mathematics

MDPI

Article

Secure DNA-Coding Image Optical Communication Using Non-Degenerate Hyperchaos and Dynamic Secret-Key

Heping Wen [1,2,3,*], Zhen Liu [1], Haowen Lai [1], Chongfu Zhang [2,*], Linhao Liu [1], Jieyi Yang [1], Yiting Lin [1], Yunqi Li [1], Yunlong Liao [1], Linchao Ma [1], Zefeng Chen [1] and Rui Li [1]

[1] Zhongshan Institute, University of Electronic Science and Technology of China, Zhongshan 528402, China
[2] School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China
[3] Guangdong Provincial Key Laboratory of Information Security Technology, Sun Yat-sen University, Guangzhou 510006, China
* Correspondence: wenheping@uestc.edu.cn (H.W.); cfzhang@uestc.edu.cn (C.Z.)

Abstract: With the aim of tackling insufficient security in the chaotic encryption algorithm for digital images in the Optical Access Network, a color image encryption scheme combining non-degenerate discrete hyperchaotic system and deoxyribonucleic acid (DNA) dynamic encoding is proposed. First, a new non-degenerate hyperchaotic system is constructed with all positive Lyapunov and more complex dynamic characteristics. Furthermore, the key sequence based on non-degenerate hyperchaotic system is generated using plaintext correlation to achieve the effect of a dynamic secret key. Next, a binary bit-planes permutation is performed on the image using one of the key sequences. Then, the chaotic key sequence is used to sequentially perform DNA encoding, obfuscation, and decoding. Finally, a binary bit-planes obfuscation is performed to obtain the final ciphertext. The research results show that the non-degenerate chaotic sequence can pass the NIST 800-22 test, and the corresponding encryption algorithm can resist various common attacks and has a strong anti-interference ability. In addition, the algorithm is verified on ARM-Embedded, which proves that the encryption system proposed in this paper is a feasible secure communication technology scheme. Therefore, the scheme proposed in this paper is helpful to provide new ideas for the design and application of high-security cryptosystem in optical access network.

Keywords: dynamic DNA coding; non-degenerate hyperchaos; image encryption; optical access network

MSC: 34C28; 68P25

Citation: Wen, H.; Liu, Z.; Lai, H.; Zhang, C.; Liu, L.; Yang, J.; Lin, Y.; Li, Y.; Liao, Y.; Ma, L.; et al. Secure DNA-Coding Image Optical Communication Using Non-Degenerate Hyperchaos and Dynamic Secret-Key. *Mathematics* **2022**, *10*, 3180. https://doi.org/10.3390/math10173180

Academic Editors: Antanas Cenys and Abdelmejid Bayad

Received: 21 July 2022
Accepted: 1 September 2022
Published: 3 September 2022

1. Introduction

Multimedia and Network technology have developed rapidly in the digital information age. Due to the advantages of large-capacity communication, long relay distance, and good confidentiality, the Optical Access Network is considered to be the development trend of next-generation communication [1–4]. Digital image is an important multimedia medium, and its privacy protection issue is particularly concerned, whereas traditional symmetric encryption algorithms such as Advanced Encryption Standard (AES) and International Data Encryption Algorithm (IDEA) do not consider the unique properties of digital images, and therefore face challenges in Optical Access Network of real-time encryption [5,6]. Chaos has the inherent randomness of a deterministic system, dense periodic points, ergodicity, and high sensitivity to initial values and control parameters [7–9]. It has many similarities with the permutation and scrambling of Shannon's cryptography. Since its emergence, it has received extensive attention from many experts and scholars [10–13], and is regarded as a security technology with great application [14,15]. However, on the one hand, many such algorithms do not have provable security, and on the other hand,

they are less combined with Optical Access Network communication, which is also an important problem that must be solved in the application [16].

A series of important theoretical and application results have been achieved in image encryption using digital chaos [17–21]. In recent years, quantum chaos [22], chaotic neural network [23], cell neural network [24], deoxyribonucleic acid (DNA) computing, and other interdisciplinary methods have been applied to image chaotic encryption technology to improve the security and effectiveness of the encryption algorithm [25–27]. Among them, the chaotic image encryption algorithm combined with DNA computing has the characteristics of good parallelism and high efficiency, which is a hot research topic, and it has become a new research direction for digital image chaotic encryption [28–35]. In 2020, the authors of [36] proposed a hyper-chaos-based image encryption algorithm that uses a 6-dimensional hyperchaotic system. DNA coding and operations are employed to change pixels. Theoretical analysis and numerical simulations demonstrate that the proposed algorithm is safe. E.E. García-Guerrero et al. [37] introduce a process to improve the randomness of five chaotic maps that are implemented on a PIC-microcontroller. The improved chaotic maps are tested to encrypt digital images in a wireless communication scheme. The experiment verified that this chaotic encryption scheme can be used in practical applications such as M2M and Internet of things (IoT). In 2021, the authors of [38] studied an image encryption algorithm based on multi-objective particle swarm optimization, DNA encoding sequence and one-dimensional Logistic map. The simulation results show that it has a better encryption effect. In order to enhance the security of images, Jiang et al. [39] proposed an asymmetric double color images cryptosystem introduced optical chaos technology that based on compression sensing (CS) and double random phase encoding (DRPE). Simulation results and security analysis confirm that the cryptosystem owns multi-layer protection, which is efficient and capable of guaranteeing the security of images and can provide an alternative solution for image security and privacy protection. In 2022, the authors of [40] presented an innovative image block encryption algorithm adopting fractional Fourier transform, hyperchaotic system, improved logistic map and DNA. The experimental results show the effectiveness and security of the cryptographic system. Zang et al. [41] proposed a method to construct a one-dimensional discrete chaotic system and design an image encryption scheme based on a uniformly distributed discrete chaotic system and DNA encoding. The experimental results demonstrate that our encryption algorithm has high key sensitivity and fast encryption speed and can resist differential and statistical attacks.

A large amount of research on the design and analysis of the combination of DNA computing and chaotic encryption algorithms for digital images have been proposed [42]. The contradiction between cryptographic design and analysis promotes the development of chaotic cryptography [43]. However, the current chaotic image encryption schemes combined with DNA computing generally still have the following shortcomings:

- Single DNA coding and operation rules that can be easily cracked;
- The encryption algorithm structure is unreasonable. Without adopting plain-related or ciphertext feedback, it is vulnerable to attack from known plaintext or chosen plaintext. In addition, in current image encryption, there are security risks in pixel-level scrambling;
- The security of the chaotic system or chaotic map of the existing chaotic cryptography is insufficient. The chaotic sequence generated by the existing low-dimensional chaotic system is difficult to pass the National Institute of Standards and Technology (NIST) test. The chaotic sequence has the risk of being estimated or identified.

Compared with the existing research, this paper innovatively carried out the following research:

- Self-designed non-degenerate hyperchaos with better performance to ensure the encryption effect.
- Different from static DNA encoding in the past, dynamic DNA encoding is used to improve the encryption effect.

- The plain-related parameters are used to generate dynamic keys to improve encryption performance.
- Hardware implementation is carried out in optical access network environment.
- Based on cryptography, the security of the algorithm is systematically analyzed.

Thus, this paper proposes an image encryption scheme that combines DNA dynamic encoding and robust hyperchaos, which improves the security of current schemes.

The rest of the paper is organized as follows. Section 2 concisely describes the DNA coding and computing rules, and the design method of non-degenerate chaotic system. Section 3 proposes the encryption algorithm designed in this paper. Section 4 presents the experimental and simulation results. The last section concludes the paper.

2. Related Theories

2.1. DNA Coding and Operation Rules

The introduction of coding and operation rules: there are four kinds of DNA, that is, adenine(A), thymine(T), cytosine(C), and guanine(G). According to the principle of permutation and combination, there are $4! = 24$ kinds of arrangements. However, DNA calculation should also meet the need of complementary rule, that is, A complements T; C complements G. Therefore, there are eight kinds of rules of DNA sequence coding that meet the need of complementary rule, as shown in Table 1. Moreover, there are three kinds of operations in DNA computing, that is, addition, subtraction, and XOR operation. The operation rules are shown in Table 2.

Table 1. DNA Base Complementary Rules.

Rules	1	2	3	4	5	6	7	8
A	00	00	01	01	10	10	11	11
T	11	11	10	10	01	01	00	00
G	01	10	00	11	00	11	01	10
C	10	01	11	00	11	00	10	01

The same information encoded by different DNA coding rules has different DNA sequences. For example, the 8-bit binary "00110110" of decimal numeral 54 adopts the first DNA coding rule, "ATGC", and the sixth coding rule, "CGTA". Similarly, different decoding rules lead to different restored information. For example, "ATCG" adopts the first decoding rule of "00111001", corresponding to decimal numeral 57. If it adopts the fourth decoding rule of "01100011", corresponding to decimal numeral 99. Obviously, the decoding results are different. The addition, subtraction, and XOR operation of DNA domain are similar to those in the binary system. Addition and subtraction are inverse operations to each other, but XOR is the inverse operation to itself.

Table 2. DNA Base Addition, Subtraction, and XOR Operation.

