entropy

Article

A Security-Enhanced Image Communication Scheme Using Cellular Neural Network

Heping Wen [1,2,3], Jiajun Xu [1], Yunlong Liao [1], Ruiting Chen [1], Danze Shen [1], Lifei Wen [1], Yulin Shi [1], Qin Lin [1], Zhonghao Liang [1], Sihang Zhang [1], Yuxuan Liu [1], Ailin Huo [1], Tong Li [1], Chang Cai [1] and Jiaqian Wen [1] and Chongfu Zhang [1,2,*]

[1] Zhongshan Institute, University of Electronic Science and Technology of China, Zhongshan 528402, China; wenheping@uestc.edu.cn (H.W.); xujiajun@stu.zsc.edu.cn (J.X.); teamoyan@stu.zsc.edu.cn (Y.L.); ruitingchen@stu.zsc.edu.cn (R.C.); shendanze@stu.zsc.edu.cn (D.S.); wenlifei@stu.zsc.edu.cn (L.W.); shiyulin@stu.zsc.edu.cn (Y.S.); liqin@stu.zsc.edu.cn (Q.L.); zhonghaoliang@stu.zsc.edu.cn (Z.L.); sihangzhang@stu.zsc.edu.cn (S.Z.); liuyuxuan@stu.zsc.edu.cn (Y.L.); ailinhuo@stu.zsc.edu.cn (A.H.); litong@stu.zsc.edu.cn (T.L.); caichang@stu.zsc.edu.cn (C.C.); jiaqianwen@stu.zsc.edu.cn (J.W.)

[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, Guangzhou 510006, China

[*] Correspondence: cfzhang@uestc.edu.cn

Abstract: In the current network and big data environment, the secure transmission of digital images is facing huge challenges. The use of some methodologies in artificial intelligence to enhance its security is extremely cutting-edge and also a development trend. To this end, this paper proposes a security-enhanced image communication scheme based on cellular neural network (CNN) under cryptanalysis. First, the complex characteristics of CNN are used to create pseudorandom sequences for image encryption. Then, a plain image is sequentially confused, permuted and diffused to get the cipher image by these CNN-based sequences. Based on cryptanalysis theory, a security-enhanced algorithm structure and relevant steps are detailed. Theoretical analysis and experimental results both demonstrate its safety performance. Moreover, the structure of image cipher can effectively resist various common attacks in cryptography. Therefore, the image communication scheme based on CNN proposed in this paper is a competitive security technology method.

Keywords: secure communication; image encryption; chaos; cellular neural network

Citation: Wen, H.; Xu, J.; Liao, Y.; Chen, R.; Shen, D.; Wen, L.; Shi, Y.; Lin, Q.; Liang, Z.; Zhang, S.; et al. A Security-Enhanced Image Communication Scheme Using Cellular Neural Network. *Entropy* **2021**, *23*, 1000. https://doi.org/10.3390/e23081000

Academic Editor: Amelia Carolina Sparavigna

Received: 23 June 2021
Accepted: 28 July 2021
Published: 31 July 2021

1. Introduction

With the rapid development of cloud computing, big data, blockchain and other emerging technologies, the privacy and sharing of messages provides convenience for people in their work and daily lives [1–4]. However, the convenience also threatens the security of cyberspace [5–8]. In particular, as a significant transmission medium, digital images may include a lot of personal privacy, confidential information and other important data, so their privacy protection gets more attention [9–12]. Encryption technology is a common means to assure the security of digital images, and has been widely used in various fields of digital image security [13–17]. Currently, there exist many mature block encryption schemes that are widely used in text encryption and these schemes have brilliant effects [18,19]. Nevertheless, due to the uniqueness of the image, such as being two-dimensional, redundancy and a strong correlation of two adjacent pixels, traditional text encryption faces severe challenges [20–22]. Moreover, the problem of real-time transmission should be considered in image encryption to improve the communication performance [9,23,24]. Therefore, it is quite necessary to study the new technologies and methods of image encryption.

In current international studies, digital image encryption is a research hotspot [25–27]. Various mechanisms and methods are introduced to enhance the security of algorithms [28,29].

In 2015, the authors of [16] proposed a multibiometric template protection scheme based on fuzzy commitment and a chaos-based system, as well as a security analysis method of unimodal biometrics leakage. The chaos-based system is used to encrypt the dual iris feature vectors. The experimental results show that the security of BCH ECC (1,023,123,170) based on multibiometrics template is improved from 80.53 bits to 167.80 bits. In 2017, the authors of [30] designed a special image encryption scheme based on the second-order Henon mapping hyperchaos and the fifth-order CNN. Experimental results show that the scheme features high security and is suitable to spread in the network. At the same time, in [31] a new image encryption method was proposed, based on the biological DNA sequences operation and the third-order CNN. The method could effectively enhance the plaintext sensitivity and features large key space and high security. In 2019, the authors of [17] proposed a new privacy protection encryption mechanism for medical systems based on the Internet of Things. Experimental results show that the encryption mechanism is robust and effective to protect the privacy of patients. In 2020, Zhang and Zhang [32] used the Chen chaos-based system and two-dimensional logistic mapping to propose a multi-image encryption system based on bitplane and chaos. The experiment also proved its high efficiency. At the same time, in [15] a new and effective color image cryptosystem was proposed. The experimental results show that the cryptosystem has high security efficiency and can be effectively applied to the IoHT framework of secure medical image transmission. In summary, more and more theories and technological achievements have been made in digital image encryption. However, in current studies, most digital images are regarded as a two-dimensional matrix to encrypt, meaning that only the spatial domain is processed [6,33–35]. However, two defects were exposed: (1) Some encryption algorithms have security flaws and are not associated with plaintext, so it is difficult for them to resist chosen-plaintext attack (CPA); (2) The cost of attacking the encryption algorithm is relatively low because chaos-based systems are relatively simple.

Aimed at solving the existing problems, we put forward a digital image encryption algorithm based on CNN in this paper. On the one hand, a CNN chaos-based system is selected to generate a chaos-based key sequence. The CNN chaos-based system has more complex behavioral characteristics, so it has better security performance than other encryption systems. On the other hand, the scheme adopts the security mechanism of generating a chaos-based key sequence by plaintext correlation. Therefore, compared with other encryption schemes based on a CNN chaos-based system, it effectively enhances the ability to resist CPA. Theoretical analysis and experimental results show that the proposed algorithm can effectively enhance the confusion, diffusion and avalanche effect of encryption. Therefore, the image encryption algorithm based on CNN is reliable.

2. Correlation Theory

The idea of a cellular neural network (CNN) was conceived by Chua and Yang in 1988 [34]. The basic units of CNN are called cells, and each cell is a nonlinear first-order circuit which is composed of a linear resistor, a linear capacitor and a voltage-controlled current source [36,37].

In order to make the mathematical model of CNN more comprehensible, a simplified CNN cell model is adopted:

$$\frac{dx_j}{dt} = -x_j + A_j p_j + G_o + G_s + I_j \tag{1}$$

where j is used as a cell marker, x_j represents the state variable, A_j represents a constant number, I_j represents the threshold value, G_s and G_o separately represent the linear combination of the state variables of the cell and the output value of the connecting cell, and p_j represents the output of the cell.

The fourth-order fully interconnected CNN equation can be defined as follows:

$$\begin{cases} \frac{dx_j}{dt} = -x_j + A_j p_j + \sum\limits_{k=1;k\neq j}^{4} A_{jk} p_j + \sum\limits_{k=1}^{4} S_{jk} x_k + I_j \\ p_j = 0.5|x_j + 1| - 0.5|x_j - 1| \end{cases} \tag{2}$$

where S represents a matrix of $j \times k$, A_j and I_j both represent a matrix of $j \times 1$, $A_{jk} = 0 (j \neq k, j = 1,2,3,4; k = 1,2,3,4)$ and it can be described by the equation of state in Equation (2) [38]:

$$\begin{cases} \dfrac{dx_1}{dt} = -x_3 - \varepsilon x_4 \\ \dfrac{dx_2}{dt} = 2x_2 + x_3 \\ \dfrac{dx_3}{dt} = 14x_1 - 14x_2 \\ \dfrac{dx_4}{dt} = 200p_4 + 100x_1 - 100x_4 \end{cases} \tag{3}$$

where ε is the control parameter of the CNN model, which can control the size and quantity of Lyapunov exponents, and the range of values for ε is 0 to 2. At this moment, the system is in a chaos-based state, and four aperiodic chaos-based sequences can be generated from it, which are very sensitive to the initial conditions $x_1(0), x_2(0), x_3(0)$ and $x_4(0)$. By calculating the Lyapunov exponents of Equation (3), it can be seen that the Lyapunov exponents of the four chaos-based sequences tend to 42.8487, 2.0230, -0.0230 and -49.0391, respectively, two of which are positive. Therefore, the CNN model is a hyperchaotic system, and the Lyapunov exponents are shown in Figure 1. When the initial values of $x_1(0)$, $x_2(0)$, $x_3(0)$ and $x_4(0)$ are 0.2, 0.2, 0.2 and 0.2, respectively, we use the fourth-order Runge–Kutta algorithm with the step size of h = 0.005 to get the two-dimensional chaos-based attractor, as shown in Figure 2a–d and the three-dimensional chaos-based attractor, as shown in Figure 2e–h.

