histogram. However, the histogram variance in the proposed algorithm is smaller than that of other algorithms, which is safe and effective.

To quantitatively test the histogram variance of ciphertext image and analyse the uniformity of encrypted image, we calculate the histogram variance in encrypted images by changing only one key value, as shown in Tables 8 and 9.

As seen from Tables 8 and 9, the histogram variance of the ciphertext image is sensitive to the key. Therefore, it further shows that the proposed algorithm can resist statistical attacks.

### 4.2. Correlation between Adjacent Pixels

In general, the correlation coefficient of adjacent pixels in a plaintext image is close to 1, making the image vulnerable to statistical attacks; therefore, the adjacent pixels in a ciphertext image should have correlations close to 0, which means that the encryption algorithm must resist statistical attacks and provide good security [36]. The formulas of the correlation coefficients can be defined as follows:

$$r_{xy} = \frac{E((x - E(x))(y - E(y)))}{\sqrt{D(x)D(y)}},$$

$$E(x) = \frac{1}{N} \sum_{i=1}^{N} x_i,$$

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

where $E(x)$ and $D(x)$ are the expectation and variance in variable $x$, respectively. First, 10,000 pairs of adjacent pixels in the plaintext image Lena are selected randomly, and the corresponding ciphertext image is obtained based on operations in three directions: horizontal, vertical, and diagonal. The correlation between the plaintext and ciphertext images in each direction is shown in Figure 11.

The correlation values of the plaintext image and the correlation values of the ciphertext images of the proposed algorithm and those in references [12, 14–19] are shown in Table 10. Table 10 shows that the correlation between adjacent pixels in the plaintext image is higher than the values reported in references [12, 14–19]. The correlation values of ciphertext images in the proposed algorithm are smaller than the correlation values of the ciphertext images in references [12, 14–19]. Therefore, the proposed algorithm can effectively resist statistical attacks.

It can be seen from Table 10 that the results of the correlation of two adjacent pixels of the ciphertext image are significant, while those of the proposed algorithm are lower than the existing ones, so the encryption effect is rather good.

### 4.3. Grey Level Co-Occurrence Matrix (GLCM) Analysis

To further evaluate the spatial distribution characteristics of image encryption pixels, the GLCM matrix of adjacent pixels can be used [37]. The characteristic value calculation of the GLCM matrix is as follows.

#### 4.3.1. Contrast

The contrast reflects the clarity and depth of the texture groove of the image. The deeper the texture groove is, the greater the contrast and the clearer the effect, which is defined as follows:
contrast = \sum_{i}^{M} \sum_{j}^{N} (i - j)^2 p(i, j), \quad (22)

where \( p(i, j) \) represents the pixel value of the GLCM matrix and \( M \) and \( N \) are the row and column of the pixel value, respectively.

4.3.2. Energy. Energy reflects the uniformity of the image grey distribution and texture thickness. Large energy values indicate a more uniform and regular texture pattern of the image, which is defined as follows:

\[
\text{energy} = \sum_{i}^{M} \sum_{j}^{N} p(i, j)^2, \quad (23)
\]
where $P(i, j)$ represents the pixel value of the GLCM matrix and $M$ and $N$ are the row and column of the pixel value, respectively.

### 4.3.3. Correlation

Correlation is used to measure the similarity of the GLCM of the image in the direction of row or column, so the value reflects the local grey correlation.
and the greater the value is, the greater the correlation is. It is defined as follows:

$$\text{correlation} = \frac{\sum_{i}^{M} \sum_{j}^{N} (ij) \cdot P(i, j) - \mu_x \cdot \mu_y}{\sigma_x \cdot \sigma_y},$$

where \( P(i, j) \) denotes the coordinate values in the GLCM matrix and \( M \) and \( N \) are the row and column of the pixel value, respectively.

4.3.4. Homogeneity. Homogeneity reflects how closely the elements in the GLCM are distributed along the GLCM diagonal, and smaller values of homogeneity reflect higher security of a ciphertext image. Homogeneity is defined as follows:

$$\text{homogeneity} = \frac{\sum_{i}^{M} \sum_{j}^{N} P(i, j)}{\sum_{i}^{M} \sum_{j}^{N} (1 + i - j)}.$$

where \( P(i, j) \) denotes the coordinate values in the GLCM matrix and \( M \) and \( N \) are the row and column of the pixel value, respectively.

The characteristic values of the GLCM matrix adjacent pixels for the Lena ciphertext image, plaintext image, and related literature encryption algorithm are shown in Table 11.

The two-dimensional histograms of the GLCM matrix of adjacent pixels in four different directions for the Lena plaintext image and ciphertext image obtained by the proposed algorithm are shown in Figures 12 and 13.

It can be seen from Table 11 and Figures 12 and 13 that the characteristic value of the GLCM matrix for the ciphertext image and the two-dimensional histogram of the ciphertext image indicates that the spatial distribution of adjacent pixels in the encrypted image is uniform; therefore, the adjacent pixels have sufficient random characteristics.