Base	Addition				Subtraction				XOR			
	AGCT				AGCT				AGCT			
A	A	G	C	T	A	T	C	G	A	T	C	G
G	G	C	T	A	G	A	T	C	G	C	T	A
C	C	T	A	G	C	G	A	T	C	G	A	T
T	T	A	G	C	T	C	G	A	T	A	G	C

2.2. Non-Degenerate Chaotic System

The general design method of non-degenerate discrete-time chaotic system is introduced below. The specific design steps are as follows. Firstly, design the asymptotically stable nominal system matrix C, so that the eigenvalues of the system are all located in the unit circle of the complex plane. Then, the nominal system matrix C is obtained by using the

nonsingular matrix P to carry out the similar transformation, thus getting $A = PCP^{-1}$, design the control matrix and uniform bounded anti-controller to implement the anti-control of the nominal system, and the globally bounded controlled system is obtained:

$$x_{k+1} = Ax_k + Bg(\sigma x_k, \varepsilon) \tag{1}$$

where Ax_k is the nominal system, B_g is the control matrix, $g(\sigma x_k, \varepsilon)$ is the uniform bounded anti-controller, σ and ε are parameters.

Finally, the control matrix and parameter controller are used for pole assignment of the controlled system Equation (1). After pole assignment, the number of positive Lyapunov exponents reaches the maximum, which meets the need of $LE_+ = n$, thus making the controlled system become a non-degenerate discrete-time hyperchaotic system.

According to the above design idea, a non-degenerate 3-D discrete-time chaotic system is designed. Firstly, all eigenvalues of the matrix C of the nominal system are in the unit circle. The matrix A is obtained by similarity transformation:

$$A = PCP^{-1} = \begin{pmatrix} 0.6500 & 0.1500 & -0.1500 \\ 0.3300 & 0.4700 & -0.3300 \\ 0.1800 & -0.1800 & 0.3200 \end{pmatrix} \tag{2}$$

Therefore, the iterative equation of asymptotically stable nominal system is expressed as:

$$\begin{cases} x_1(k+1) = a_{11}x_1(k) + a_{12}x_2(k) + a_{13}x_3(k) \\ x_2(k+1) = a_{21}x_1(k) + a_{22}x_2(k) + a_{23}x_3(k) \\ x_3(k+1) = a_{31}x_1(k) + a_{32}x_2(k) + a_{33}x_3(k) \end{cases} \tag{3}$$

The poles of the nominal system are assigned by matrix $E_{3\times3}$ and uniformly bounded inverse controller $g(\sigma x_k, \varepsilon_k)$.

$$g(\sigma x_k, \varepsilon_k) = \begin{cases} \bmod(\sigma_1 x_1(k), \varepsilon_1) \\ \bmod(\sigma_2 x_2(k), \varepsilon_2) \\ \bmod(\sigma_3 x_3(k), \varepsilon_3) \end{cases} \tag{4}$$

where mod is the fetch operation.

The results of the designed discrete-time chaotic system are shown in Equations (5) and (6).

$$\begin{aligned} \begin{pmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{pmatrix} &= A_{3\times3} \begin{pmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{pmatrix} + E_{3\times3} g(\sigma x_k, \varepsilon_k) \\ &= A_{3\times3} \begin{pmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \bmod(\sigma_1 x_1(k), \varepsilon_1) \\ \bmod(\sigma_2 x_2(k), \varepsilon_2) \\ \bmod(\sigma_3 x_3(k), \varepsilon_3) \end{pmatrix} \end{aligned} \tag{5}$$

$$\begin{cases} x_1(k+1) = a_{11}x_1(k) + a_{12}x_2(k) + a_{13}x_3(k) + \bmod(\sigma_1 x_1(k), \varepsilon_1) \\ x_2(k+1) = a_{21}x_1(k) + a_{22}x_2(k) + a_{23}x_3(k) + \bmod(\sigma_2 x_2(k), \varepsilon_2) \\ x_3(k+1) = a_{31}x_1(k) + a_{32}x_2(k) + a_{33}x_3(k) + \bmod(\sigma_3 x_3(k), \varepsilon_3) \end{cases} \tag{6}$$

where $\sigma_1 = \sigma_2 = \sigma_3 = \sigma, \varepsilon_1 = \varepsilon_2 = \varepsilon_3 = \varepsilon$. Lyapunov exponent of chaotic system is bigger with the σ and the ε. When $\sigma = 2 \times 10^5$, $\varepsilon = 4 \times 10^4$, the Lyapunov exponent of 3-D discrete-time chaotic system is $LE_i = 12.2061$, $i = 1, 2, 3$. All the Lyapunov exponent are positive, which meets the need of $LE_+ = n$ and it is a non-degenerate discrete-time hyperchaotic system.

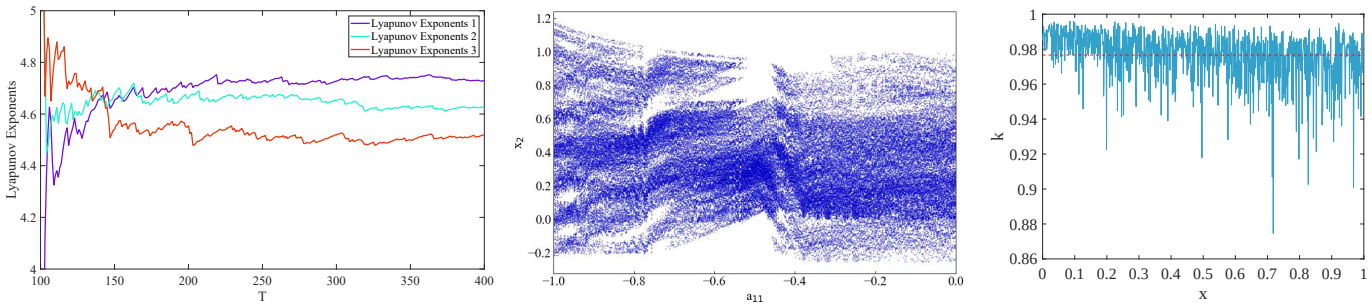
The NIST 800-22 test suite is a statistical package of 16 tests for testing random(arbitrary length) binary sequences produced by the hardware of software-based cryptographic ran-

dom or pseudorandom number generators. In this test, all sequences needed in encryption passed the test successfully, and the partial test results are shown in Table 3.

Table 3. NIST-800-22 test results.

Statistical Tests	*p*-Values			Results
	Seq 1	Seq 2	Seq 3	
Frequency (Monobit) Test	0.494392	0.191687	0.851383	successful
Block-Frequency Test	0.304126	0.494392	0.935716	successful
Cumulative-Sums Test	0.191687	0.494392	0.262249	successful
Runs Test	0.534146	0.911413	0.319084	successful
Longest-Run Test	0.554420	0.334538	0.102526	successful
Binary Matrix Rank Test	0.595549	0.001628	0.494392	successful
Discrete Fourier Transform Test	0.051942	0.798139	0.798139	successful
Non-Overlapping Templates Test	0.006661	0.008266	0.008879	successful
Overlapping Templates Test	0.759756	0.366918	0.455937	successful
Maurer's Universal Statistical Test	0.494392	0.191687	0.350485	successful
Approximate Entropy Test	0.108791	0.983453	0.051942	successful
Random-Excursions Test (x = −4)	0.022503	0.048716	0.148094	successful
Random-Excursions Variant Test (x = −9)	0.022503	0.122325	0.014216	successful
Serial Test-1	0.798139	0.637119	0.897763	successful
Serial Test-2	0.383827	0.040108	0.851383	successful
Linear-Complexity Test	0.224821	0.236810	0.455937	successful

The Lyapunov exponent is a numerical value of statistical characteristics. The bifurcation diagram is to describe the output ranges of a dynamical system along with its parameter's change. The 0–1 Gottwald-Melbourne test can determine the regular motion and chaotic motion by calculating the parameter k asymptotically close to 0 or 1. Figure 1 shows the bifurcation diagram, Lyapunov exponent, and the 0–1 Gottwald-Melbourne test results. The three-dimensional non-degenerate discrete chaotic system has three positive Lyapunov exponents, which are 4.6848, 4.6175, and 4.5113 respectively. All three are positive values, which proves that the three-dimensional non-degenerate discrete chaotic system is a hyper-chaotic system. From its bifurcation diagram, we can see that, when a_{11} is close to -0.5, the chaos performance becomes better. In the 0–1 Gottwald-Melbourne test, the k value of the average result of 10,000 times is 0.9985, which is close to the theoretical value of 1, which verifies the excellent performance of the chaotic system The results are shown in Figure 1.



(**a**) (**b**) (**c**)

Figure 1. The chaos performance of non-degenerate chaotic maps: (**a**) Lyapunov exponent; (**b**) Bifurcation diagrams; (**c**) The 0–1 Gottwald-Melbourne test.

3. The Proposed Encryption Algorithm

The proposed algorithm includes initial value generation, plaintext image bit-level scrambling, DNA dynamic coding, DNA domain obfuscation encryption, and DNA decoding. The block diagram of the encryption machines of the cryptosystem is shown in Figure 2.