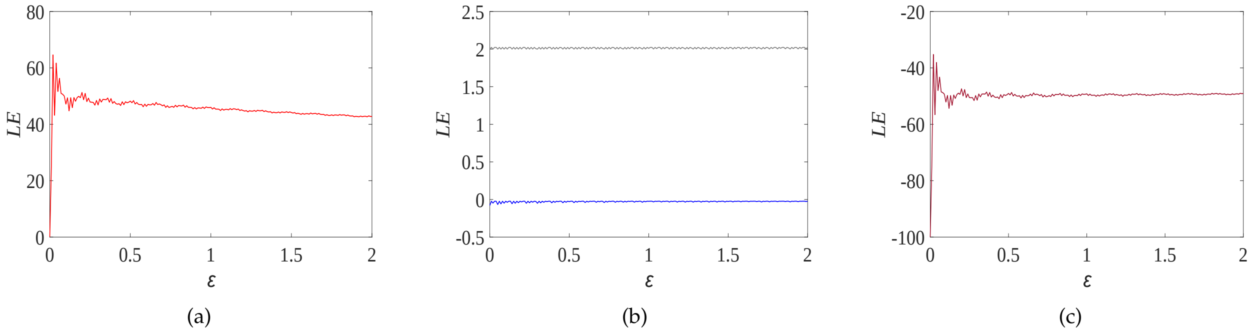


Figure 1. Lyapunov exponents spectrum. The exponents tend to 42.8487, 2.0230 and −0.0230, and −49.0391, as can be seen in (**a**–**c**), respectively.

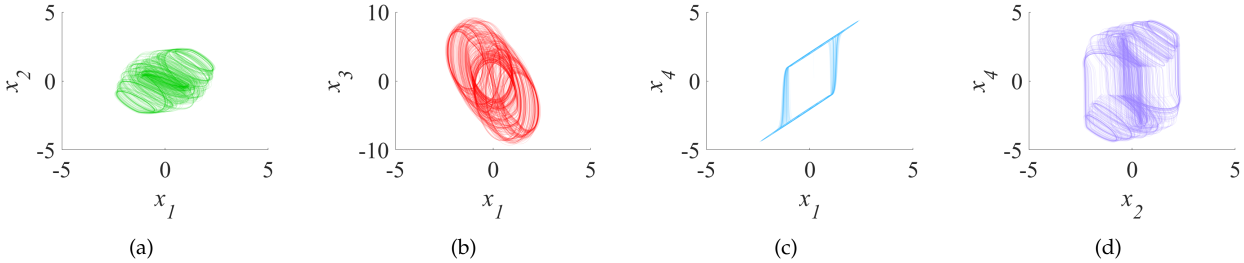


Figure 2. *Cont.*

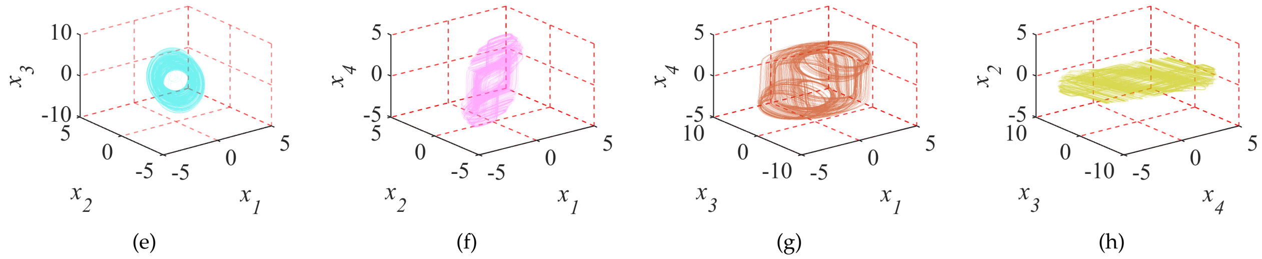


Figure 2. Chaos-based attractors generated by the fourth-order CNN: (**a**) x_1, x_2; (**b**) x_1, x_3; (**c**) x_1, x_4; (**d**) x_2, x_4; (**e**) x_1, x_2, x_3; (**f**) x_1, x_2, x_4; (**g**) x_1, x_3, x_4; (**h**) x_4, x_3, x_2.

3. The Proposed Encryption Algorithm

The encryption algorithm of chaos-based image usually adopts the classical structure “permutation–diffusion” [39,40]. However, due to the lack of security, a chaos-based image encryption algorithm based on a “confusion–permutation–diffusion” structure is proposed in this paper [35].

The encryption and decryption processes are shown in Figure 3. IEA-CNN represents the image encryption algorithm based on a cellular neural network, IDA-CNN represents the image decryption algorithm based on a cellular neural network. In order to enhance the ability to resist CPA, the image encryption system of this paper adopts the security mechanisms of chaos-based key sequences produced by plaintext association and ciphertext feedback diffusion encryption. The specific steps of the encryption algorithm are given as follows:

Step 1: *Preprocessing Sequences*

The secret key of the image encryption algorithm contains the Message-Digest Algorithm 5 (MD5) value of plain image, the initial value of the fourth-order CNN and the controlling parameters. The MD5 can be used to disturb the initial value key parameters of CNN chaos; so that the key sequence changes with different plain images, the specific treatment methods are calculated using the following formulas:

$$\begin{cases} x'_1(0) = x_1(0) + (m_1 \oplus m_2 \oplus m_3 \oplus m_4)/256 \\ x'_2(0) = x_2(0) + (m_5 \oplus m_6 \oplus m_7 \oplus m_8)/256 \\ x'_3(0) = x_3(0) + (m_9 \oplus m_{10} \oplus m_{11} \oplus m_{12})/256 \\ x'_4(0) = x_4(0) + (m_{13} \oplus m_{14} \oplus m_{15} \oplus m_{16})/256 \end{cases} \tag{4}$$

where $\oplus$ is bitwise XOR operation, $x_1(0), x_2(0), x_3(0)$ and $x_4(0)$ are the initial values of the fourth-order CNN key parameters; $x_1{}'(0), x_2{}'(0), x_3{}'(0)$ and $x_4{}'(0)$ are the initial values updated after the disturbance from MD5. Obviously, the new initial values will change with the different plain images. Then, a preprocessing operation is adopted for the chaos-based sequences. The generating methods of obfuscated sequences are shown as follows:

$$\begin{cases} real\_X = [x_1; x_2; x_3; x_4] \\ K_c{}' = floor(\bmod(real\_X \times 10^{10}, 256)) \\ K_c = reshape(K_c{}', H, W) \end{cases} \tag{5}$$

where *real_X* is composed of four sequences produced by the fourth-order CNN chaos-based system. The sequences diagram of four sequences generated by chaos-based mapping of the fourth-order CNN is shown in Figure 4. The size of K_c is equal to $H \times W$, H

and W are pixel rows and pixel columns of the plain images for image confusion. The generating method of permutation sequences is shown as follows:

$$\begin{cases} seq\_H = x_2(1, 1:H) \\ seq\_W = x_3(2, 1:8\times W) \\ [value_1, K_{pr}] = sort(seq\_H) \\ [value_2, K_{pc}] = sort(seq\_W) \end{cases} \tag{6}$$

where *sort* is the sorting function of array elements; x_2 represents a two-dimensional sequence of *real_X*; x_3 represents the three-dimensional sequence of *real_X*; *seq_H* represents the chaos-based sequence of length H extracted from x_2; *real_W* represents the chaos-based sequence of length $8 \times W$ extracted from x_3; K_{pr} means that the pixel row is generated by the sorting function and the length is H; K_{pc} means that the pixel column is generated by the sorting function and the length is $8 \times W$; $value_1$ and $value_2$ are the sorted chaos-based sequence values.

The generating method of diffusion sequences is shown as follows:

$$\begin{cases} K_d = mod(floor([x_1, x_3, x_2, x_4] \times 10^5), 256) \\ K_d{}' = mod(floor([x_3, x_4, x_1, x_2] \times 10^5), 256) \end{cases} \tag{7}$$

where the lengths of K_d and $K_d{}'$ are $H \times W$, and the key sequences of K_d and $K_d{}'$ are used for diffusion.