4.4. Entropy Analysis. Entropy is suitable for quantitative evaluations of the encryption effects of image encryption algorithms [38], which can measure the distribution of image grey values. The more uniform the grey distribution is, the greater the image information entropy. Additionally, little information in a plain image can be obtained from the grey distribution of a ciphertext image by an attacker, so the encryption algorithm has high security. Entropy is mainly divided into global entropy and local entropy. Global entropy is defined as follows:

$$M = -\sum_{i=1}^{L} P_i \log P_i,$$

where \( P_i \) is the probability of the image pixel value and \( L \) is the image grey level. The ciphertext image global entropy is close to 8, suggesting that it is unlikely that the algorithm will divulge information. The global entropy values for the plaintext and ciphertext images obtained by the proposed algorithm and in references [12–19] are listed in Table 12.

Table 12 shows that the global entropy of the algorithm proposed in this paper is close to 8, which indicates that it can resist statistical attacks.

On the basis of global entropy, local entropy, which overcomes the shortcomings of global entropy such as inaccuracy, inconsistency, and low efficiency, is proposed. Local entropy is an improvement of global entropy. First, nonoverlapping blocks are selected randomly, and then, the average global entropy of small image blocks is calculated. According to the algorithm described in the literature, 40 nonoverlapping blocks with a size of 32 × 32 are tested. The
Table 8: Variance in histograms compared among all keys in the proposed algorithm.

| Ciphertext    | $x_1$  | $a_1$  | $b_1$  | $x_2$  | $a_2$  | $b_2$  | $x_3$  | $a_3$  | $b_3$  |
|---------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| Scene         | 261.5430 | 258.0517 | 262.3394 | 261.3313 | 262.9711 | 253.2538 | 261.9252 | 253.8197 | 261.4685 |
| Skeleton      | 302.9506 | 304.5759 | 297.7399 | 304.5759 | 304.6851 | 305.7361 | 304.7073 | 305.4255 | 305.7293 |
| Remote sensing | 254.0500 | 253.8370 | 252.2495 | 255.0578 | 253.4440 | 254.9473 | 253.0216 | 253.1120 | 251.2266 |
| Lena          | 237.7072 | 236.2798 | 236.8458 | 232.2816 | 237.9360 | 233.0170 | 236.6012 | 235.5416 | 235.2537 |
| Average       | 264.0627 | 263.1861 | 262.2937 | 263.3116 | 264.7590 | 261.7386 | 264.0638 | 261.7247 | 263.4195 |

Table 9: Percentage of variances difference of histograms compared among all keys in the proposed algorithm.

| Ciphertext    | $x_1$ (%)  | $a_1$ (%)  | $b_1$ (%)  | $x_2$ (%)  | $a_2$ (%)  | $b_2$ (%)  | $x_3$ (%)  | $a_3$ (%)  | $b_3$ (%)  |
|---------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|
| Scene         | 9.9045     | 8.8091     | 1.1076     | 9.9247     | 9.9327     | 9.6758     | 9.9190     | 9.6979     | 9.9254     |
| Skeleton      | 11.4726    | 11.5720    | 12.5700    | 11.5671    | 11.5002    | 11.6809    | 11.5391    | 11.6315    | 11.6061    |
| Remote sensing | 9.6208    | 8.0924     | 9.0297     | 7.6865     | 9.5763     | 9.7405     | 9.5818     | 9.6709     | 8.5317     |
| Lena          | 8.0019     | 9.6477     | 9.6217     | 8.8215     | 8.9868     | 8.9026     | 8.9600     | 8.9976     | 8.9307     |
| Average       | 9.7499     | 9.5303     | 8.0823     | 9.4999     | 9.9990     | 9.5634     | 9.9870     | 9.9995     | 9.7485     |

Figure 11: Correlation between adjacent pixels. (a) Horizontal correlation for plain image Lena. (b) Vertical correlation for ciphertext image Lena. (c) Diagonal correlation for plain image Lena. (d) Horizontal correlation for ciphertext image Lena. (e) Vertical correlation for plain image Lena. (f) Diagonal correlation for ciphertext image Lena.
Table 10: Correlation between the adjacent pixels of plaintext and ciphertext images.

| Algorithm                  | Horizontal | Vertical | Diagonal |
|----------------------------|------------|----------|----------|
| Proposed (Scenery)         | −0.0126    | −0.0018  | −0.0051  |
| Proposed (Skeleton)        | −0.0096    | 0.0001   | −0.0156  |
| Proposed (Remote sensing)  | 0.0021     | −0.0157  | −0.0263  |
| Proposed (Lena)            | −0.0049    | −0.0065  | 0.0057   |
| Reference [16] (Lena)      | −0.0147    | 0.0045   | 0.0088   |
| Reference [14] (Lena)      | −0.0304    | −0.0056  | −0.0058  |
| Reference [12] (Lena)      | 0.0001     | 0.0084   | −0.0034  |
| Reference [17] (Lena)      | −0.0001    | −0.0075  | 0.0060   |
| Reference [15] (Lena)      | 0.0097     | 0.0051   | 0.0104   |
| Reference [18] (Lena)      | −0.0802    | 0.0160   | −0.0120  |
| Reference [19] (Lena)      | 0.0067     | −0.0036  | −0.0038  |