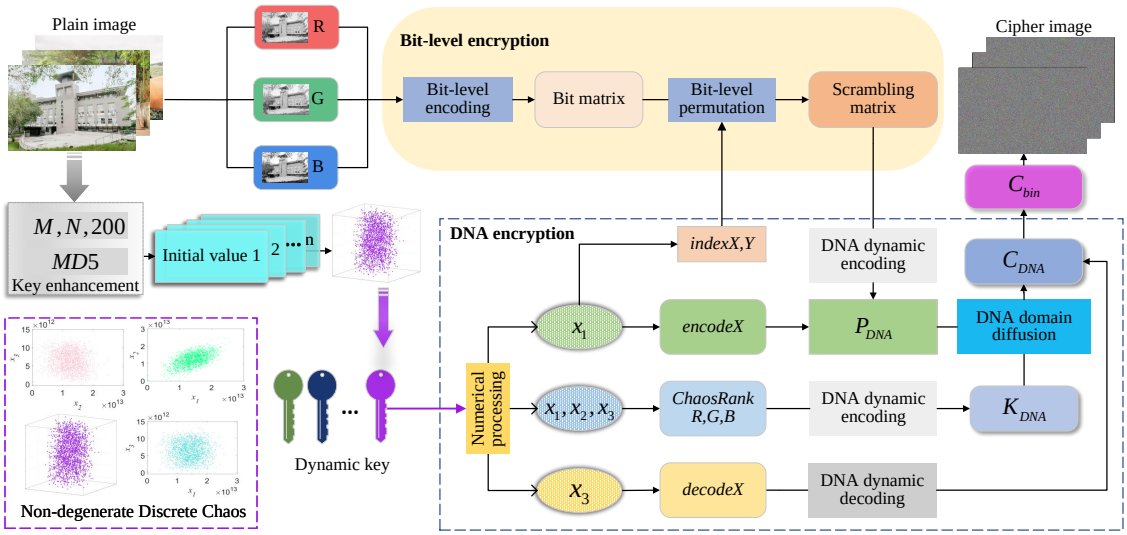


Figure 2. The block diagram of the encryption machines.

3.1. The Original Principles of Chaotic Sequences

Step 1: Select the key parameter.

The algorithm of this paper selects the $MD5$ value of plaintext, asymmetrically systematic parameter of non-degenerate chaotic system, feedback controlling parameter and original value as four kinds of parameters. To make chaotic key sequences more sensitive to plaintexts and keys, a total perturbation parameter $sumK$ is generated after all key parameters are treated as follows:

$$sumK = \sum a_{ij} + \sin\left(\sum \sigma_i + \sum \varepsilon_i\right) + \sum x_i(0) \bmod 1 \tag{7}$$

where $sumK$ is perturbation parameter.

Step 2: Preprocessing of plaintext $MD5$.

Take the 128-bit hash value of the plaintext, divide it into 8 16-bit , and then process it into 3 disturbance parameters.

$$n_j = \text{hex2dec}(h_{2j-1} + h_{2j})/(3 \times 2^{16}), \quad j = 1, 2 \ldots, 5 \tag{8}$$

where hex2dec is a function that converts a hexadecimal number represented by a string to a decimal number.

Step 3: Initial value disturbance.

The two perturbation parameters are slightly perturbed to the initial value and feedback control parameters of the chaotic system according to the following methods:

$$\begin{cases} x_1' = x_1 + sumK + n_1 \\ x_2' = x_2 + sumK + n_2 \\ x_3' = x_3 + sumK + n_3 \end{cases} \tag{9}$$

$$\begin{cases} \sigma_i' = \sigma_i + n_4 \\ \varepsilon_i' = \varepsilon_i + n_5 \end{cases} \tag{10}$$

Step 4: Initial value processing.

Eliminating the harmful initial transients of chaotic mapping. Three sequences are obtained and expressed as $x_i(k), i = 1, 2, 3$.

$$\begin{cases} encodeX = \bmod\left(|x_2| \times 10^{14} - \left\lfloor |x_2| \times 10^{14} \right\rfloor, 8\right) \\ decodeX = \bmod\left(|x_3| \times 10^{14} - \left\lfloor |x_3| \times 10^{14} \right\rfloor, 8\right) \end{cases} \tag{11}$$

where the lengths of *encodeX* and *decodeX* are both $M \times N$, and each value controls the encoding and decoding rules of 12 DNA characters.

Step 5: Preprocess each chaotic sequence.

The applications of the three chaotic sequences are as follows: the first chaotic sequence is used to generate the row and column index of the image, and to scramble the color image at bit level. The second and third chaotic sequences control the dynamic encryption and decryption of DNA matrix respectively; the first to third chaotic sequences are used to generate the R, G, and B components of the key image of the size $M \times N$. In order to achieve bitrate scrambling, the color image is transformed into $3M \times 8N$ bit matrix.

$$\begin{cases} ChaosRankR = \bmod\left(\left(|x_1| \times 10^{14} - \lfloor |x_1| \times 10^{14} \rfloor\right) \times 10^3, 256\right) \\ ChaosRankG = \bmod\left(\left(|x_2| \times 10^{14} - \lfloor |x_2| \times 10^{14} \rfloor\right) \times 10^3, 256\right) \\ ChaosRankB = \bmod\left(\left(|x_3| \times 10^{14} - \lfloor |x_3| \times 10^{14} \rfloor\right) \times 10^3, 256\right) \end{cases} \tag{12}$$

where *ChaosRankR*, *ChaosRankG*, *ChaosRankB* as R, G, B three components are used to generate the key image of DNA domain obfuscation encryption.

3.2. DNA Dynamic Encoding

The first sequence generated by discrete 3-D hyper-chaos is used to control the DNA encoding method of the image. The key image $ChaosRankR, ChaosRankG, ChaosRankB$ constituted by the first to the third chaotic sequences is adjusted to the image bit matrix $M \times 24N$ of K_{bin} in the same way, and then the DNA matrix of K_{bin} is generated by the same encoding method. The scrambling bit matrix P_{bin} and the key bit matrix K_{bin} are represented as P_{DNA} and K_{DNA} respectively after DNA coding. The specific process of DNA dynamic coding is as follows:

$$\begin{cases} P_{DNA}(12(i-1)+1) : (12(i-1)+12) = DNA\_encode(P_{bin}(24(i-1)) + 1 : (24(i-1)+24), encodeX) \\ K_{DNA}(12(i-1)+1) : (12(i-1)+12) = DNA\_encode(K_{bin}(24(i-1)) + 1 : (24(i-1)+24), encodeX) \end{cases} \tag{13}$$

3.3. Diffusion

To make the encryption algorithm has a stronger ability to resist differential attacks, the diffusion method of ciphertext feedback in DNA domain makes the relationship between plaintext, key, and ciphertext more complex. The DNA domain diffusion process is expressed as:

$$\begin{cases} C_{DNA}(i) = (P_{DNA}(i) + K_{DNA}(i)) \oplus K_{DNA}(i) + c0 & i = 1 \\ C_{DNA}(i) = (P_{DNA}(i) + K_{DNA}(i)) \oplus K_{DNA}(i) + C_{DNA}(i-1) & i = 2, 3, \cdots, 12M \times N \end{cases} \tag{14}$$

where "+" and "$\oplus$" are addition and XOR operations in DNA domain, corresponding to Table 2, c_0 are initial diffusion key parameters of DNA. The final DNA ciphertext was obtained after C_{DNA} ciphertext feedback.

3.4. DNA Dynamic Decoding

The DNA decoding process is similar to the encoding process, but it is worth noting that decoding and encoding control sequences are different, so it is not a simple inverse process, but equivalent to double encryption transformation. The DNA decoding process is expressed as:

$$C_{bin}(24(i-1)+1 : (24(i-1)+24)) = DNA\_decode(C_{DNA}(12(i-1)+1 : (12(i-1)+12), decodeX)) \tag{15}$$

where C_{bin} is transformed into a color ciphertext image Crgb of $M \times N \times 3$ after size transformation. Therefore, the whole encryption process is completed. Decoding algorithm is the inverse process of encryption, which will not be described here. To describe the DNA domain encryption of the encryption system more clearly, Figure 3 is given as an example.

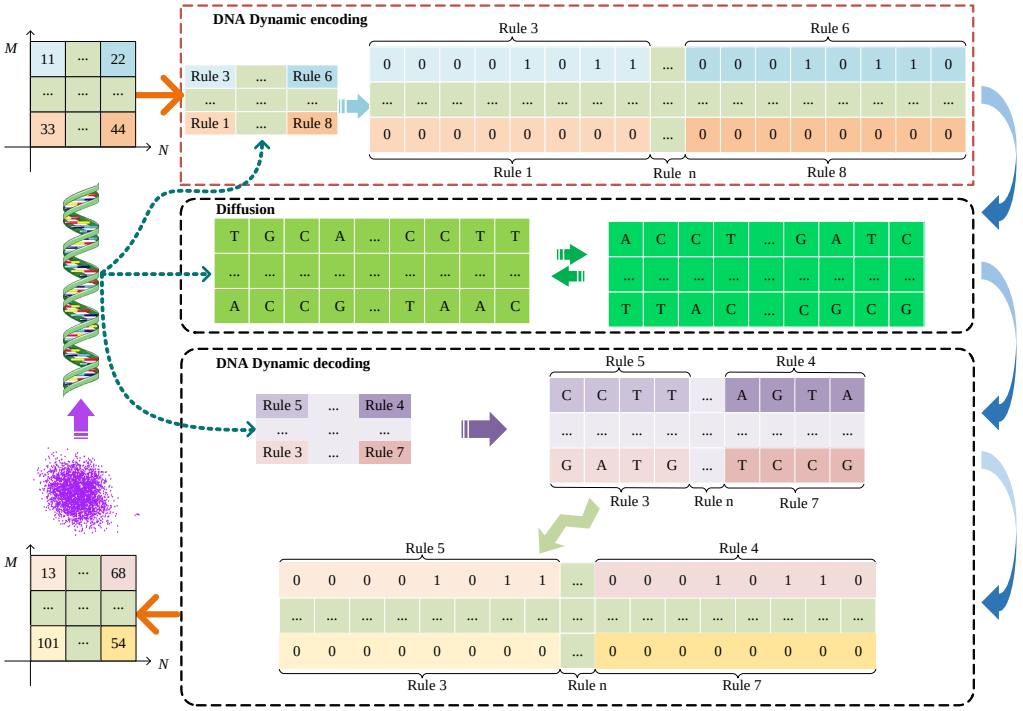


Figure 3. The mechanism process of $M \times N$ image encryption.