Step 2: *Confusion*

The key sequence K_c is used to obfuscate the plain image P. The image can be visualized and hidden to get the obfuscated image I_1, the method is shown as follows:

$$I_1(i) = K_c(i) \oplus P(i), i = (1, 2, \cdots, H \times W) \tag{8}$$

Step 3: *Permutation*

The key sequences $K_{pr}(i)$ and $K_{pc}(j)$ are used to replace the pixels in I_1 to get I_3, the method is shown as follows:

$$\begin{cases} I_2 = \mathrm{swap}(I_1(:, K_{pc}(i)), I_1(:, i)) \\ I_3 = \mathrm{swap}(I_2(K_{pr}(j), :), I_2(j, :)) \end{cases} \tag{9}$$

where *swap* function is used to swap the values of two pixels. The number of bit level rows is equal to the number of pixel level rows, and the number of bit level columns is equal to 8 times the number of pixel level columns, thus, $i = 1, 2, \cdots, H$ and $j = 1, 2, \cdots, 8 \times W$. I_2 and I_3 are the images after double bit column transform and row transform permutation, respectively.

Step 4: *Diffusion*

All the ciphertext pixels in I_3 are diffused dynamically. K_d and K'_d are used for the image diffusion operation to generate the final ciphertext image C.

The first ciphertext pixel $C(1)$ is generated, and the diffusion encryption equation is shown as follows:

$$\begin{cases} C(1) = I_3(1) \oplus K_d(1) \oplus (sum(1) \dot{+} K_d{}'(1)) \\ sum(1) = \sum_{i=1}^{L} I_3(i) \end{cases} \tag{10}$$

where the operator $\dot{+}$ can be defined as $a \dot{+} b \overset{\Delta}{=} \mathrm{mod}(a + b, 256)$, $I_3(1)$ is the first pixel of the permutation image I_3, $K_d(1)$ and $K_d{}'(1)$ are the first element of the diffusion encryption sequences, and $sum(1)$ represents the sum of all pixels of the permutation image I_3.

Then ciphertext pixel $C(i)$ is produced and its diffusion formula is shown as follows:

$$\begin{cases} C(i) = I_3(i) \oplus (C(i-1) \dotplus K_d(i)) \oplus (sum(i) \dotplus K_d{}'(i)) \\ sum(i) = sum(i-1) - I_3(i) \end{cases} \tag{11}$$

where $i = 2, 3, \ldots, L$ and the i represents the ith pixel of the permutation image I_3. $C(i-1)$ is the $(i-1)$th ciphertext pixel. $sum(i)$ is the sum of the $(L-i+1)$ pixels of the permutation image I_3. According to Equation (11), starting from the second ciphertext pixel $C(2)$, the cipher image C is generated by computing iteratively $C(i)$, i in $\{1, 2, \cdots, L\}$, until the Lth ciphertext $C(L)$ is generated.

Decryption is the inverse process of encryption, whose process is first confusion, then permutation, and finally diffusion. While the decryption process is to first reverse diffuse the encrypted image, then reverse permutate the reverse diffuse image, and finally reverse confuse the reverse permutation image to get the decrypted image. When the decryption key and the encryption key are matched, the image can be restored correctly. However, when the decryption key is not equal to the encryption key, even if there is a small error, the correct image cannot be decrypted.

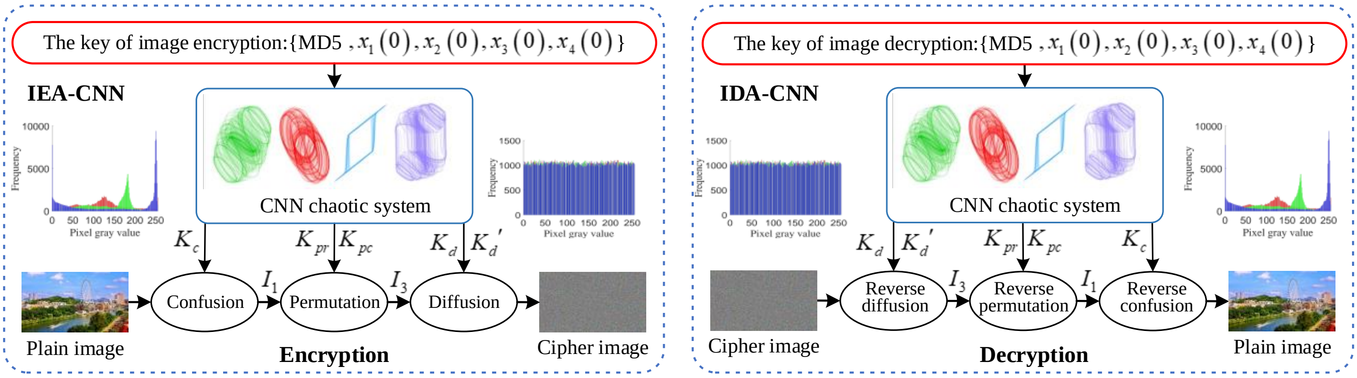


Figure 3. Principle and mechanism of image encryption and decryption.

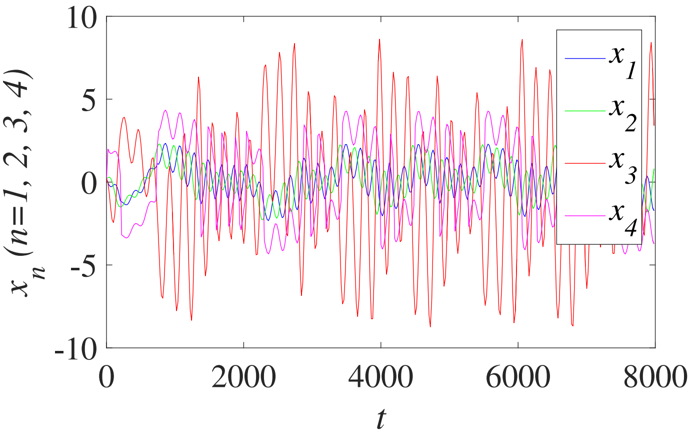


Figure 4. Sequence diagram of the fourth-order CNN.

4. Experimental Verification and Discussion

In the analysis of the experimental results, we use MATLAB 2020b to simulate and validate the proposed image encryption system which is executed on a PC with Windows 10 64 bit operating system, Intel (R) Core (TM) i7-8250 CPU @ 1.60 GHz 1.80 GHz processor and 8 GB memory. In order to prove the effectiveness and practicability of the proposed image encryption scheme, we selected the images from "USC-SIPI Image Database" and "Ground Truth Database" as the test images [41,42].

4.1. Key Space Analysis

In the encryption system, the range of valid value of key can be expressed by key space. The image encryption algorithm designed in this paper uses a fourth-order CNN system and the secret key parameters involved are the initial values of the fourth-order CNN chaos-based system $x_1(0), x_2(0), x_3(0), x_4(0)$. Because the computer precision used in experimental simulation is 10^{-15}, the size of this part of encryption system key space is $(10^{15})^4 = 10^{60} \approx 2^{199}$. Considering that MD5 of 128 bits can also be used as part of the secret key, the total secret key space 2^{327} and the encryption system can resist the exhaustive attack effectively [43,44].

4.2. Nist 800-22 Test

The NIST 800-22 test is an internationally recognized random number test. It consists of 16 different tests. As long as the 16 test results are greater than or equal to 0.001, the random array can be considered to be qualified. In this test, we divide the generated 3,000,000 bits of byte stream data into 10 segments of 300,000 bits. The K_c, K_{pr}, K_{pc}, K_d and K_d' sequences needed in encryption passed the test successfully, and the test results of the K_d' sequence are shown in Table 1. The experimental results show that the random numbers generated by our algorithm fully conform to the international standard, and have strong randomness.

Table 1. NIST-800-22 test results.

Statistical Tests	*p*-Values										Result
	Seq1	Seq2	Seq3	Seq4	Seq5	Seq6	Seq7	Seq8	Seq9	Seq10	
ApproximateEntropy Text	0.8094	0.1941	0.0781	0.3518	0.4390	0.3812	0.4203	0.1690	0.1884	0.0589	Successful
BlockFrequency Text	0.9347	0.2822	0.9547	0.0925	0.6961	0.4518	0.1352	0.4160	0.3816	0.1934	Successful
CumulativeSums Text-1	0.7034	0.9290	0.7701	0.4770	0.0354	0.6270	0.4488	0.2083	0.4378	0.5493	Successful
CumulativeSums Text-2	0.8561	0.9968	0.8754	0.7377	0.0426	0.2912	0.2621	0.1019	0.3783	0.1853	Successful
FFT Text	0.9732	0.9066	0.4508	0.2911	0.4921	0.1912	0.8145	0.4508	0.0226	0.1359	Successful
Frequency Text	0.8666	0.8408	0.9040	0.4541	0.0235	0.6507	0.7674	0.1743	0.9330	0.5541	Successful
LinearComplexity Text	0.2833	0.8136	0.5262	0.2415	0.6749	0.4776	0.9849	0.2676	0.8014	0.3305	Successful
LongestRun Text	0.3615	0.2823	0.5065	0.4150	0.7894	0.7386	0.0683	0.1561	0.5800	0.2138	Successful
OverlappingTemplate Text	0.2713	0.8537	0.8457	0.6464	0.2555	0.1803	0.4144	0.9091	0.7819	0.7349	Successful
Rank Text	0.6985	0.1675	0.6198	0.2927	0.5757	0.3860	0.3147	0.8761	0.3737	0.2093	Successful
Runs Text	0.6066	0.6691	0.6771	0.2721	0.3432	0.1041	0.5789	0.7783	0.6718	0.6011	Successful
Serial Text-1	0.0096	0.8837	0.0110	0.5441	0.1669	0.0331	0.8454	0.1955	0.7045	0.6886	Successful
Serial Text-2	0.1784	0.6697	0.2170	0.5832	0.0293	0.3877	0.9621	0.4920	0.7287	0.5582	Successful

4.3. Histogram Analysis

There are three channels—R, G and B—in color images; the abscissa of the histogram containing these three channels reflects the statistical characteristics of the distribution of every pixel [45,46]. Different plain images and cipher images, as well as their relevant histograms, are shown in Figure 5. The experimental results show that the pixel values of the R, G and B channels of color cipher image are almost uniformly distributed, so the influence of statistical analysis is greatly eliminated [47,48].