Table 11: Characteristics of GLCM.

| Algorithm                  | Contrast  | Correlation | Energy   | Homogeneity |
|----------------------------|-----------|-------------|----------|-------------|
| Proposed (Scenery)         | 10.5877   | −0.0032     | 0.0156   | 0.3900      |
| Proposed (Skeleton)        | 10.5349   | −0.0075     | 0.0156   | 0.3878      |
| Proposed (Remote sensing)  | 10.5670   | −0.0040     | 0.0156   | 0.3886      |
| Proposed (Lena)            | 10.6041   | −0.0003     | 0.0156   | 0.3872      |
| Reference [16] (Lena)      | 10.5144   | 0.0021      | 0.0156   | 0.3895      |
| Reference [14] (Lena)      | 10.4742   | 0.0050      | 0.0156   | 0.3878      |
| Reference [12] (Lena)      | 10.4233   | 0.0075      | 0.0156   | 0.3911      |
| Reference [13] (Lena)      | 10.5589   | −0.0079     | 0.0156   | 0.3879      |
| Reference [17] (Lena)      | 10.6372   | −0.0074     | 0.0156   | 0.3877      |
| Reference [18] (Lena)      | 10.3418   | −0.0186     | 0.0157   | 0.3985      |
| Reference [19] (Lena)      | 10.5101   | 0.0025      | 0.0156   | 0.3898      |

Figure 12: Two-dimensional histogram of the plaintext image. (a) 0°. (b) 45°. (c) 90°. (d) 135°.
corresponding theoretical value of local entropy is 7.8087, which is defined as follows:

$$\text{local entropy} = \frac{1}{n} \sum_{j=1}^{n} H(B_j),$$  \hspace{1cm} (27)

where $B_j$ is a nonoverlapping image block. Table 13 lists the local entropy test results of the proposed encryption algorithm and related literature.

Table 13 shows that the value of the proposed encryption algorithm in this paper is the closest to the theoretical value, so the ciphertext image obtained by the proposed encryption algorithm has good security.

### 4.5. Key Space Analysis

In the key space $K = \{x_1, x_2, x_3, a_1, a_2, a_3, b_1, b_2, b_3\}$ of the proposed algorithm, the range of $x_1, x_2$, and $x_3$ is $[0, 1]$, and the range of $a_1, a_2, a_3 \in [0, \infty]$, $b_1, b_2, b_3 \in [0, 4]$. If a dual-precision representation is used up to 14 digits after the decimal point, the total key space can reach $10^{42}$, that is, the key length is approximately $\log_2(10^{42}) \approx 140$ bits. Generally, an algorithm key length of 128 bits is considered safe [39], so the key space of the proposed algorithm is large enough to effectively resist exhaustive attacks.

### 4.6. Differential Analysis (Plaintext Sensitivity)

A plaintext image and one with a small change are applied for encryption and to determine how a small change in the original image affects the encrypted image. This kind of attack is called a differential attack. To ensure the security of an image encryption scheme against a differential attack, two quantitative measures are used, including the number of pixels change rate (NPCR) and unified averaged changed intensity (UACI). Values of $\text{NPCR} > 33.4\%$ and $\text{UACI} > 99.6\%$ ensure that an image encryption algorithm is immune to differential attacks [40].
Images with only one pixel value difference are tested with the encryption algorithm. The number of changed image pixels is a percentage of the total number of pixels, which is represented by the NPCR. For two images with only one pixel value difference, after the same encryption algorithm is run, the degree of image pixel change is represented by the UACI. Two images with only one pixel value difference are encrypted as $C_1$ and $C_2$, where $H(i, j)$ is the image pixel difference value and $M, N$ is the image size.

$$H(i, j) = \begin{cases} 
0, & C_1(i, j) = C_2(i, j), \\
1, & C_1(i, j) \neq C_2(i, j), 
\end{cases}$$

$$\text{NPCR} = \frac{\sum_{i,j} H(i, j)}{MN} \times 100\%,$$

$$\text{UACI} = \frac{\sum_{i,j} \left( \frac{1}{255} \| C_1(i, j) - C_2(i, j) \| \right)}{MN} \times 100\%.$$

In the Lena plaintext image, the pixel value of the coordinates (120, 120) changes from “156” to “155”. A comparative analysis of the NPCR and UACI values of the proposed algorithm and those from references [12, 14, 16, 18, 19] is shown in Table 14.

It can be seen from Table 14 that the proposed encryption algorithm is very sensitive to tiny changes in the plain image; even if there is only a one-bit difference between two plain images, the decrypted images will be completely different.