4. Experimental Results and Analysis Discussion

4.1. Experimental Settings and Results

4.1.1. Experimental Environment

The experiment is operated in a PC installing system of Windows 10 64 bit while the environment is MATLAB r2019b, the processor is Intel(R) Core (TM) i5-10200H CPU @ 2.40 GHz and install memory is 16 GB. In order to compare with other algorithms easily, the experiment is conducted on multiple different colored images and takes classical Lena colored image, with size 256 × 256, as the main experimental subject. Among them, the encrypted time of the encrypted Lena colored image is 2.5338 s. The images before and after encryption are shown in Figure 4. The image after encryption is shown as snowflakes with a better encryption effect.

Moreover, to verify the effectiveness and feasibility of the cryptosystem, we carried out experiments on the Optical Access Network experimental platform based on ARM Embedded system. This digital image secure communication system based on the Optical Access Network is mainly composed of two sets of ARM Embedded system development board and a Gigabit single mode single fiber optical transceiver TP-LINKTL-FC311A-3. The maximum transmission distance is about 10 km and the maximum transmission rate is above 155 Mbit/s. The ARM development main board is Raspberry Pi 4B, the chip is Broadcom BCM2711 of Cortex-A72 architecture, and the operating system is 32-bit Linux 5.4, which is equipped with 3.5-inch Liquid Crystal Display (LCD) display. The wireless router is used for network communication at the sending end and the receiving end, and the address is obtained by Dynamic Host Configuration Protocol (DHCP), which are 192.168.1.114 and 192.168.1.115 respectively. The sending end is responsible for the display, encryption and transmission of the plain images, while the receiving end stores, displays and decrypts the receiving cipher images. The experimental hardware platform and experimental results are shown in Figure 5. The encryption and decryption time of the embedded end are 1918 and 1846 milliseconds respectively, reaching the expected ideal value. As shown in Figure 5, the encrypted images, which improve the security of secure communication. In addition, the experimental environment is aimed at common Optical Access Network secure communication platforms, so it is more universal.

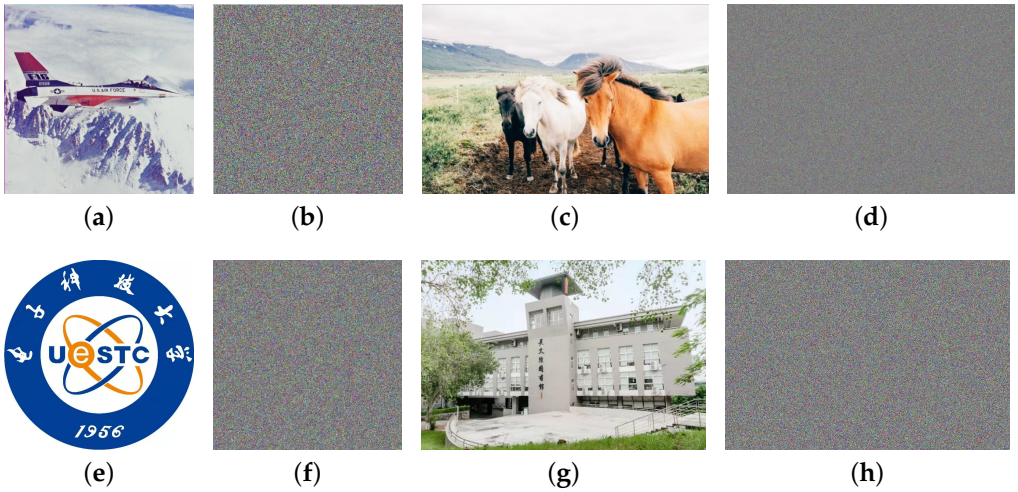


Figure 4. Images before and after encryption: (**a**) 1# plain image; (**b**) Cipher image of (**a**); (**c**) 2#plain image; (**d**) Cipher image of (**c**); (**e**) 3#plain image; (**f**) Cipher image of (**e**); (**g**) 4#plain image; (**h**) Cipher image of (**g**).

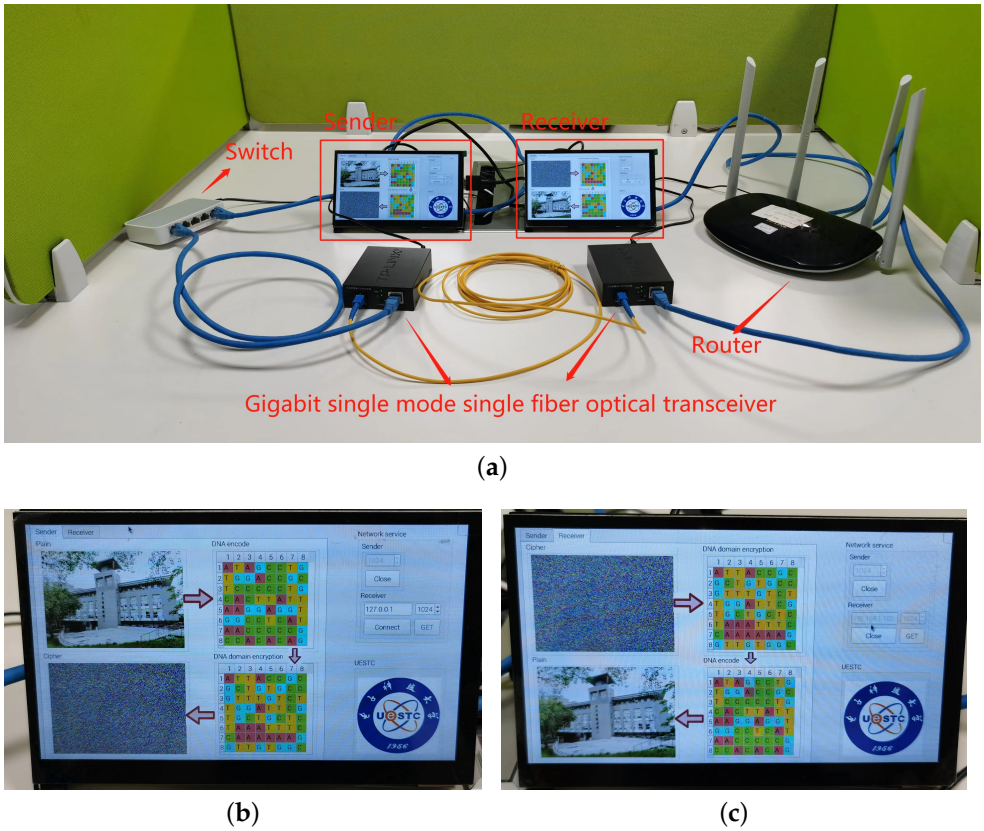


Figure 5. Experimental result in Optical Access Network secure communication platform: (**a**) The overall physical diagram; (**b**) 4# plain image; (**c**) Cipher image of (**b**).

4.1.2. Its Application in Optical Access Network

This image cryptosystem can be applied in a general Optical Access Network secure communication scenario. A schematic diagram for Optical Access Network secure communication is illustrated in Figure 6. Both the sender and receiver are embedded terminals that store, display and transmit digital images. To enhance the security of the information, we encrypt the image information at the sender side and then send it to two receivers separately. The first receiver receives a normal cipher image, while the second receiver receives

a corrupted cipher image. When a legitimate user uses the correct key, the ciphertext image can be restored effectively. In addition, the attacker can also obtain the ciphertext image using special means. However, without the correct key, the ciphertext image cannot be restored. Therefore, our proposed encryption scheme is applied in different Optical Access Network communication environment.

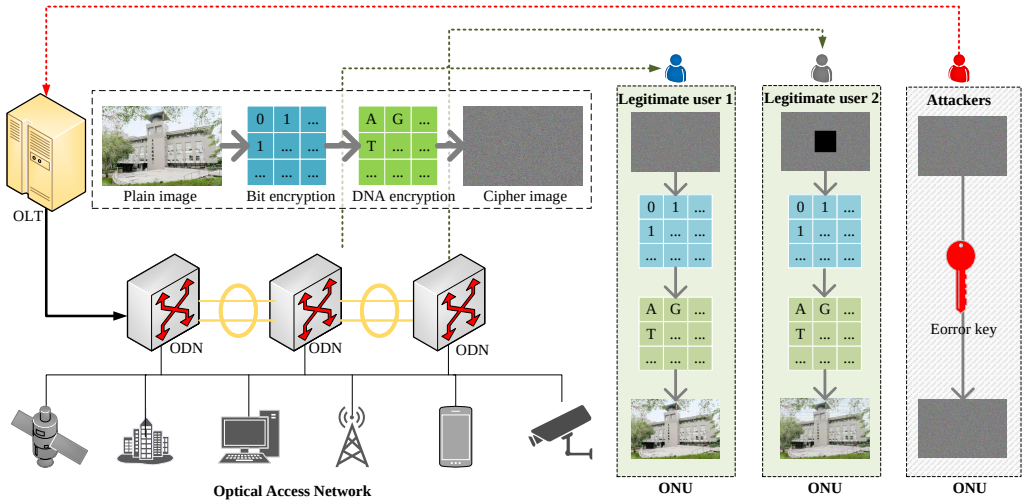


Figure 6. A diagram of the image cryptosystem used in Optical Access Network secure communication.

