information

Article

Image Encryption Algorithm Using Multi-Level Permutation and Improved Logistic–Chebyshev Coupled Map

Mingfang Jiang [1,2] and Hengfu Yang [1,2,*]

[1] School of Computer Science, Hunan First Normal University, Changsha 410205, China; kycjiangmingfang@hnfnu.edu.cn
[2] Hunan Provincial Key Laboratory of Informationization for Basic Education, Hunan First Normal University, Changsha 410205, China
* Correspondence: hengfuyang@hnfnu.edu.cn

Abstract: To improve the randomness of the Chebyshev chaotic sequences by coupling the Logistic map and the Chebyshev map, a new one-dimensional Logistic–Chebyshev chaotic map (LCCM) is first presented in this paper. Several tests, including the bifurcation diagram, Lyapunov exponents, and information entropy, are employed to analyze the dynamics of the LCCM. The proposed LCCM has better ergodicity and unpredictability than the traditional Chebyshev map. Next, a new image encryption algorithm based on the LCCM and multi-level manipulation is proposed. The LCCM is used to control the pixel permutation, bit-level shuffling, and subsequent pixel diffusion based on the modulo and XOR operation. Extensive experiments, including histogram analysis, information entropy, adjacent pixel correlation, and key sensitivity, show that the image encryption algorithm has high security and can effectively resist malicious attacks.

Keywords: chaotic encryption; image encryption; Chebyshev map; multi-level permutation

Citation: Jiang, M.; Yang, H. Image Encryption Algorithm Using Multi-Level Permutation and Improved Logistic–Chebyshev Coupled Map. *Information* **2023**, *14*, 456. https://doi.org/10.3390/info14080456

Academic Editor: Murilo da Silva Baptista

Received: 2 July 2023
Revised: 28 July 2023
Accepted: 9 August 2023
Published: 13 August 2023

1. Introduction

Chaos is the stochastic behavior of nonlinear systems. Chaotic sequences generated via chaotic systems have the characteristics of noise-like, complex structures and extreme sensitivity to initial conditions. Now, chaotic maps are widely used in the design of image encryption systems. In 1998, Friedrich proposed a chaotic image encryption scheme [1], which consists of two stages: confusion and diffusion. The two processes can hide effectively high redundancy and strong correlation of digital images. Logistic maps and Chebyshev maps are typical 1D nonlinear dynamic systems. These two chaotic maps are simple but possess good nonlinear dynamic characteristics and are widely used in chaotic image encryption algorithms. Sabery and Yaghoobi proposed a new chaotic image encryption method based on Logistic maps [2]. The encryption scheme produces encrypted images by changing both the image pixels and gray-level values. Zhu and Li proposed an improved Logistic encryption algorithm [3]. In the algorithm, nine chaotic sequences produced via the Logistic map are used to confuse and diffuse image pixels. To reduce the time consumption of image encryption, Hazarika and Saikia [4] proposed a selective encryption method using a Logistic map, where the encryption/decryption process is implemented in spatial or DCT domains. Hua et al. [5] proposed a new 2D Sine–Logistic modulation map (2D-SLMM) using the Logistic and Sine maps and further developed a new image encryption algorithm by exploiting a new chaotic magic transform (CMT) and the 2D-SLMM compound chaotic map. Hua and Zhou proposed a 2D Logistic-adjusted-Sine map (2D-LASM) and designed a new image encryption scheme using 2D-LASM [6]. Wei and Jiang proposed a fast image encryption method using parallel compressive sensing (PCS), a Logistic–Tent system (LTS), and DNA sequencing [7]. A mechanism of adding random values to plain images is designed to further enhance encryption security. In 2022, Kumar et al. [8] presented an image encryption technique using Logistic and Tent maps. Compared to the Logistic map, the