To quantitatively analyse the differential attack of the proposed algorithm, in the Lena plaintext image, random selection of the pixel value of the coordinates (120, 120), (140, 140), and (150, 150) and a change in a one-bit pixel value are performed on the plaintext images to obtain im1, im2, and im3. The ciphertext image is encrypted by the proposed encryption algorithm. The NPCR and UACI values of im1, im2, and im3 for five encryption rounds are compared with reference [19] and are listed in Tables 15 and 16.

As seen from Tables 15 and 16, after the first two rounds of encryption, the values of NPCR and UACI are close to the theoretical values, so the proposed algorithm has good security.

4.7. Key Sensitivity Analysis. The security of a cryptosystem is highly related to key sensitivity, and a chaotic system is highly sensitive to initial conditions. That is, a slight change in the key will lead to the failed decryption of the encrypted image [37]. For example, the encryption key obtained from the Lena plaintext image is shown in Tables 17 and 18. The key stream is generated by the encryption key. On decryption, Key 1, Key 2, Key 3, Key 4, Key 5, Key 6, Key 7, Key 8, and Key 9 in Tables 13 and 14 are obtained by slightly changing the initial value of the Lena plaintext image, which is carried out to generate fusion pseudorandom sequences as the key stream to decrypt the ciphertext image. The result is shown in Figure 14.

Figure 14 shows that the decryption key is slightly transformed, and the decrypted image is completely different from the plaintext image, which indicates that the algorithm is very sensitive to the key.

To quantitatively analyse the key sensitivity, we use the initial key, Key 0, in Tables 17 and 18 and slightly changed Key 1, Key 2, Key 3, Key 4, Key 5, Key 6, Key 7, Key 8, and Key 9 to encrypt the Lena plaintext image. The ciphertext image and difference image are shown in Figure 15, and the difference value is shown in Table 19.

As seen from Figure 15 and Table 19, when the key is slightly changed, the difference value of the ciphertext image obtained is very large. Therefore, the proposed encryption algorithm in this paper is highly sensitive to the key.

4.8. Classical Types of Attacks. In cryptography, the attack algorithms for passwords are commonly plaintext attacks, chosen-ciphertext attacks, known plaintext attacks, and known ciphertext attacks [41, 42].

The security of the encryption algorithm should depend only on the security of the key. The encryption algorithm itself does not need to be kept secretly. The only secret that needs to be kept secretly is the key shared by both sides of communication. Cryptanalysis is based on this. According to the amount of information the attacker can obtain, there are four types of attacks on a cryptographic system:

1. Ciphertext-only attack: the attacker observes only the ciphertext and tries to recover the corresponding plaintext.
(2) Known plaintext attack: the attacker knows one or more plaintext ciphertext pairs encrypted with the same key and attempts to recover the plaintext corresponding to other ciphertexts.

(3) Chosen-plaintext attack: the attacker can select plaintext and obtain the corresponding ciphertext, which is similar to obtaining the use right of the encryptor. Its goal is to determine the plaintext corresponding to other ciphertexts.

(4) Chosen-ciphertext attack: the attacker can select ciphertext and obtain the corresponding plaintext, and its purpose is still to recover the plaintext corresponding to other ciphertexts.

A chosen-plaintext attack is the most threatening kind of cryptosystem attack. A secure encryption system must be able to resist a chosen-plaintext attack. In a chosen-plaintext attack, the attacker can arbitrarily select a certain number of plaintexts in advance, which allows the attacked encryption algorithm to obtain the ciphertext. The goal of the attacker is to obtain the key information through this process so that the attacker can more effectively crack the encryption algorithm and related keys. In the worst case, the key for decryption is obtained by the attacker.

This paper mainly analyses the security of the improved Feistel network for a chosen-plaintext attack. We choose an all-zero matrix as the plaintext and obtain the ciphertext matrix based on encryption algorithm Step 4, as shown below:

| Table 15: NPCR values of ciphertext image (%) |
|-----------------------------------------------|
| Round | Proposed (im1) | Proposed (im2) | Proposed (im3) | Reference [19] |
|-------|----------------|----------------|----------------|----------------|
| 1     | 99.39          | 99.28          | 99.37          | 99.24          |
| 2     | 99.41          | 99.35          | 99.37          | 99.39          |
| 3     | 99.51          | 99.47          | 99.41          | 99.45          |
| 4     | 99.58          | 99.51          | 99.52          | 99.56          |
| 5     | 99.61          | 99.59          | 99.60          | 99.61          |

| Table 16: UACI values of ciphertext image (%) |
|-----------------------------------------------|
| Round | Proposed (im1) | Proposed (im2) | Proposed (im3) | Reference [19] |
|-------|----------------|----------------|----------------|----------------|
| 1     | 19.32          | 22.35          | 20.56          | 21.87          |
| 2     | 19.51          | 10.57          | 25.46          | 30.56          |
| 3     | 33.31          | 33.41          | 33.39          | 33.44          |
| 4     | 33.38          | 33.41          | 33.40          | 33.48          |
| 5     | 33.38          | 33.40          | 33.40          | 33.49          |