4.2. Theoretical and Statistical Security Analysis

Recently, it is pointed out that the current chaotic image encryption scheme is mainly aimed at statistical analysis and differential analysis. In cryptography, statistical analysis and differential analysis are necessary, but not sufficient, on this basis this article performs analysis as well as discussion on several aspects like chaotic system, NIST test of chaotic sequence, statistical analysis, information entropy, difference analysis and exhaustive attack analysis [44,45].

4.2.1. Key Sensitivity

In image encryption, key sensitivity performance is often used as an important indicator to measure the security of an encryption system. The key sensitivity is generally expressed by the difference between corresponding images when decrypting or encrypting the same image with a slightly different key. In order to test the sensitivity to the key in the scheme, the three sequences generated by the initial key are superimposed on each other to form a color map, as shown in Figure 7. The parameter $x_1(0)$ of the initial key is scrambled with minimum precision to generate a new key, and the three new chaotic sequences are combined into a new color image in the same way, as shown in Figure 7. By differentiating the two field images, the difference map and its corresponding histogram can be obtained. From the figure, we can see that the difference value of the difference histogram of the chaotic series is mainly concentrated around 0, and there is still a large amount of color data in the difference map, which proves that the chaotic sequence generated by the slight change of the initial chaotic value is very different.

We also processed the two chaotic series by means of time series, and compared the generated three-dimensional chaotic series respectively. The experimental results are shown in Figure 8. The sensitivity of the chaotic system to the initial value is also proved.

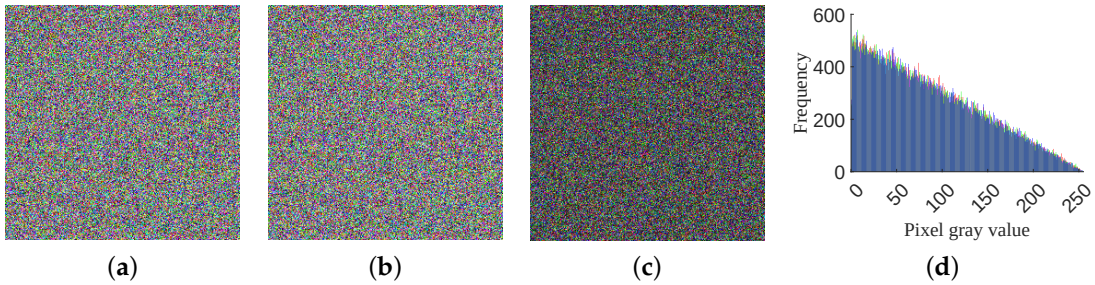


Figure 7. Chaotic sequence sensitivity: (**a**) Primal Chaos Sequence; (**b**) Chaos sequence image after change; (**c**) Chaotic sequence difference graph; (**d**) Chaotic sequence difference histogram.

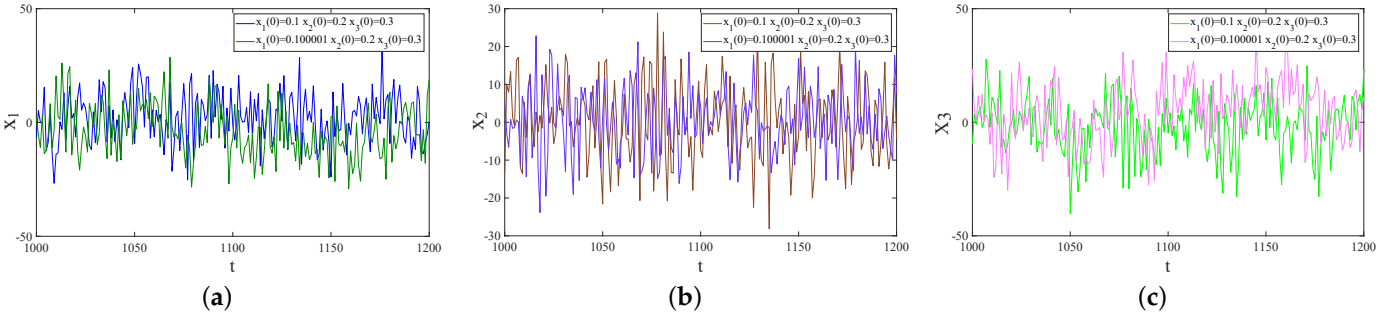


Figure 8. Key sequence sensitivity timing diagram: (**a**) Comparison before and after $x_1(0)$ key perturbation; (**b**) Comparison x_2 before and after $x_1(0)$ key perturbation; (**c**) Comparison x_3 before and after $x_1(0)$ key perturbation.

4.2.2. Key Space

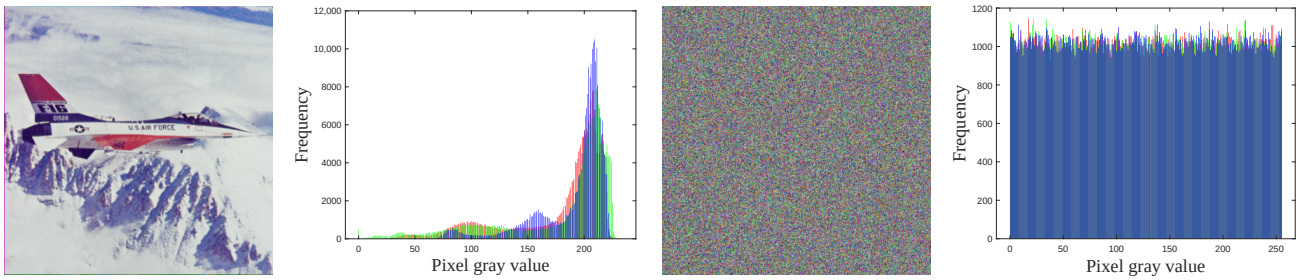
Cryptography experts pointed out that in order to improve the ability to resist exhaustive attacks, the key length of chaotic passwords should not be less than 128 bits. The key space can be denoted as $S \in \{a_{ij}, \delta_i, \epsilon_i, x_i(0), Hash256\}, i, j = 1, 2, 3$. Including SHA2 hash value, nominal system parameters of non-degenerate hyperchaotic system, feedback controller parameters and initial value of four different types of keys, a total of 55 key parameters. It can be seen from the experimental analysis that the key length of the nominal system parameters is $10^{16\times9} \approx 2^{144}$, the key length of the parameter part of the feedback controller is $10^{11\times6} = 10^{66} \approx 2^{219}$, the key length of the chaotic initial value part is $10^{17+16+15} = 10^{48} \approx 2^{159}$, the SHA2 hash has a password length of 2^{256}. Therefore, the total key length is $144 + 219 + 159 + 256 = 778$ bits. It can be seen from Table 4 that, for $\delta_i, \epsilon_i, x_i(0)$, compared with other existing encryption schemes, the key space of this paper has obvious advantages.

Table 4. Key space size comparison table.

	This Article	Ref. [27]	Ref. [39]	Ref. [46]
Key space/bits	778	128	378	309

4.2.3. Statistics Histogram

The RGB component histograms of the plaintext image, the bit-scrambled image, and the ciphertext image encrypted in the DNA domain are shown in Figure 9. Compared with the plaintext image, the histogram after bit scrambling has changed to some extent. The histograms of the encrypted three channels all show a pseudonoise distribution state, which verifies that the ciphertext has good statistical properties.



(**a**) (**b**) (**c**) (**d**)

Figure 9. Image and histogram before and after bit-level scrambling and DNA encryption: (**a**) Plain image of $1^{\#}$; (**b**) Histogram of the plain image of $1^{\#}$; (**c**) Cipher image of $1^{\#}$; (**d**) Histogram of the ciper image of $1^{\#}$.

4.2.4. The Coefficient of Adjacent Pixels

Normally correlation coefficient is used to compare the relevant properties between plaintext image and ciphertext image. (x, y) are pixels of two different images and the correlation coefficient between them is defined as follows:

$$\begin{cases} r_{xy} = \dfrac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \\ cov(x,y) = \dfrac{1}{N}\displaystyle\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \\ D(x) = \dfrac{1}{N}\displaystyle\sum_{i=1}^{N}(x_i - E(x))^2 \\ E(x) = \dfrac{1}{N}\displaystyle\sum_{i=1}^{N} x_i \end{cases} \tag{16}$$

where $cov(x, y)$ is the covariance between pixel values x and y, N is the number of adjacent pixel pairs of the image to be analyzed. $E(x)$ and $D(x)$ are respectively the expected and mean square error of pixel value x. r_{xy} is the correlation coefficient of pixel value x and y. $D(y)$ is the mean square error of the pixel value y. Adjacent pixels in the horizontal, vertical, and diagonal direction of plaintext image have a stronger correlation yet adjacent pixels of cipher text image have no correlation. All three components R, G, B have similar statistic properties, and the detailed experimental results are shown in Figure 10. Table 5 shows the encryption quality of the proposed scheme and the classic encryption schemes in recent years.

Table 5. Comparison results of correlation coefficients of adjacent pixels.