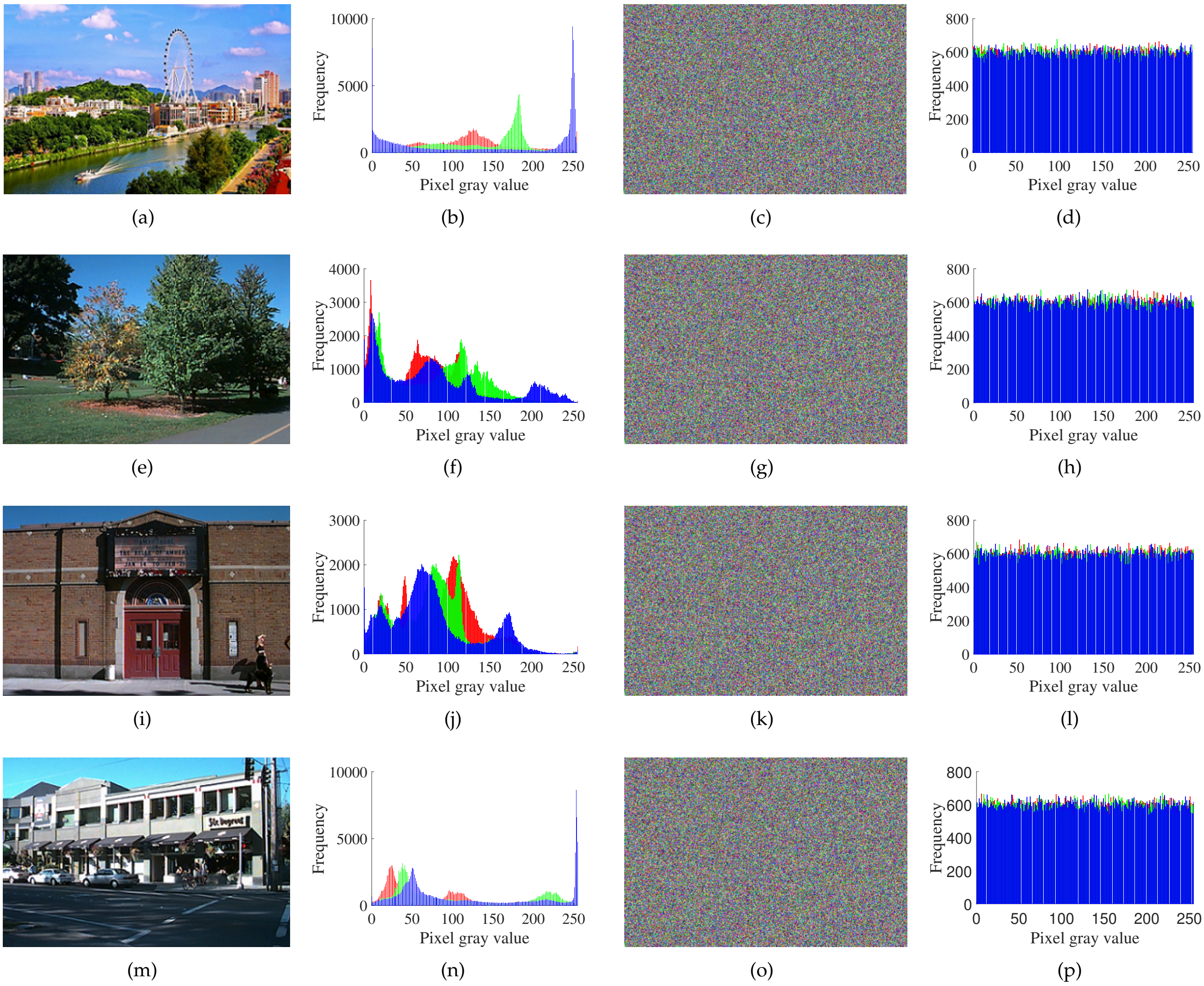


Figure 5. The histograms of images before and after encryption: (**a**) plain image of "Zhong shan"; (**b**) histogram of the plain image of "Zhong shan"; (**c**) cipher image of "Zhong shan"; (**d**) histogram of the cipher image of "Zhong shan"; (**e**) plain image of "Greenlake10"; (**f**) histogram of the plain image of "Greenlake10"; (**g**) cipher image of "Greenlake10"; (**h**) histogram of the cipher image of "Greenlake10"; (**i**) plain image of "Greenlake13"; (**j**) histogram of the plain image of "Greenlake13"; (**k**) cipher image of "Greenlake13"; (**l**) histogram of the cipher image of "Greenlake13"; (**m**) plain image of "Greenlake47"; (**n**) histogram of the plain image of "Greenlake47"; (**o**) cipher image of "Greenlake47"; (**p**) histogram of cipher image of "Greenlake47".

4.4. Correlation Analysis

For the plain image, the correlation between adjacent pixels is strong [49,50]. Gray value of a pixel tends to be close to the gray values of its adjacent pixels. Therefore, the attacker can speculate about the gray value of a pixel from the gray value of its adjacent pixels [51,52]. An encryption system with good performance should satisfy the requirement that adjacent pixels of cipher image have low correlation coefficients to each other in order to resist the statistical attack. Correlation coefficients are commonly used to measure the correlation of two pixels and the calculations of it are defined as [53,54]:

$$\begin{cases} E(x) = \frac{1}{N}\sum\limits_{i=1}^{N} x_i \\ D(x) = \frac{1}{N}\sum\limits_{i=1}^{N} (x_i - E(x))^2 \\ \operatorname{cov}(x,y) = \frac{1}{N}\sum\limits_{i=1}^{N} (x_i - E(x))(y_i - E(y)) \\ \gamma_{xy} = \frac{\operatorname{cov}(x,y)}{\sqrt{D(x)}\times\sqrt{D(y)}} \end{cases} \tag{12}$$

where the gray value of every pixel is represented by x and y, while $E(x)$ represents the mean value, $D(x)$ represents the variance, $cov(x, y)$ represents the covariance and γ_{xy} represents the correlation coefficients.

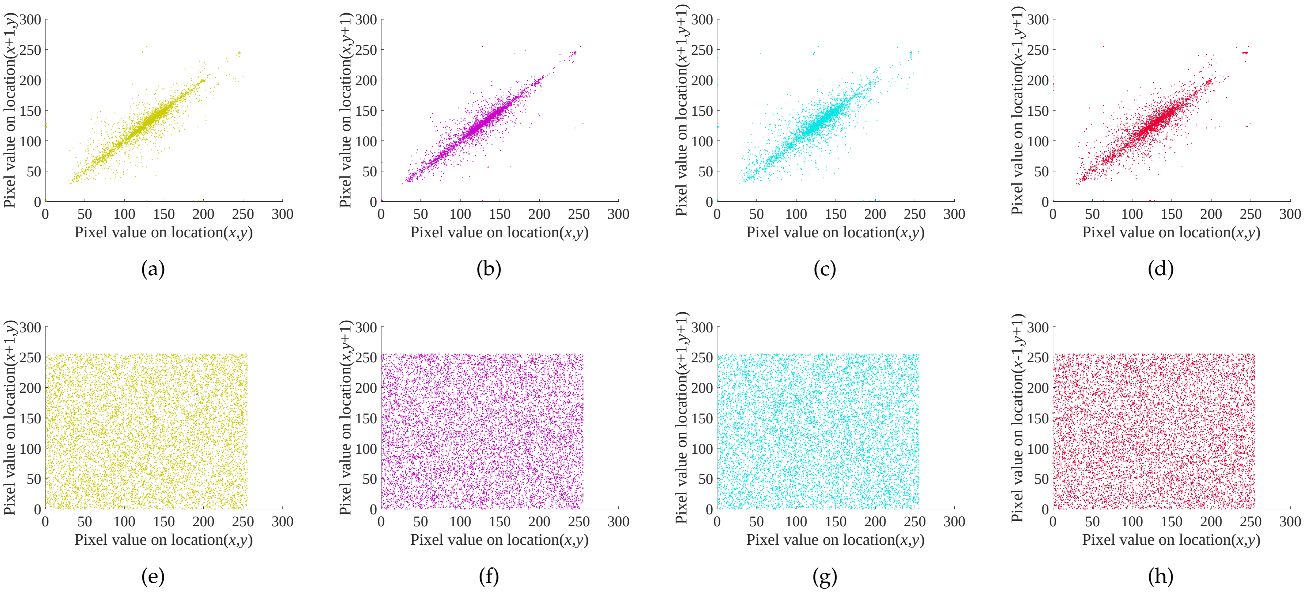
The correlation coefficients before and after encryption of the selected image are shown in Table 2 where "Anti-Diag". represents the correlation coefficient in the anti-diagonal direction. Figure 6 shows the correlation of plain image and cipher image in horizontal, vertical, diagonal and anti-diagonal directions. It can be seen that there is no obvious correlation between adjacent pixels of a cipher image. Therefore, the cipher images encrypted by the algorithm designed in this paper have high security and can resist the statistical analysis [55].