| Table 17: Encryption keys. |
|-----------------------------|
| Keys | $x_1$ | $x_2$ | $x_3$ | $a_1$ |
|------|-------|-------|-------|-------|
| Key 0 | 0.7881 | 0.7936 | 0.5640 | 4.2188 |
| Key 1 | 0.7881 + 10^{-15} | 0.7936 | 0.5640 | 4.2188 |
| Key 2 | 0.7881 | 0.7936 + 10^{-15} | 0.5640 | 4.2188 |
| Key 3 | 0.7881 | 0.7936 | 0.5640 + 10^{-15} | 4.2188 + 10^{-15} |
| Key 4 | 0.7881 | 0.7936 | 0.5640 | 4.2188 |
| Key 5 | 0.7881 | 0.7936 | 0.5640 | 4.2188 |
| Key 6 | 0.7881 | 0.7936 | 0.5640 | 4.2188 |
| Key 7 | 0.7881 | 0.7936 | 0.5640 | 4.2188 |
| Key 8 | 0.7881 | 0.7936 | 0.5640 | 4.2188 |
| Key 9 | 0.7881 | 0.7936 | 0.5640 | 4.2188 |

| Table 18: Encryption keys. |
|-----------------------------|
| Keys | $b_1$ | $a_2$ | $b_2$ | $a_3$ | $b_3$ |
|------|-------|-------|-------|-------|-------|
| Key 0 | 2.0117 | 3.5547 | 3.3594 | 4.3555 | 4.7656 |
| Key 1 | 2.0117 | 3.5547 | 3.3594 | 4.3555 | 4.7656 |
| Key 2 | 2.0117 | 3.5547 | 3.3594 | 4.3555 | 4.7656 |
| Key 3 | 2.0117 | 3.5547 | 3.3594 | 4.3555 | 4.7656 |
| Key 4 | 2.0117 + 10^{-15} | 3.5547 | 3.3594 | 4.3555 | 4.7656 |
| Key 5 | 2.0117 + 10^{-15} | 3.5547 | 3.3594 | 4.3555 | 4.7656 |
| Key 6 | 2.0117 | 3.5547 + 10^{-15} | 3.3594 | 4.3555 | 4.7656 |
| Key 7 | 2.0117 | 3.5547 | 3.3594 + 10^{-15} | 4.3555 | 4.7656 |
| Key 8 | 2.0117 | 3.5547 | 3.3594 | 4.3555 + 10^{-15} | 4.7656 |
| Key 9 | 2.0117 | 3.5547 | 3.3594 | 4.3555 | 4.7656 + 10^{-15} |
Figure 14: Sensitivity to the ciphertext image test results for the Lena image. (a) Decryption image by Key 1. (b) Decryption image by Key 2. (c) Decryption image by Key 3. (d) Decryption image by Key 4. (e) Decryption image by Key 5. (f) Decryption image by Key 6. (g) Decryption image by Key 7. (h) Decryption image by Key 8. (i) Decryption image by Key 9.

Figure 15: Continued.
Figure 15: Continued.
\[ \mathbf{R}_i = [\{0\}] \oplus \left( \text{mod} \left( K_1(i) \times [\{0\}] \times K_2(i), 256 \right) \right), \quad (29) \]

\[ L_i = \text{mod} \left( [\{0\}] + R_i^2 + U_i, 256 \right), \quad (30) \]

where \([\{0\}]\) is the all-zero subblock plaintext matrix. From (29) and (30), if \([\{0\}]\) subblocks are chosen to form the plaintext matrix for encryption, the ciphertext matrix can still be obtained, but the keys \(K_1(i)\) and \(K_2(i)\) and the \([\{0\}]\) subblock \(\rightarrow \text{mod} \left( K_1(i) \times [\{0\}] \times K_2(i), 256 \right)\) operation are utilized in the matrix product. \(K_1(i)\) and \(K_2(i)\) are transformed and added to the all-zero matrices so that the attacker cannot obtain the encryption algorithm and key-related information through the relationship between the plaintext matrix and the ciphertext matrix. Moreover, \([\{255\}]\) is chosen with all 255 subblocks as the plaintext matrix, and the ciphertext matrix is obtained based on encryption algorithm Step 4, as shown below:

\[ R_i = [\{255\}] \oplus \left( \text{mod} \left( K_1(i) \times [\{255\}] \times K_2(i), 256 \right) \right), \quad (31) \]

\[ L_i = \text{mod} \left( [\{255\}] + R_i^2 + U_i, 256 \right). \quad (32) \]

From (31) and (32), if \([\{255\}]\) subblocks are chosen as the plaintext matrix for encryption, the ciphertext matrix can still be obtained, but \([\{255\}]\) utilizes the Hadamard product with keys \(K_1(i)\) and \(K_2(i)\) in \(\text{mod} \left( K_1(i) \times [\{255\}] \times K_2(i), 256 \right)\), and keys \(K_1(i)\) and \(K_2(i)\) cannot be separated. Therefore, the attacker still cannot obtain the encryption algorithm and key-related information. Thus, the algorithm proposed in this paper can resist chosen-plaintext attacks.