Component	Direction	Original Image	Algorithm in This Paper	Ref. [47]	Ref. [48]	Ref. [49]
R	Horizontal	0.9752	−0.0352	−0.0063	0.0076	0.0023
	Vertical	0.9754	0.0402	−0.0016	0.0017	−0.0130
	Diagonal	0.9321	0.0122	0.0156	0.0110	−0.0061
G	Horizontal	0.9660	0.0092	−0.0032	−0.0048	−0.0236
	Vertical	0.9627	0.0082	0.0335	0.0274	0.0308
	Diagonal	0.9395	0.0038	−0.0095	0.0342	−0.0179
B	Horizontal	0.9484	0.0024	−0.0044	−0.0056	−0.0266
	Vertical	0.9667	−0.0185	−0.0079	0.0150	−0.0057
	Diagonal	0.9075	−0.0275	0.0034	−0.0115	0.0378

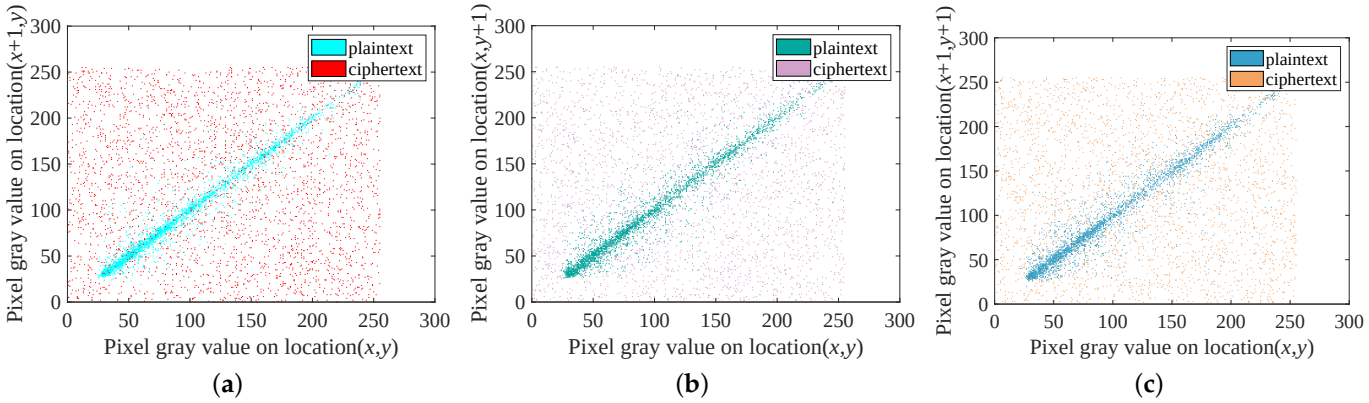


Figure 10. Adjacent pixels' correlation of plaintext images and ciphertext images: (**a**) Horizontal correlation of plaintext images and ciphertext images; (**b**) Vertical correlation of plaintext images and ciphertext images; (**c**) Diagonal correlation of plaintext images and ciphertext images.

4.2.5. Information Entropy

Information entropy, as an important measurement index in image encryption, reflects the uncertainty of image information. In general, the larger the information entropy value, the greater the uncertainty of image information, and the smaller the visibility of information, indicating that the encryption performance of the algorithm is better. Therefore, we compare the information entropy of the images before and after encryption. The experimental result is shown in Table 6 and the formula is as follows:

$$H = -\sum_{i=0}^{L} p(i) log_2 p(i) \tag{17}$$

where $p(i)$ is the probability that the gray value i appears. L is the number of gray levels of the image. The algorithm information entropy proposed in this paper is close to the theoretical value 8, which has a certain improvement compared with the similar references.

Table 6. Image information entropy.

Image	R	G	B	S
Original image	7.5549	7.0167	6.7347	7.4235
Ciphertext image in this paper	7.9997	7.9968	7.9969	7.9991
Ref. [47]	7.9993	7.9994	7.9994	7.9998
Ref. [48]	7.9992	7.9993	7.9992	7.9998
Ref. [49]	7.9993	7.9993	7.9993	7.9998

4.2.6. Differential Analysis

In image encryption algorithms, the measurement of plaintext sensitivity usually uses the number of Number of Pixels Changes Rate(NPCP), Unified Average Changing Intensity(UACI), and Block average changing intensity(BACI) indexes, the calculation equations of which are:

$$\begin{cases} NPCR = \dfrac{1}{H \times W} \displaystyle\sum_{i=1}^{H} \sum_{j=1}^{W} D_{ij} \times 100\% \\ UACI = \dfrac{1}{H \times W} \displaystyle\sum_{i=1}^{H} \sum_{j=1}^{W} \left(\dfrac{x(i,j) - x'(i,j)}{255} \right) \times 100\% \\ BACI = \dfrac{1}{(H-1)(W-1)} \displaystyle\sum_{i=1}^{(H-1)(W-1)} \dfrac{m_i}{255} \end{cases} \tag{18}$$

$$D_{ij}=\begin{cases} 1, & x(i,j) \neq x'(i,j) \\ 0, & x(i,j) = x'(i,j) \end{cases} \tag{19}$$

where $H \times W$ is the size of the image while x and x' are separately the cipher texts, which changed a pixel for the plain text. m_i is an average value of the absolute value of the difference between any two elements. The theoretical values of NPCR, UACI, and BACI are 99.5865%, 29.9530%, and 22.8812%, which are exceedingly close to the theoretical values and show that the encrypted algorithm has a strong ability to resist plain text attack. The experimental results are shown in Figure 11.

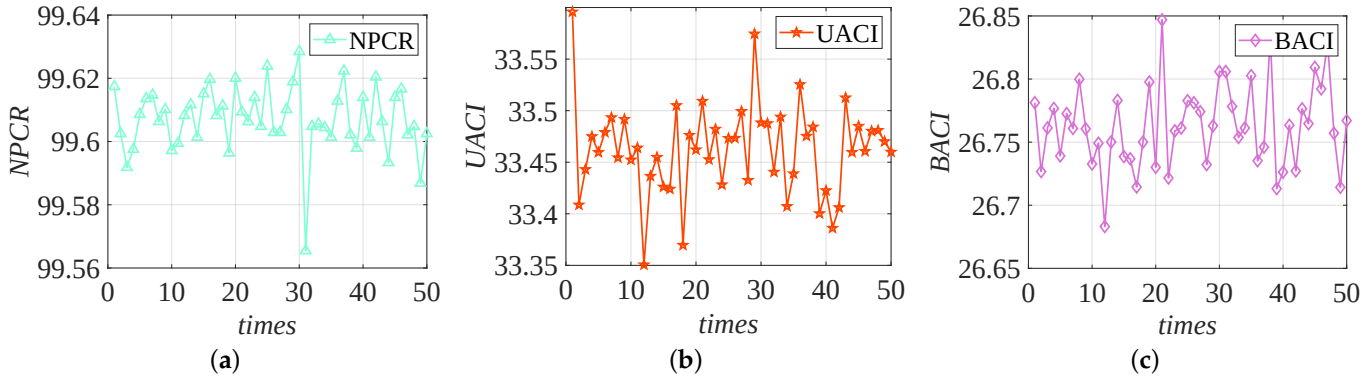


Figure 11. NPCR (**a**), UACI (**b**) and BACI (**c**) values obtained from 50 experiments.

4.2.7. Resistance to Noise Attack

- Resistance to salt and pepper noise attack

We separately added 5%, 10%, and 20% salt and pepper noise into the plaintext image. We can see from Figure 12 that the image adding noise can still have effective recognizable image information after decryption.

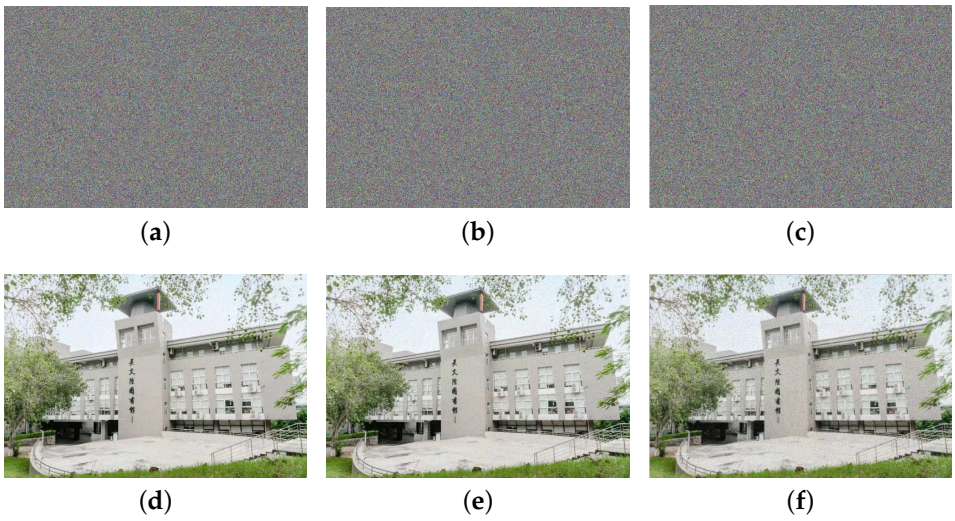


Figure 12. The ciphertext and decryption image after adding salt-and-pepper noise: (**a**) 5% salt-and-pepper noise ciphertext images; (**b**) 10% salt-and-pepper noise ciphertext images; (**c**) 20% salt-and-pepper noise ciphertext images; (**d**) 5% salt-and-pepper noise decryption images; (**e**) 10% salt-and-pepper noise decryption image; (**f**) 20% salt-and-pepper noise decryption image.