Table 2. Correlation coefficients of two adjacent pixels.

Pictures	Plain Image				Cipher Image			
	Vert.	Horiz.	Diag.	Anti-Diag.	Vert.	Horiz.	Diag.	Anti-Diag.
7.1.02.tiff	0.9480	0.9429	0.9113	0.9456	−0.0021	0.0303	0.0087	−0.0002
7.1.09.tiff	0.9309	0.9654	0.9208	0.9207	−0.0083	−0.0257	−0.0354	−0.0225
5.1.12.tiff	0.9709	0.9608	0.9429	0.9403	−0.0256	−0.0035	0.0040	−0.0157
5.2.10.tiff	0.9415	0.9364	0.9032	0.9015	0.0032	0.0163	−0.0069	−0.0107

4.5. Sensitivity Analysis

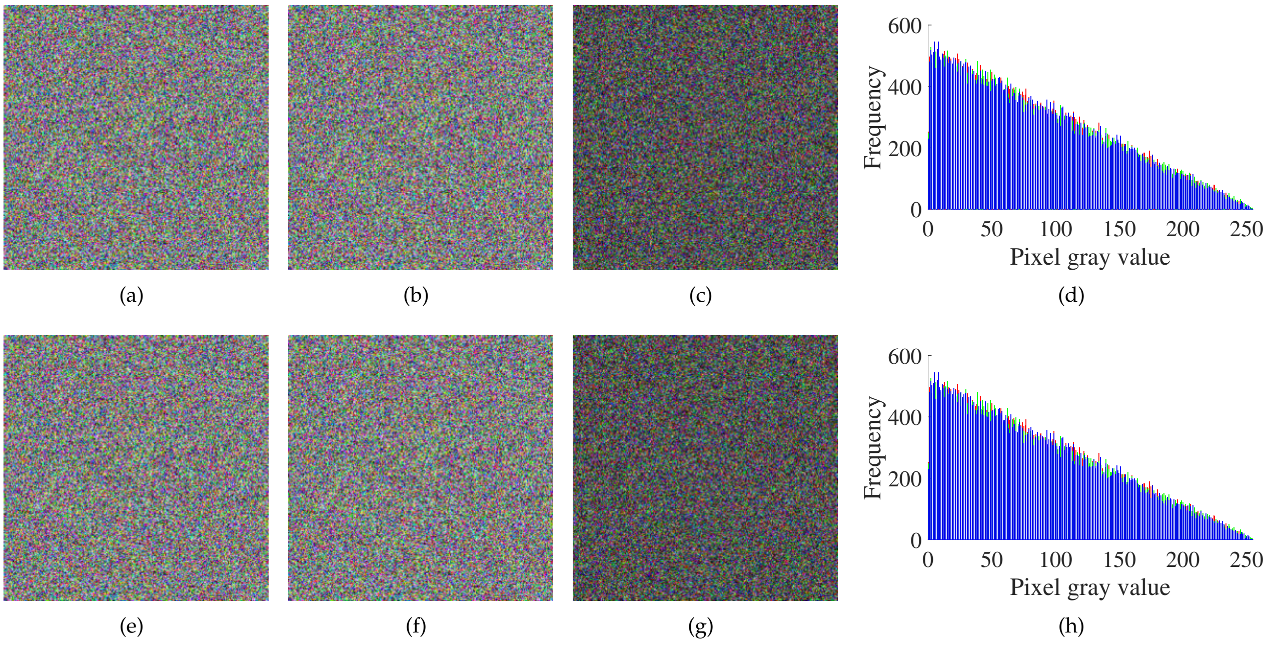
Key sensitivity is an essential indicator of the security of the encryption system. It represents the difference in the decryption results when the same cipher image is decrypted with slightly different keys. For the sake of detecting the susceptibility of the scheme to the key, the first three sequences generated by the initial key are superimposed and combined into a color map, and the minimum precision of $x_1(0)$ is 10^{-15}. The initial key $x_1(0)$ is perturbed with the minimum precision to generate four new sequences, and the first three new sequences are superimposed and combined into a new color map. The two color images are differentiated to get the difference image and the histogram corresponding to the difference image. The initial key $x_2(0)$ is processed in the same way, as shown in Figure 7. By adding 10^{-3} to the initial key $x_1(0)$, four sequences are obtained through cellular neural chaos, and these four sequences are compared with the four sequences generated by no change of $x_1(0)$, as shown in Figure 8. It can be seen from Figures 7 and 8 that the encryption system designed in this paper has high security and strong sensitivity to keys, which increases the difficulty for attackers to decipher the cipher image.



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

(e) (f) (g) (h)

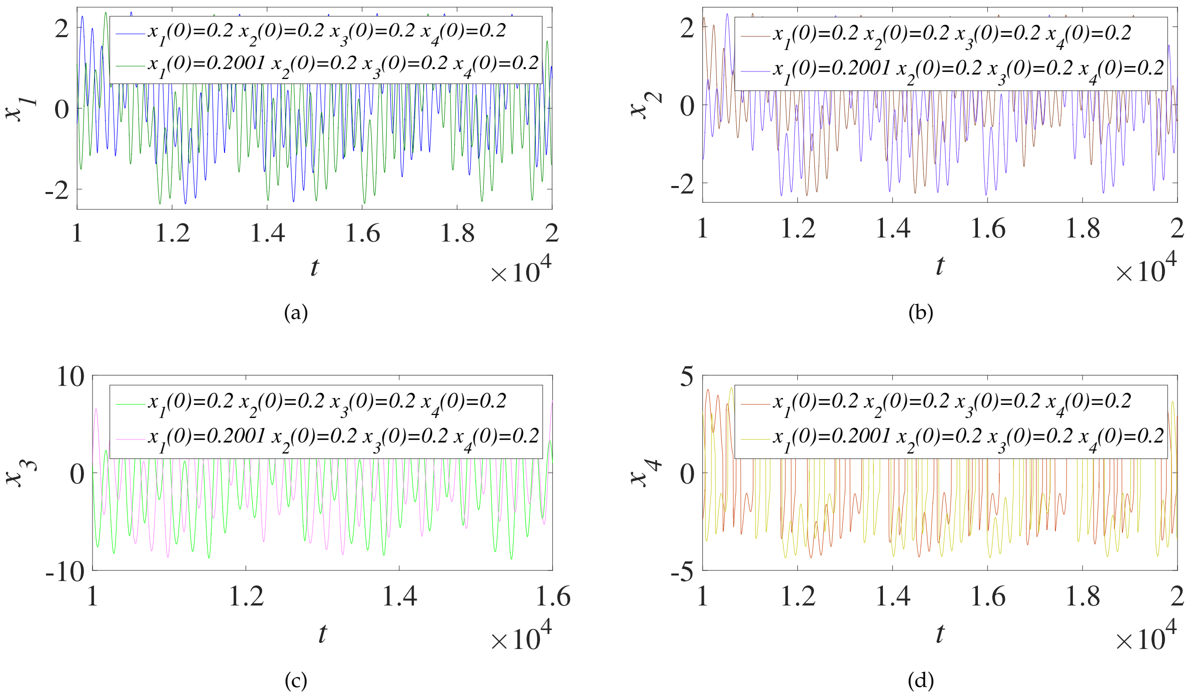
Figure 6. Correlation coefficients distribution map of plain image and cipher image of "7.1.02.tiff": (**a**) "7.1.02.tiff" plain image horizontal correlation; (**b**) "7.1.02.tiff" plain image is vertical correlation; (**c**) "7.1.02.tiff" plain image diagonal correlation; (**d**) "7.1.02.tiff" plain image against angular direction correlation; (**e**) "7.1.02.tiff" cipher image horizontal correlation; (**f**) "7.1.02.tiff" cipher image vertical correlation; (**g**) "7.1.02.tiff" cipher image diagonal correlation; (**h**) "7.1.02.tiff" cipher image inverse diagonal correlation.