Chebyshev map has a larger parameter space and is often employed to mix the plaintext image information [9]. Huang [10] presented a chaotic image encryption algorithm that uses the nonlinear Chebyshev map to generate the key stream. The multiple permutations of pixels are used to decrease the strong correlation between the adjacent pixels in the original plain image. But image encryption schemes based on the Chebyshev map have low sensitivity to the changes in plain images [11]. To overcome this defect, Qi et al. [12] constructed a new two-dimensional Henon–Chebyshev map (2D-HCM) by compounding the Henon map [13] and Chebyshev map. The new chaotic system has better chaotic behaviors, a wider chaotic range, and finer ergodicity. The 2D-HCM is further used to design a chaotic image encryption method. To enlarge the key space of Logistic maps, Dai and Wang proposed a chaotic encryption algorithm suitable for medical images by combining Logistic maps and Chebyshev maps [14]. Liu et al. proposed a 2D chaotic map, called the Logistic-Adjusted-Chebyshev map (2D-LACM) [15], which enlarges the range of chaotic control parameters. Wang and Du proposed two compound chaotic systems [16]. They are Logistic–Chebyshev map (1DLCM) and Logistic–Chebyshev dynamic coupled map lattices (LCDCML). These two systems are used to design a pixel-level and bit-level image encryption algorithm. Experimental simulation shows the feasibility and effectiveness of the encryption scheme. Basha et al. [17] proposed a bit-level color image encryption algorithm using a Logistic–Sine–Tent–Chebyshev (LSTC) map, and the LSTC map, cyclic shifts, and the XOR operation are exploited in the image diffusion. Extensive experiments show its good resistance to statistical attacks and differential attacks.

To enhance the randomness and ergodicity of compound Chebyshev maps, an improved Logistic–Chebyshev hybrid chaotic system is developed by using the output of the Logistic sequence as the input of the Chebyshev map. Then, a new bit-level image encryption algorithm is proposed based on the improved chaotic map. The rest of the paper is organized as follows: Section 2 introduces the improved Logistic–Chebyshev chaotic map. Section 3 expounds on our proposed image encryption method based on the improved chaotic system. Experiments and results are presented in Section 4. Finally, a brief conclusion is drawn in Section 5.

2. Improved Logistic–Chebyshev Map

Although Logistic and Chebyshev chaotic maps have wide applications in image encryption, they also have some defects, such as blank windows and small parameter space. To overcome these defects, this section presents the new compound Logistic–Chebyshev chaotic map by coupling Logistic and Chebyshev chaotic maps (LCCM).