- The ability to defend against occlusion noise attack

We respectively added occlusion noise of sizes 96 × 96, 200 × 200, and 300 × 300, into the ciphertext image and we can see from Figure 13 that the image adding noise can still have effective recognizable image information after decryption.

In order to better present the difference between the restored plaintext image and the original plaintext image after being attacked, we have ten different degrees of attacks, and calculate the Structural Similarity (SSIM) values of the two. The experimental results are shown in Figure 14.

$$SSIM(p,c) = \frac{\left(2\mu_p\mu_c + (0.01L)^2\right)\left(2\sigma_{pc} + (0.03L)^2\right)}{\left(u_p^2 + u_c^2 + (0.01L)^2\right)\left(\sigma_p^2 + \sigma_c^2 + (0.03L)^2\right)} \tag{20}$$

where the average values of the plain image P and the cipher image C are denoted by u_p and u_c separately. The variance of the plain image and the cipher image denoted by σ_p^2 and σ_c^2 indicates that the covariance of the plain image and the cipher image is represented by σ_{pc}. $(0.01L)^2$ and $(0.03L)^2$ are used as constant numbers to maintain stability. L represents the dynamic range of pixel values. Figure 14 shows that the encryption quality of our proposed scheme has better cryptographic performance.

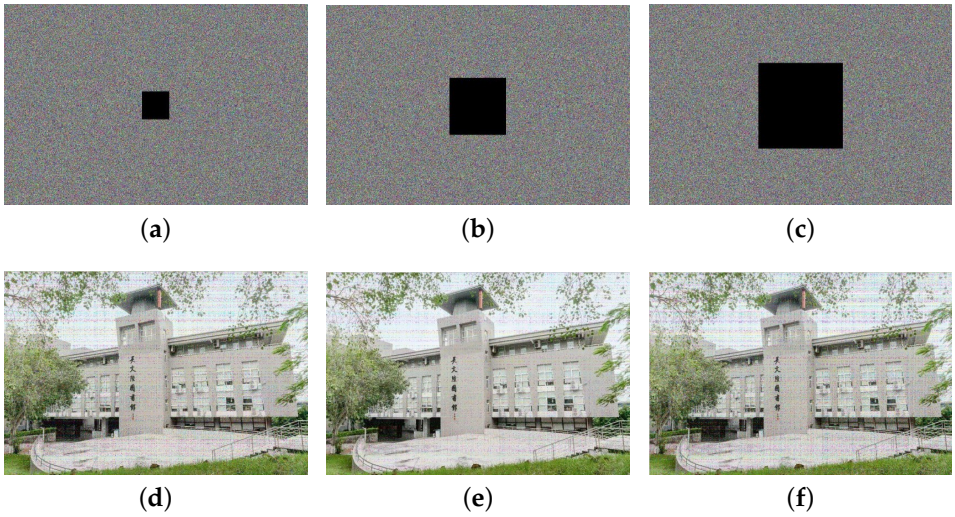


Figure 13. The ciphertext and decryption image after adding occlusion noise: (**a**) 96 × 96 occlusion noise ciphertext images; (**b**) 200 × 200 occlusion noise ciphertext images; (**c**) 300 × 300 occlusion noise ciphertext images; (**d**) 96 × 96 occlusion noise decryption images; (**e**) 200 × 200 occlusion noise decryption image; (**f**) 300 × 300 occlusion noise ciphertext images.

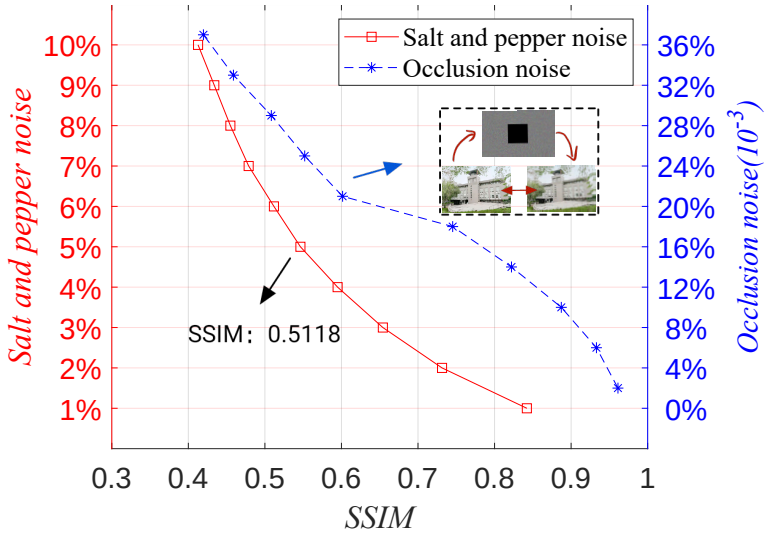


Figure 14. SSIM value under salt and pepper noise and occlusion noise.

5. Conclusions

Based on DNA computing and non-degenerate discrete hyperchaos, this paper proposes a color image encryption scheme, which combines DNA dynamic encoding as well as non-degenerate high-dimensional discrete chaos. This paper proposes that positive Lyapunov exponent non-degenerate three-dimensional discrete hyperchaos should be used as the core of encryption algorithm, and the methods of clear correlation, binary bit planes scrambling, DNA dynamic encoding and decoding as well as DNA domain ciphertext diffusion can be adopted to encrypt colored image. Experimental results and theoretical analysis show that it is better to adopt a non-degenerate high-dimensional discrete hyperchaotic key sequence which can pass the NIST 800-22 test. The encryption programmer can resist various kinds of common attacks. As a result, an image encryption programmer, which combines DNA dynamic encoding and discrete high-dimensional hyperchaos, has characteristics like higher operating speed, higher safety, and excellent robustness. In the future, we will continue to enhance the security of the cryptosystem on this basis, so as to provide a new method for designing and applying a more secure and reliable cryptosystem in the optical access network.

Author Contributions: Conceptualization, H.W. and C.Z.; methodology, Z.L.; software, H.L.; validation, L.L. and Y.L. (Yiting Lin); formal analysis, L.L.; investigation, Y.L. (Yunqi Li) and R.L.; data curation, Y.L. (Yunlong Liao) and L.M.; writing—original draft preparation, J.Y.; writing—review and editing, J.Y. and Z.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Science Foundation of China under Grant 62071088; in part by the Project for Innovation Team of Guangdong University under Grant 2018KCXTD033; in part by Opening Project Guangdong Province Key Laboratory of information Security Technology under Grant 2020B1212060078; in part by Special Projects for Key Fields of the Education Department of Guangdong Province under Grant 2021ZDZX1083; and in part by Project for Zhongshan Science and Technology under Grant 2021B2062; in part by the Science and Technology Projects of Guangdong Province under Grant 2021A0101180005; and in part by the Construction Project of Professional Quality Engineering in Guangdong Province under Grant YLZY202201.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable to this article as no datasets were generated during the current study.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Kanso, A.; Ghebleh, M.; BouKhuzam, M. A Probabilistic Chaotic Image Encryption Scheme. *Mathematics* **2022**, *10*, 1910. [CrossRef]
2. Fang, J.S.; Tsai, J.S.H.; Yan, J.J.; Chiang, L.H.; Guo, S.M. Secure Data Transmission and Image Encryption Based on a Digital-Redesign Sliding Mode Chaos Synchronization. *Mathematics* **2022**, *10*, 518. [CrossRef]
3. Zhang, S.; Liu, L.; Xiang, H. A Novel Plain-Text Related Image Encryption Algorithm Based on LB Compound Chaotic Map. *Mathematics* **2021**, *9*, 2778. [CrossRef]
4. Xian, Y.; Wang, X. Fractal sorting matrix and its application on chaotic image encryption. *Inf. Sci.* **2021**, *547*, 1154–1169. [CrossRef]
5. Wei, H.; Cui, M.; Zhang, C.; Wu, T.; Wen, H.; Zhang, Z.; Chen, Y.; Qiu, K. Chaotic key generation and application in OFDM-PON using QAM constellation points. *Opt. Commun.* **2021**, *490*, 126911. [CrossRef]
6. Wu, T.; Zhang, C.; Chen, Y.; Cui, M.; Huang, H.; Zhang, Z.; Wen, H.; Zhao, X.; Qiu, K. Compressive sensing chaotic encryption algorithms for OFDM-PON data transmission. *Opt. Express* **2021**, *29*, 3669–3684. [CrossRef]
7. Hu, G.; Li, B. Coupling chaotic system based on unit transform and its applications in image encryption. *Signal Process.* **2021**, *178*, 107790. [CrossRef]
8. Chai, X.; Gan, Z.; Lu, Y.; Chen, Y.; Han, D. A novel image encryption algorithm based on the chaotic system and DNA computing. *Int. J. Mod. Phys. C* **2017**, *28*, 1750069. [CrossRef]
9. Wu, X.; Wang, K.; Wang, X.; Kan, H.; Kurths, J. Color image DNA encryption using NCA map-based CML and one-time keys. *Signal Process.* **2018**, *148*, 272–287. [CrossRef]
10. Zhang, C.; Yan, Y.; Wu, T.; Zhang, X.; Wen, G.; Qiu, K. Phase Masking and Time-Frequency Chaotic Encryption for OFDM-PON. *IEEE Photonics J.* **2018**, *10*, 7203009.