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

(e) (f) (g) (h)

Figure 7. The key sensitivity test: (**a**) $x_1(0), x_2(0), x_3(0), x_4(0)$; (**b**) $x_1(0) + 10^{-15}, x_2(0), x_3(0), x_4(0)$; (**c**) Difference image after key perturbation; (**d**) Difference histogram after key perturbation; (**e**) $x_1(0), x_2(0), x_3(0), x_4(0)$; (**f**) $x_1(0), x_2(0) + 10^{-15}, x_3(0), x_4(0)$; (**g**) Difference image after key perturbation; (**h**) Difference histogram after key perturbation.



(a)

(b)

(c)

(d)

Figure 8. Comparison of four sequences (**a**–**d**) before and after key $x_1(0)$ perturbation.

Plaintext sensitivity is also one of the important indexes of encryption system security, which indicates the ability of encryption system to resist the differential attack. A secure encryption system should be highly sensitive to plain image. The Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) can be used to represent the difference between two plain images with one pixel difference. The calculation formula is [56]:

$$\begin{cases} NPCR = \frac{1}{H \times W} \times \sum_{i=1}^{H} \sum_{j=1}^{W} D(i,j) \times 100\% \\ UACI = \frac{1}{H \times W} \times \sum_{i=1}^{H} \sum_{j=1}^{W} \frac{|v_1(i,j) - v_2(i,j)|}{255} \times 100\% \end{cases} \tag{13}$$

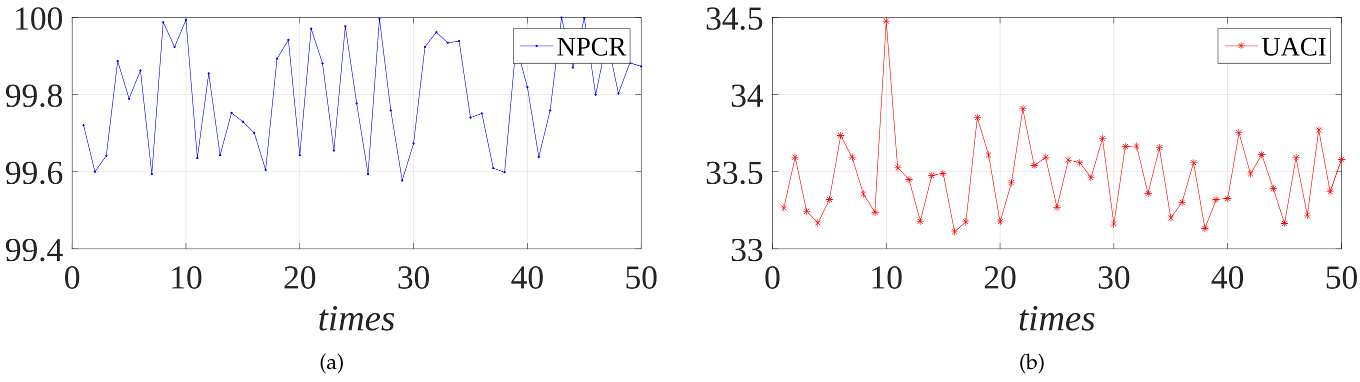
where $D(i,j) = \begin{cases} 0, v_1(i,j) = v_2(i,j) \\ 1, v_1(i,j) \neq v_2(i,j) \end{cases}$. $v_1(i,j)$ and $v_2(i,j)$ denote the pixel values at positions v_1 and v_2. For a digital image with a gray level of 256, 99.6094% and 33.4635% are ideal values of the NPCR and UACI, respectively.

Firstly, select a pixel from the "Lena" gray image randomly so that we can obtain a new image by changing its pixel value. Then, the two gray images which differ by only one pixel are each encrypted to obtain two ciphertext images. Finally, the NPCR and UACI values of the two encrypted images are obtained and the above operations will be repeated 50 times to obtain 50 groups of NPCR and UACI values. The NPCR and UACI average values of the gray images are shown in Table 3.

Table 3. NPCR and UACI.

Pictures	NPCR (99.6094%)	UACI (33.4635%)
1.2.04.tiff	99.6093%	33.5974%
1.2.07.tiff	99.6078%	33.5580%
1.2.08.tiff	99.6154%	33.5209%
5.1.11.tiff	99.5544%	33.4018%

The NPCR and UACI values obtained each time are shown in Figure 9. The NPCR and UACI average values are very close to the theoretical value. Therefore, the encryption system designed in this paper is extremely sensitive to both plain images and keys. The encryption algorithm designed in this study is safer and can resist the differential attack.



(a)

(b)

Figure 9. NPCR (**a**) and UACI (**b**).

4.6. Information Entropy Analysis

The degree of the randomness of the system can be expressed by information entropy. The information entropy of the image is positively correlated with the encryption effect. The larger the information entropy is, the better effect the encryption will have. The formula of information entropy is defined as [57]:

$$H(n) = -\sum_{i=0}^{G-1} -1P(n_i)\log_2 P(n_i) \tag{14}$$

where G represents the number of gray level values of the image and $P(n_i)$ the frequency of pixels with gray value i. The range of gray value of an image with a gray level of 256 is $[0, 255]$, and 8 is its ideal information entropy. When the value of information entropy is closer to 8, the image encryption has better effect [58].

Table 4 shows the information entropy before and after image encryption. The information entropy of the cipher image is very close to the theoretical value of information entropy. It is proven that the pixel value distribution of the cipher image is highly random and the encryption effect is better. Therefore, the algorithm can effectively resist the information entropy attack [33].

Table 4. Information entropy of the plain image and cipher image.

Pictures	Plain Image	Cipher Image
7.1.02.tiff	4.0045	7.9993
5.1.11.tiff	6.4523	7.9970
5.1.12.tiff	6.7057	7.9972
5.2.10.tiff	5.7056	7.9992

4.7. Psnr and Ssim

Peak Signal-to-Noise Ratio (PSNR) and Structural SIMilarity (SSIM) are often used to reflect the encryption quality. PSNR is essentially the same as the Mean Square Error (MSE) and can be obtained by MSE. The calculation formula is [59]:

$$\begin{cases} MSE = \frac{1}{H\times W}\sum\limits_{i=1}^{H}\sum\limits_{j=1}^{W}(P(i,j)-C(i,j))^2 \\ PSNR = 10\times\log_{10}\left(\frac{Q^2}{MSE}\right) \end{cases} \tag{15}$$

where the height and width of the image are represented by H and W, respectively, the pixel level of the image is represented by Q, the plain image pixels are represented by $P(i,j)$, and the cipher image pixels are represented by $C(i,j)$. SSIM is defined as [59]:

$$SSIM(p,c) = \frac{\left(2\mu_p\mu_c+(0.01L)^2\right)\left(2\sigma_{pc}+(0.03L)^2\right)}{\left(u_p^2+u_c^2+(0.01L)^2\right)\left(\sigma_p^2+\sigma_c^2+(0.03L)^2\right)} \tag{16}$$

where the average values of the plain image P and the cipher image C are denoted by u_p and u_c, respectively. 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 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.

The range of SSIM is from -1 to 1. When the two images are the same, SSIM is 1. The smaller the PSNR and SSIM are, the better the encryption quality is. Tables 5 and 6 show the encryption quality of the proposed scheme and the classic encryption schemes in recent years.

Table 5. PSNR of cipher image with different algorithms.

Pictures	This Paper	Ref. [1]	Ref. [60]	Ref. [28]
7.1.02.tiff	8.9518	9.1033	8.9731	8.9801
5.2.10.tiff	8.7620	8.7684	8.7660	8.7621
5.1.13.tiff	4.9032	4.9585	4.9168	4.9141
5.2.08.tiff	9.6225	9.6389	9.6378	9.6198

Table 6. SSIM of cipher image based on different algorithms.

Pictures	This Paper	Ref. [1]	Ref. [60]	Ref. [28]
7.1.02.tiff	0.0102	0.0108	0.0103	0.0109
5.1.11.tiff	0.0101	0.0099	0.0101	0.0109
5.2.10.tiff	0.0087	0.0098	0.0100	0.0091
5.1.13.tiff	0.0037	0.0057	0.0085	0.0067

The experimental results show that the PSNR and SSIM values obtained by the proposed algorithm are lower than those of other proposed approaches. Therefore, this encryption scheme has certain advantages, and the image encryption quality is high.