In addition, it can be seen from equations (4)–(9) that the encrypted key stream depends not only on the initial state value of the chaotic system but also on the plaintext image. When different plaintext images are encrypted, the corresponding key stream is different. Since the information obtained is related to the selected image, the attacker cannot obtain useful information by encrypting some special images. Therefore, the encryption scheme can resist the ciphertext-only attack, known as the plaintext attack and chosen-ciphertext attack.

4.9. Time Complexity Analysis. The complexity of the image encryption algorithm mainly refers to the time complexity. The time complexity of the encryption algorithm mainly includes running time and the big O complexity analysis algorithm [43]. Therefore, this paper analyses the complexity of the encryption algorithm in these two aspects. The experimental environment consists of an Intel Core i7, 2.3 GHz CPU, 8 GB of memory, and a 250 GB hard disk running the Windows 10 Professional operating system. For the Lena plaintext image of size \(M \times N\), the execution times of the encryption algorithms from the literature are shown in Table 20.

From the results shown in Table 20, it can be seen that the encryption speed of the proposed encryption algorithm is acceptable by comparison with other encryption algorithms.
4.10. Peak Signal-to-Noise Ratio Analysis. Peak signal-to-noise ratio (PSNR) is a parameter to describe the quality of image encryption; it is defined on the basis of the concept of mean square error (MSE). The MSE and PSNR are defined as follows:

\[
\text{MSE} = \left( \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} (I_{0}(i, j) - I_{e}(i, j))^2 \right)^{\frac{1}{2}}
\]

\[
\text{PSNR} = 20 \log\left( \frac{I_{\text{max}}}{\text{MSE}} \right)
\]
where $I_0(i,j)$ is the plaintext image pixel value, $I_e(i,j)$ is ciphertext image pixel value, $M$ and $N$ are the row and column of the image, respectively, and $I_{\text{max}}$ is the maximum pixel value of the image. For the image, the smaller the value of PSNR is, the greater the difference between plaintext image and ciphertext is, and the better the encryption effects. The Lena ciphertext images obtained by the proposed encryption algorithm and the encryption algorithm of related literature are compared, as shown in Table 21.

It can be seen from Table 21 that the PSNR of the ciphertext image of the proposed encryption algorithm is lower than that of the algorithms in other literature. Therefore, the proposed encryption algorithm has a good effect.

### 4.11. Robustness Analysis

In practice, the transmission of information is prone to multiple disturbances and attacks, so image encryption algorithms must be strongly robust. The robustness analysis of the image encryption algorithm includes antinoise attacks and clipping attacks. To test the antinoise attack performance of the proposed encryption algorithm in this paper, Gaussian noise of different intensities is added to the encrypted ciphertext image in Figure 16.

It can be seen that the decrypted image can still be distinguished, so it can be seen that the proposed encryption algorithm in this paper can resist the noise attack.

In practical applications, after the attacker intercepts the ciphertext image, he may use a cutting attack to destroy the image. To simulate this type of image data loss, an anticlipping attack experiment was performed. Taking $256 \times 256$ Lena image as the experimental subject, Figure 17 shows the experimental results.

It can be seen that the proposed encryption algorithm can still restore most of the original image information under different proportions of clipping attacks, so it has a certain ability to resist cutting attacks.

### 5. Conclusions

A new chaotic system is proposed in this paper and is combined with GANs and DNA encoding operations to generate fusion random sequences and DNA random sequences. These two types of random sequences are utilized as the key stream and combined with an improved Feistel network scrambling and diffusion mechanism and one-time pad mechanism to encrypt images, achieve local encryption and overall encryption, thereby obtaining better encryption security. The experimental simulation results indicate that the proposed encryption algorithm has a large key space and high sensitivity, and it can resist exhaustive attacks. The pixel values of the encrypted image are evenly distributed, and the correlation between adjacent pixel values is weak, indicating that the algorithm can effectively resist statistical analysis attacks. Therefore, the proposed algorithm can be applied in image encryption protection and it also provides a new idea for the combination of deep learning, GANs, DNA sequence coding, and image information security technology.

In future work, we plan to construct the parallel mechanism of key stream generation, which makes the training time of the generative adversarial networks model shorter and storage space smaller, to improve the overall efficiency of the encryption system. In addition, further development of the algorithm to be universally adaptable remains ongoing.

### Data Availability

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

### Conflicts of Interest

The authors declare that they have no conflicts of interest.

### Acknowledgments

The authors would like to sincerely thank the teachers in the research group. This work was supported by the National Natural Science Foundation of China under Grant 61973248 and the Key Project of Shaanxi Key Research and Development Program under Grant 2018ZDXM-GY-089.