11. Wu, T.; Zhang, C.; Huang, H.; Zhang, Z.; Wei, H.; Wen, H.; Qiu, K. Security Improvement for OFDM-PON via DNA Extension Code and Chaotic Systems. *IEEE Access* **2020**, *8*, 75119–75126. [CrossRef]
12. Li, H.; Hua, Z.; Bao, H.; Zhu, L.; Chen, M.; Bao, B. Two-Dimensional Memristive Hyperchaotic Maps and Application in Secure Communication. *IEEE Trans. Ind. Electron.* **2021**, *68*, 9931–9940. [CrossRef]
13. Zang, H.; Zhao, X.; Wei, X. Construction and application of new high-order polynomial chaotic maps. *Nonlinear Dyn.* **2022**, *107*, 1247–1261. [CrossRef]
14. Hu, Y.; Yu, S.; Zhang, Z. On the Cryptanalysis of a Bit-Level Image Chaotic Encryption Algorithm. *Math. Probl. Eng.* **2020**, *2020*. [CrossRef]
15. Liu, S.; Li, C.; Hu, Q. Cryptanalyzing two image encryption algorithms based on a first-order time-delay system. *IEEE Multimed.* **2022**, *29*, 74–84. [CrossRef]
16. Ma, Y.; Li, C.; Ou, B. Cryptanalysis of an image block encryption algorithm based on chaotic maps. *J. Inf. Secur. Appl.* **2020**, *54*, 102566. [CrossRef]
17. Wen, H.; Yu, S. Cryptanalysis of an image encryption cryptosystem based on binary bit planes extraction and multiple chaotic maps. *Eur. Phys. J. Plus* **2019**, *134*, 337. [CrossRef]
18. Wen, H.; Yu, S.; Lv, J. Breaking an Image Encryption Algorithm Based on DNA Encoding and Spatiotemporal Chaos. *Entropy* **2019**, *21*, 246. [CrossRef]
19. Li, C.; Lin, D.; Lv, J. Cryptanalyzing an Image-Scrambling Encryption Algorithm of Pixel Bits. *IEEE Multimed.* **2017**, *24*, 64–71. [CrossRef]
20. Hua, Z.; Zhou, Y.; Bao, B. Two-Dimensional Sine Chaotification System with Hardware Implementation. *IEEE Trans. Ind. Inform.* **2020**, *16*, 887–897. [CrossRef]
21. Chai, X.; Fu, X.; Gan, Z.; Lu, Y.; Chen, Y. A color image cryptosystem based on dynamic DNA encryption and chaos. *Signal Process.* **2019**, *155*, 44–62. [CrossRef]
22. Wen, H.; Zhang, C.; Chen, P.; Chen, R.; Xu, J.; Liao, Y.; Liang, Z.; Shen, D.; Zhou, L.; Ke, J. A Quantum Chaotic Image Cryptosystem and Its Application in IoT Secure Communication. *IEEE Access* **2021**, *9*, 20481–20492. [CrossRef]
23. Wen, H.; Yu, S.; Lv, J. Encryption algorithm based on Hadoop and non-degenerate high-dimensional discrete hyperchaotic system. *Acta Phys. Sin.* **2017**, *66*, 14.
24. Wen, H.; Xu, J.; Liao, Y.; Chen, R.; Shen, D.; Wen, L.; Shi, Y.; Lin, Q.; Liang, Z.; Zhang, S. A Security-Enhanced Image Communication Scheme Using Cellular Neural Network. *Entropy* **2021**, *23*, 1000. [CrossRef]
25. Wen, H.; Zhang, C.; Huang, L.; Ke, J.; Xiong, D. Security Analysis of a Color Image Encryption Algorithm Using a Fractional-Order Chaos. *Entropy* **2021**, *23*, 258. [CrossRef]
26. Dou, Y.; Liu, X.; Fan, H.; Li, M. Cryptanalysis of a DNA and chaos based image encryption algorithm. *Optik* **2017**, *145*, 456–464. [CrossRef]
27. Murillo-Escobar, M.A.; Cruz-Hernández, C.; Abundiz-Pérez, F.; López-Gutiérrez, R.M.; Del Campo, O.A. A RGB image encryption algorithm based on total plain image characteristics and chaos. *Signal Process.* **2015**, *109*, 119–131. [CrossRef]
28. Adleman, L. Molecular computation of solutions to combinatorial problems. *Science* **1994**, *266*, 1021–1024. [CrossRef]
29. Dove, A. The long arm of DNA. *Nat. Biotechnol.* **1999**, *17*, 649–651. [CrossRef]
30. Zhang, Q.; Guo, L.; Wei, X. Image encryption using DNA addition combining with chaotic maps. *Math. Comput. Model.* **2010**, *52*, 2028–2035. [CrossRef]
31. Ozkaynak, F.; Yavuz, S. Analysis and improvement of a novel image fusion encryption algorithm based on DNA sequence operation and hyper-chaotic system. *Nonlinear Dyn.* **2014**, *78*, 1311–1320. [CrossRef]
32. Zhan, K.; Wei, D.; Shi, J.; Yu, J. Cross-utilizing hyperchaotic and DNA sequences for image encryption. *J. Electron. Imaging* **2017**, *26*, 013021. [CrossRef]
33. Liu, Y.; Zhang, J. A Multidimensional Chaotic Image Encryption Algorithm based on DNA Coding. *Multmedia Tools Appl.* **2020**, *79*, 29–30. [CrossRef]
34. Cui, M.; Chen, Y.; Zhang, C.; Liang, X.; Wu, T.; Liu, S.; Wen, H.; Qiu, K. Chaotic RNA and DNA for security OFDM-WDM-PON and dynamic key agreement. *Opt. Express* **2021**, *29*, 25552–25569. [CrossRef] [PubMed]
35. Janakiraman, S.; Thenmozhi, K.; Rayappan, J.B.B.; Amirtharajan, R. Lightweight chaotic image encryption algorithm for real-time embedded system: Implementation and analysis on 32-bit microcontroller. *Microprocess. Microsyst.* **2018**, *56*, 1–12. [CrossRef]
36. Wang, T.; Wang, M. Hyperchaotic image encryption algorithm based on bit-level permutation and DNA encoding. *Opt. Laser Technol.* **2020**, *132*, 106355. [CrossRef]
37. García-Guerrero, E.E.; Inzunza-González, E.; López-Bonilla, O.R.; Cárdenas-Valdez, J.R.; Tlelo-Cuautle, E. Randomness improvement of chaotic maps for image encryption in a wireless communication scheme using PIC-microcontroller via Zigbee channels. *Chaos Solitons Fractals* **2020**, *133*, 109646. [CrossRef]
38. Wang, X.; Li, Y. Chaotic image encryption algorithm based on hybrid multi-objective particle swarm optimization and DNA sequence. *Opt. Lasers Eng.* **2021**, *137*, 106393.1–106393.16. [CrossRef]
39. Jiang, X.; Xiao, Y.; Xie, Y.; Liu, B.; Ye, Y.; Song, T.; Chai, J.; Liu, Y. Exploiting optical chaos for double images encryption with compressive sensing and double random phase encoding. *Opt. Commun.* **2021**, *484*, 126683. [CrossRef]
40. Qobbi, Y.; Jarjar, A.; Essaid, M.; Benazzi, A. Image encryption algorithm based on genetic operations and chaotic DNA encoding. *Soft Comput.* **2022**, *26*, 5823–5832. [CrossRef]

41. Zang, H.; Tai, M.; Wei, X. Image Encryption Schemes Based on a Class of Uniformly Distributed Chaotic Systems. *Mathematics* **2022**, *10*, 1027. [CrossRef]
42. Chai, X.; Fu, X.; Gan, Z.; Zhang, Y.; Lu, Y.; Chen, Y. An efficient chaos-based image compression and encryption scheme using block compressive sensing and elementary cellular automata. *Neural Comput. Appl.* **2018**, *32*, 4961–4988. [CrossRef]
43. Li, C.; Lin, D.; Lü, J.; Hao, F. Cryptanalyzing an Image Encryption Algorithm Based on Autoblocking and Electrocardiography. *IEEE Multimed.* **2018**, *25*, 46–56. [CrossRef]
44. Li, S.; Gonzalo, A. Some basic cryptographic requirements for chaos-based cryptosystems. *Int. J. Bifurc. Chaos* **2006**, *16*, 2129–2151.
45. Murillo-Escobar, M.A.; Meranza-Castillón, M.O.; López-Gutiérrez, R.M.; Cruz-Hernández, C. Suggested Integral Analysis for Chaos-Based Image Cryptosystems. *Entropy* **2019**, *21*, 815. [CrossRef]
46. Song, C.; Qiao, Y. A Novel Image Encryption Algorithm Based on DNA Encoding and Spatiotemporal Chaos. *Entropy* **2015**, *17*, 6954–6968. [CrossRef]
47. Arslan, A.S.; Shahid, J. Novel image encryption cryptosystem based on binary bit planes extraction and multiple chaotic maps. *Eur. Phys. J. Plus* **2018**, *133*, 331.
48. Yin, Q.; Wang, C. A New Chaotic Image Encryption Scheme Using Breadth-First Search and Dynamic Diffusion. *Int. J. Bifurc. Chaos Appl. Sci. Eng.* **2018**, *28*, 1850047. [CrossRef]
49. Xu, L.; Li, Z.; Li, J.; Hua, W. A novel bit-level image encryption algorithm based on chaotic maps. *Opt. Lasers Eng.* **2016**, *78*, 17–25. [CrossRef]