4.8. Robust Noise Analysis

Robustness means that the system still has certain performance under interference or at random. Image robustness refers to the fact that the image still has a certain degree of fidelity after undergoing various signal processing or attacks. The image can still be recognized, with low distortion. Add 20% salt-and-pepper noise and 80 × 80 occlusion noise to the cipher image "Figure 5a". The experimental results are shown in the figure below [34,60,61].

It can be seen from Figure 10 that the decrypted images can still be easily identified with high fidelity after noise is added to the cipher image, which indicates the robustness of the image encryption system that can resist noise attacks.

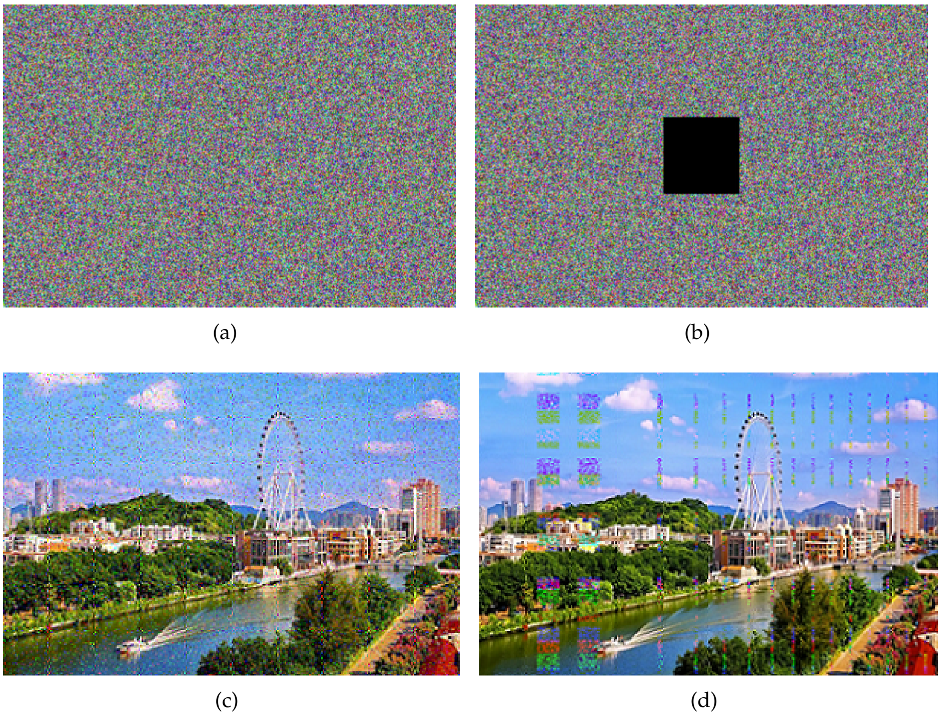


Figure 10. (**a**) Salt-and-pepper noise cipher image; (**b**) Occlusion noise cipher image; (**c**) Decryption of cipher image with salt-and-pepper noise; (**d**) Decryption of cipher image with occlusion noise.

5. Conclusions

This paper proposes a security-enhanced image communication scheme based on CNN under the cryptanalysis. First, the complex characteristics of CNN are used to generate some sequences. Then, a plain image and these CNN-based sequences are confused, permuted and diffused to get the cipher image. Utilizing the complex dynamics of CNN can effectively enhance the confusion, diffusion and avalanche of encryption. Theoretical analysis and experimental results both demonstrate its safety performance. From the perspective of cryptanalysis, the structure of an image cipher can effectively resist various common attacks. Therefore, the image communication scheme based on CNN proposed in this paper is a competitive security technology method.

Author Contributions: Methodology, H.W.; Project administration, H.W. and C.Z.; Software, J.X., R.C. and D.S.; Supervision, C.Z.; Validation, J.X., Y.L. (Yunlong Liao), R.C., L.W., Y.S., Q.L., Z.L., S.Z., Y.L. (Yuxuan Liu), A.H., T.L., C.C. and J.W.; Writing—original draft, J.X.; Writing—review & editing, H.W. 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 and Grant 61571092; in part by the Project for Innovation Team of Guangdong University under Grant 2018KCXTD033; in part by the Project for Zhongshan Social Public Welfare Science and Technology under Grant 2019B2007; in part by the Science and Technology Projects of Guangdong Province under Grant 2021A0101180005; in part by the Research Project for Talent of UESTC Zhongshan Institute under Grant 418YKQN07 and Grant 419YKQN23; and in part by the Opening Project Guangdong Province Key Laboratory of Information Security Technology under Grant 2020B1212060078.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chunyan, S.; Yulong, Q. A Novel Image Encryption Algorithm Based on DNA Encoding and Spatiotemporal Chaos. *Entropy* **2015**, *17*, 6954–6968.
2. Gopalakrishnan, T.; Ramakrishnan, S. Chaotic Image Encryption with Hash Keying as Key Generator. *IETE J. Res.* **2017**, *63*, 172–187. [CrossRef]
3. Li, A.; Belazi, A.; Kharbech, S.; Talha, M.; Xiang, W. Fourth Order MCA and Chaos-Based Image Encryption Scheme. *IEEE Access* **2019**, *7*, 66395–66409. [CrossRef]
4. Chai, X.; Chen, Y.; Broyde, L. A novel chaos-based image encryption algorithm using DNA sequence operations. *Opt. Lasers Eng.* **2017**, *88*, 197–213. [CrossRef]
5. Kalpana, M.; Ratnavelu, K.; Balasubramaniam, P.; Kamali, M. Synchronization of chaotic-type delayed neural networks and its application. *Nonlinear Dyn.* **2018**, *93*, 543–555. [CrossRef]
6. Li, M.; Guo, Y.; Huang, J.; Li, Y. Cryptanalysis of a chaotic image encryption scheme based on permutation-diffusion structure. *Signal Process. Image Commun.* **2018**, *62*, 164–172. [CrossRef]
7. Zhang, X.; Wang, L.; Zhou, Z.; Niu, Y. A chaos-based image encryption technique utilizing hilbert curves and h-fractals. *IEEE Access* **2019**, *7*, 74734–74746. [CrossRef]
8. Xie, E.Y.; Li, C.; Yu, S.; Lu, J. On the cryptanalysis of Fridrich's chaotic image encryption scheme. *Signal Process.* **2017**, *132*, 150–154. [CrossRef]
9. Panna, B.; Kumar, S.; Jha, R.K. Image Encryption Based on Block-wise Fractional Fourier Transform with Wavelet Transform. *IETE Tech. Rev.* **2019**, *36*, 600–613. [CrossRef]
10. Noshadian, S.; Ebrahimzade, A.; Kazemitabar, S. Optimizing chaos based image encryption. *Multimed. Tools Appl.* **2018**, *77*, 25569–25590. [CrossRef]
11. Musanna, F.; Dangwal, D.; Kumar, S.; Malik, V. A chaos-based image encryption algorithm based on multiresolution singular value decomposition and a symmetric attractor. *Imaging Sci. J.* **2020**, *68*, 24–40. [CrossRef]
12. Li, Y.; Wang, C.; Chen, H. A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation. *Opt. Lasers Eng.* **2017**, *90*, 238–246. [CrossRef]
13. El-Khamy, S.; Korany, N.; ElSherif, M. A security enhanced robust audio steganography algorithm for image hiding using sample comparison in discrete wavelet transform domain and RSA encryption. *Multimed. Tools Appl.* **2017**, *76*, 24091–24106. [CrossRef]
14. Feng, W.; Zhang, J. Cryptanalzing a Novel Hyper-Chaotic Image Encryption Scheme Based on Pixel-Level Filtering and DNA-Level Diffusion. *IEEE Access* **2020**, *8*, 209471–209482. [CrossRef]
15. Tsafack, N.; Sankar, S.; Abd-El-Atty, B.; Kengne, J.; Jithin, K.C.; Belazi, A.; Mehmood, I.; Bashir, A.; Song, O.Y.; Abd El-Latif, A. A New Chaotic Map With Dynamic Analysis and Encryption Application in Internet of Health Things. *IEEE Access* **2020**, *8*, 137731–137744. [CrossRef]
16. Wang, N.; Li, Q.; Abd El-Latif, A.; Peng, J.; Yan, X.; Niu, X. A novel template protection scheme for multibiometrics based on fuzzy commitment and chaotic system. *Signal Image Video Process.* **2015**, *9*, 99–109. [CrossRef]