### References

[1] D. Coppersmith, “The data encryption standard (DES) and its strength against attacks,” *IBM Journal of Research and Development*, vol. 38, no. 3, pp. 243–250, 1994.
[2] D. Shah and T. Shah, “A novel discrete image encryption algorithm based on finite algebraic structures,” *Multimedia Tools and Applications*, vol. 79, no. 37-38, pp. 28023–28042, 2020.
[3] X. Liu, D. Xiao, and C. Liu, “Quantum image encryption algorithm based on bit-plane permutation and sine logistic map,” *Quantum Information Processing*, vol. 19, no. 8, 2020.
[4] T. Etem and T. Kaya, “A novel true random bit generator design for image encryption,” *Physica A: Statistical Mechanics and Its Applications*, vol. 540, Article ID 122750, 2020.
[5] L. S. Sui, C. Du, X. Zhang, A. L. Tian, and A. Anand, “Double-image encryption based on interference and logistic map under the framework of double randomphase encoding,” *Optics and Lasers in Engineering*, vol. 122, pp. 113–122, 2019.
[6] F. Wei, Y. G. He, H. M. Li, and C. L. Li, "Cryptography of the integrated chaotic systems based image encryption algorithm," *Optik*, vol. 186, pp. 449–457, 2019.

[7] S. M. Ismail, L. A. Said, A. G. Radwan, A. H. Madian, and M. F. Abu-Elyazeed, "Generalized double-humped logistic map-based medical image encryption," *Journal of Advanced Research*, vol. 10, pp. 85–98, 2018.

[8] C. Pak and L. Huang, "A new color image encryption using combination of the 1D chaotic map," *Signal Processing*, vol. 138, pp. 129–137, 2017.

[9] N. Ramadan, H. E. H. Ahmed, S. E. Elkhamy, and E. Fathi, "Chaos-based image encryption using an improved quadratic chaotic map," *American Journal of Signal Processing*, vol. 6, no. 1, pp. 1–13, 2016.

[10] Q. Lu, C. X. Zhu, and G. J. Wang, "A novel S-box design algorithm based on a new compound chaotic system," *Entropy*, vol. 21, no. 10, 2019.

[11] B. Y. Irani, P. Ayubi, F. A. Jabalkandi, M. Y. Valandar, and M. J. Barani, "Digital image scrambling based on a new one-dimensional coupled Sine map," *Nonlinear Dynamics*, vol. 97, no. 4, pp. 2693–2721, 2019.

[12] X. Liu, Y. Song, and G.-P. Jiang, "Hierarchical bit-level image encryption based on chaotic map and feistel network," *International Journal of Bifurcation and Chaos*, vol. 29, no. 2, Article ID 1950061, 2019.

[13] X.-Y. Wang and Z.-M. Li, "A stream/block combination image encryption algorithm using logistic matrix to scramble," *International Journal of Nonlinear Sciences and Numerical Simulation*, vol. 20, no. 2, pp. 167–177, 2019.

[14] X. Zhang, L. Wang, G. Cui, and Y. Niu, "Entropy-based block scrambling image encryption using DES structure and chaotic systems," *International Journal of Optics*, vol. 2019, Article ID 3594534, 13 pages, 2019.

[15] X. Zhang, Z. Zhou, and Y. Niu, "An image encryption method based on the feistel network and dynamic DNA encoding," *IEEE Photonics Journal*, vol. 10, no. 4, pp. 1–14, 2018.

[16] R. I. Abdelfatah, "A new fast double-chaotic based Image encryption scheme," *Multimedia Tools and Applications*, vol. 79, no. 1-2, pp. 1241–1259, 2019.

[17] X. Wang and J. Yang, "A novel image encryption scheme of dynamic S-boxes and random blocks based on spatiotemporal chaotic system," *Optik*, vol. 217, Article ID 164884, 2020.

[18] S. Zhu and C. Zhu, "Secure image encryption algorithm based on hyperchaos and dynamic DNA coding," *Entropy*, vol. 22, no. 7, p. 772, 2020.

[19] L.-P. Chen, H. Yin, L.-G. Yuan, A. M. Lopes, J. A. T. Machado, and R.-C. Wu, "A novel color image encryption algorithm based on a fractional-order discrete chaotic neural network and DNA sequence operations," *Frontiers of Information Technology & Electronic Engineering*, vol. 21, no. 6, pp. 866–879, 2020.

[20] N.-R. Zhou, L.-X. Huang, L.-H. Gong, and Q.-W. Zeng, "Novel quantum image compression and encryption algorithm based on DQWT and 3D hyper-chaotic Henon map," *Quantum Information Processing*, vol. 19, no. 9, 2020.

[21] H.-S. Ye, N.-R. Zhou, and L.-H. Gong, "Multi-image compression-encryption scheme based on quaternion discrete fractional Hartley transform and improved pixel adaptive diffusion," *Signal Processing*, vol. 175, Article ID 107652, 2020.