17. Abd EL-Latif, A.A.; Abd-El-Atty, B.; Abou-Nassar, E.M.; Venegas-Andraca, S.E. Controlled alternate quantum walks based privacy preserving healthcare images in Internet of Things. *Opt. Laser Technol.* **2020**, *124*, 105942. [CrossRef]
18. 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]
19. Hua, Z.; Zhou, Y.; Huang, H. Cosine-transform-based chaotic system for image encryption. *Inf. Sci.* **2019**, *480*, 403–419. [CrossRef]
20. Liu, Y.; Zhang, J.; Han, D.; Wu, P.; Moon, Y.S. A multidimensional chaotic image encryption algorithm based on the region of interest. *Multimed. Tools Appl.* **2020**, *79*, 17669–17705. [CrossRef]
21. Li, C.; Luo, G.; Qin, K.; Li, C. An image encryption scheme based on chaotic tent map. *Nonlinear Dyn.* **2017**, *87*, 127–133. [CrossRef]
22. Ozkaynak, F. Brief review on application of nonlinear dynamics in image encryption. *Nonlinear Dyn.* **2018**, *92*, 305–313. [CrossRef]
23. Ouannas, A.; Karouma, A.; Grassi, G.; Pham, V.; Luong, V.S. A novel secure communications scheme based on chaotic modulation, recursive encryption and chaotic masking. *Alex. Eng. J.* **2021**, *60*, 1873–1884. [CrossRef]
24. Ratnavelu, K.; Kalpana, M.; Balasubramaniam, P.; Wong, K.; Raveendran, P. Image encryption method based on chaotic fuzzy cellular neural networks. *Signal Process.* **2017**, *140*, 87–96. [CrossRef]
25. Cheng, G.; Wang, C.; Xu, C. A novel hyper-chaotic image encryption scheme based on quantum genetic algorithm and compressive sensing. *Multimed. Tools Appl.* **2020**, *79*, 29243–29263. [CrossRef]
26. Roy, A.; Misra, A.; Banerjee, S. Chaos-based image encryption using vertical-cavity surface-emitting lasers. *Optik* **2019**, *176*, 119–131. [CrossRef]
27. Li, C.; Zhang, Y.; Xie, E.Y. When an attacker meets a cipher-image in 2018: A year in review. *J. Inf. Secur. Appl.* **2019**, *48*, 102361. [CrossRef]
28. 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]
29. He, C.; Ming, K.; Wang, Y.; Wang, Z. A Deep Learning Based Attack for The Chaos-based Image Encryption. *arXiv* **2019**, arXiv:1907.12245.
30. Li, G.; Yang, B.; Pu, Y.; Xu, W. Synchronization of generalized using to image encryption. *Int. J. Pattern Recognit. Artif. Intell.* **2017**, *31*, 1754009. [CrossRef]
31. Norouzi, B.; Mirzakuchaki, S. An image encryption algorithm based on DNA sequence operations and cellular neural network. *Multimed. Tools Appl.* **2017**, *76*, 13681–13701. [CrossRef]
32. Zhang, L.; Zhang, X. Multiple-image encryption algorithm based on bit planes and chaos. *Multimed. Tools Appl.* **2020**, *79*, 20753–20771. [CrossRef]
33. Li, M.; Fan, H.; Xiang, Y.; Li, Y.; Zhang, Y. Cryptanalysis and Improvement of a Chaotic Image Encryption by First-Order Time-Delay System. *IEEE Multimed.* **2018**, *25*, 92–101. [CrossRef]
34. Zhang, X.; Liu, W.; Dundar, M.; Badve, S.; Zhang, S. Towards large-scale histopathological image analysis: Hashing-based image retrieval. *IEEE Trans. Med. Imaging* **2015**, *34*, 496–506. [CrossRef]
35. Zhang, X.; Wang, C.; Zheng, Z. An efficient chaotic image encryption algorithm based on self-adaptive model and feedback mechanism. *KSII Trans. Internet Inf. Syst.* **2017**, *11*, 1785–1801.
36. Musanna, F.; Kumar, S. A novel fractional order chaos-based image encryption using Fisher Yates algorithm and 3-D cat map. *Multimed. Tools Appl.* **2019**, *78*, 14867–14895. [CrossRef]
37. Wang, J.; Zhi, X.; Chai, X.; Lu, Y. Chaos-based image encryption strategy based on random number embedding and DNA-level self-adaptive permutation and diffusion. *Multimed. Tools Appl.* **2021**, *80*, 16087–16122. [CrossRef]
38. Lin, M.; Long, F.; Guo, L. Grayscale image encryption based on Latin square and cellular neural network. In Proceedings of the 2016 Chinese Control and Decision Conference (CCDC), Yinchuan, China, 28–30 May 2016; pp. 2787–2793.
39. Alawida, M.; Samsudin, A.; Sen Teh, J.; Alkhawaldeh, R.S. A new hybrid digital chaotic system with applications in image encryption. *Signal Process.* **2019**, *160*, 45–58. [CrossRef]
40. Preishuber, M.; Huetter, T.; Katzenbeisser, S.; Uhl, A. Depreciating Motivation and Empirical Security Analysis of Chaos-Based Image and Video Encryption. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2137–2150. [CrossRef]
41. The USC-SIPI Image Database. Available online: http://sipi.usc.edu/database (accessed on 23 June 2021).
42. The Ground Truth Database. Available online: http://www.cs.washington.edu/research/imagedatabase (accessed on 23 June 2021).
43. Chen, G.; Mao, Y.; Chui, C. A symmetric image encryption scheme based on 3D chaotic cat maps. *Chaos Solitons Fractals* **2004**, *21*, 749–761. [CrossRef]
44. Wen, H.; Yu, S.; Luuml, J. Breaking an Image Encryption Algorithm Based on DNA Encoding and Spatiotemporal Chaos. *Entropy* **2019**, *21*, 246. [CrossRef] [PubMed]
45. Sasikaladevi, N.; Geetha, K.; Sriharshini, K.; Durga Aruna, M. RADIANT - hybrid multilayered chaotic image encryption system for color images. *Multimed. Tools Appl.* **2019**, *78*, 11675–11700. [CrossRef]
46. 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] [PubMed]
47. Khan, M.; Ahmad, J.; Javaid, Q.; Saqib, N. An efficient and secure partial image encryption for wireless multimedia sensor networks using discrete wavelet transform, chaotic maps and substitution box. *J. Mod. Opt.* **2017**, *64*, 531–540. [CrossRef]

48. Weng, 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.
49. Faragallah, O.S.; Afifi, A.; ElShafai, W.; ElSayed, H.S.; Naeem, E.A.; Alzain, M.A.; AlAmri, J.F.; Soh, B.; ElSamie, F.E.A. Investigation of Chaotic Image Encryption in Spatial and FrFT Domains for Cybersecurity Applications. *IEEE Access* **2020**, *8*, 42491–42503. [CrossRef]
50. 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]
51. Mani, P.; Rajan, R.; Shanmugam, L.; Hoon Joo, Y. Adaptive control for fractional order induced chaotic fuzzy cellular neural networks and its application to image encryption. *Inf. Sci.* **2019**, *491*, 74–89. [CrossRef]
52. Meng, L.; Yin, S.; Zhao, C.; Li, H.; Sun, Y. An improved image encryption algorithm based on chaotic mapping and discrete wavelet transform domain. *Int. J. Netw. Secur.* **2020**, *22*, 155–160.
53. Luo, Y.; Yu, J.; Lai, W.; Liu, L. A novel chaotic image encryption algorithm based on improved baker map and logistic map. *Multimed. Tools Appl.* **2019**, *78*, 22023–22043. [CrossRef]
54. 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]
55. Pan, X.; Wu, J.; Li, Z.; Zhang, C.; Deng, C.; Zhang, Z.; Wen, H.; Gao, Q.; Yang, J.; Yi, Z.; et al. Laguerre-Gaussian mode purity of Gaussian vortex beams. *Optik* **2021**, *230*, 166320. [CrossRef]
56. Yan, X.; Wang, X.; Xian, Y. Chaotic Image Encryption Algorithm Based on Fractional Order Scrambling Wavelet Transform and 3D Cyclic Displacement Operation. *IEEE Access* **2020**, *8*, 208718–208736. [CrossRef]
57. Li, C.; Lin, D.; Feng, B.; Lu, J.; Hao, F. Cryptanalysis of a Chaotic Image Encryption Algorithm Based on Information Entropy. *IEEE J. Transl. Eng. Health Med.* **2018**, *6*, 75834–75842. [CrossRef]
58. Joshi, A.B.; Kumar, D.; Mishra, D.; Guleria, V. Colour-image encryption based on 2D discrete wavelet transform and 3D logistic chaotic map. *J. Mod. Opt.* **2020**, 67, 933–949. [CrossRef]
59. Li, G.; Wang, L. Double chaotic image encryption algorithm based on optimal sequence solution and fractional transform. *Vis. Comput.* **2019**, *35*, 1267–1277. [CrossRef]
60. Yin, Q.; Wang, C. A New Chaotic Image Encryption Scheme Using Breadth-First Search and Dynamic Diffusion. *Int. J. Bifurc. Chaos* **2018**, *28*, 1850047. [CrossRef]
61. Lai, H.; Yan, P.; Shu, X.; Wei, Y.; Yan, S. Instance-aware hashing for multi-label image retrieval. *IEEE Trans. Image Process.* **2016**, *25*, 2469–2479. [CrossRef]