[22] N. Zhou, W. Chen, X. Yan, and Y. Wang, "Bit-level quantum color image encryption scheme with quantum cross-exchange operation and hyper-chaotic system," *Quantum Information Processing*, vol. 17, no. 6, 2018.

[23] Z.-J. Huang, S. Cheng, L.-H. Gong, and N.-R. Zhou, "Nonlinear optical multi-image encryption scheme with two-dimensional linear canonical transform," *Optics and Lasers in Engineering*, vol. 124, Article ID 105821, 2020.

[24] X. Wang and S. Gao, "Image encryption algorithm based on the matrix semi-tensor product with a compound secret key produced by a Boolean network," *Information Sciences*, vol. 539, pp. 195–214, 2020.

[25] 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.

[26] J. Liang, Y. Xue, and J. Wang, "Bi-objective memetic GP with dispersion-keeping Pareto evaluation for real-world regression," *Information Sciences*, vol. 539, pp. 16–35, 2020.

[27] X. Wang and S. Gao, "Application of matrix semi-tensor product in chaotic image encryption," *Journal of the Franklin Institute*, vol. 356, no. 18, pp. 11638–11667, 2019.

[28] Y. Zhang, A. Chen, Y. Tang, J. Dang, and G. Wang, "Plaintext-related image encryption algorithm based on perceptron-like network," *Information Sciences*, vol. 526, pp. 180–202, 2020.

[29] X. Liu, X. Su, P. Shi, C. Shen, and Y. Peng, "Event-triggered sliding mode control of nonlinear dynamic systems," *Automatica*, vol. 112, Article ID 108738, 2020.

[30] R. May, "Simple mathematical models with very complicated dynamics," *Unsereil in Chaos*, vol. 261, pp. 85–93, 2017.

[31] H. Li, J. Xie, and W. Wei, "Permutation entropy and Lyapunov exponent: detecting and monitoring the chaotic edge of a closed planar under-actuated system," *Mechanical Systems and Signal Processing*, vol. 123, pp. 206–221, 2019.

[32] L. Gong, R. Wu, and N. Zhou, "A New 4D Chaotic system with coexisting hidden chaotic attractors," *International Journal of Bifurcation and Chaos*, vol. 30, no. 10, Article ID 2050142, 2020.

[33] M. Yang, K. Hu, Y. Du, Z. Wei, Z. Sheng, and J. Hu, "Underwater image enhancement based on conditional generative adversarial network," *Signal Processing: Image Communication*, vol. 81, Article ID 115723, 2020.

[34] H. J. Shiu, K. L. Ng, J. F. Fang, R. C. T. Lee, and C. H. Huang, "Data hiding methods based upon DNA sequences," *Information Sciences*, vol. 180, no. 11, pp. 2196–2208, 2010.

[35] S. M. Pincus, "Approximate entropy as a measure of system complexity," *Proceedings of the National Academy of Sciences*, vol. 88, no. 6, pp. 2297–2301, 1991.

[36] K. C. Jithin and S. Sankar, "Colour image encryption algorithm combining Arnold map, DNA sequence operation, and a Mandelbrot set," *Journal of Information Security and Applications*, vol. 50, Article ID 102428, 2020.

[37] R. Vidhya, M. Brindha, and N. Ammasai Gounden, "A secure image encryption algorithm based on a parametric switching chaotic system," *Chinese Journal of Physics*, vol. 62, pp. 26–42, 2019.

[38] S. Pare, A. K. Bhandari, A. Kumar, and G. K. Singh, "An optimal color image multilevel thresholding technique using grey-level co-occurrence matrix," *Expert Systems with Applications*, vol. 87, pp. 335–362, 2017.

[39] A. Babaei, H. Motameni, and R. Enayatifar, "A new permutation-diffusion-based image encryption technique using cellular automata and DNA sequence," *Optik*, vol. 203, Article ID 164000, 2020.

[40] C. Lin, X. Shen, and Z. Li, "Cryptographic analysis on the key space of optical phase encryption algorithm based on the design of discrete random phase mask," *Optics & Laser Technology*, vol. 49, pp. 108–117, 2013.
[41] B. Arpaci, E. Kurt, and K. Celik, “A new algorithm for the color image encryption via the modified Chua’s circuit,” *Engineering Science and Technology: An International Journal*, vol. 23, no. 3, pp. 595–604, 2019.

[42] Y. Qin, Y. Wan, and Q. Gong, “Learning-based chosen-plaintext attack on diffractive-imaging-based encryption scheme,” *Optics and Lasers in Engineering*, vol. 127, Article ID 105979, 2020.

[43] Z. Sun, G. Pedretti, P. Mannocci, E. Ambrosi, A. Bricalli, and D. Ielmini, “Time complexity of in-memory solution of linear systems,” *IEEE Transactions on Electron Devices*, vol. 67, no. 7, pp. 2945–2951, 2020.