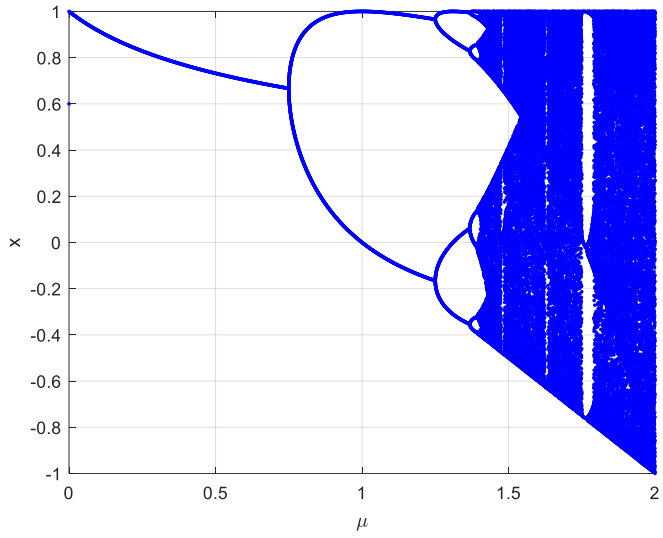
2.1. Classic Chaotic Maps

2.1.1. Logistic Map

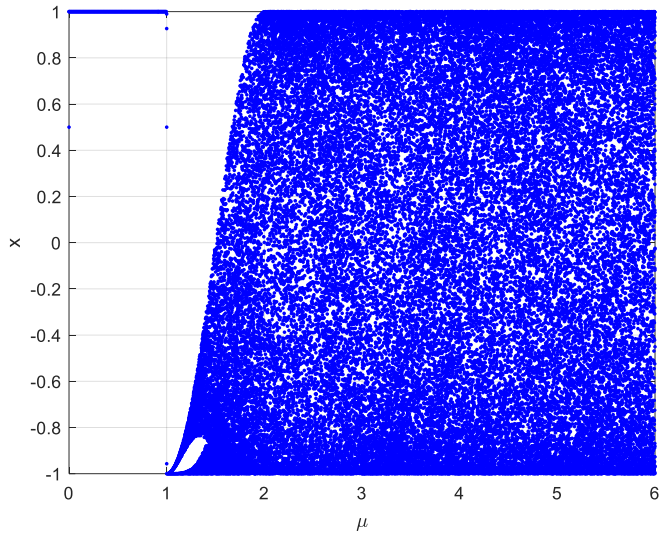
The Logistic map is a quadratic polynomial map, which is one of the typical maps representing complex nonlinear behavior. The mathematical expression is written as follows:

$$x_{n+1} = 1 - \mu x_n^2,\ \mu \in (0, 2], x_n \in (-1, 1) \tag{1}$$

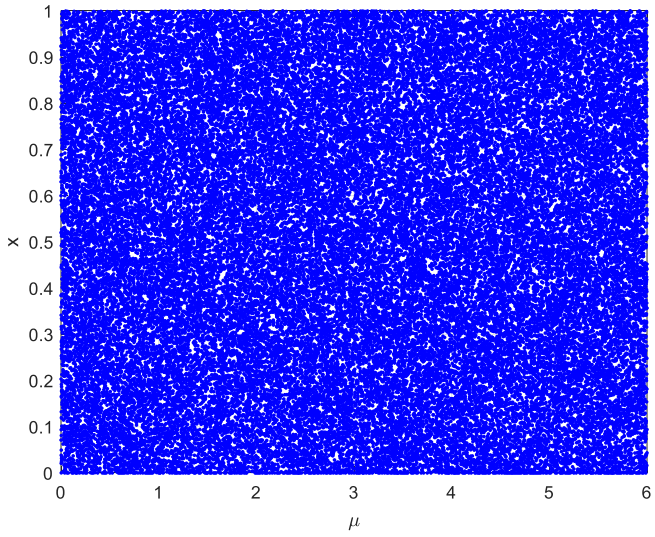
where μ is the bifurcation parameter, and only when $1.401155 \leq \mu \leq 2$, the Logistic map falls into the chaotic state. The bifurcation diagram of the Logistic map is shown in Figure 1a.



(**a**)



(**b**)



(**c**)

Figure 1. Bifurcation diagrams. (**a**) Logistic, (**b**) Chebyshev, and (**c**) LCCM.

2.1.2. Chebyshev Map

The Chebyshev map is one of the 1D chaotic maps with good nonlinear dynamic characteristics. The map is in a chaotic state when the control parameter is $\omega \in [2, \infty)$. It can be defined as follows, where Figure 1b is the bifurcation diagram of the Chebyshev chaotic map:

$$x_{n+1} = cos(\omega \cdot arccos x_n),\ x_n \in [-1, 1] \tag{2}$$

2.2. Proposed LCCM Chaotic System

To enhance the randomness and enlarge the parameter space of chaotic systems, we design a compound chaotic system by combining Logistic and Chebyshev chaotic maps (LCCM). The detailed process of building the chaotic system is as follows:

First, to improve the randomness of the Chebyshev chaotic map, we compound the Logistic map and the Chebyshev map by using the output of the Logistic map as the input of the Chebyshev map, as shown below:

$$x_{n+1} = cos(\mu \cdot arccos(1 - 2x_n^2)) \tag{3}$$

Second, considering that the Chebyshev map is not in a chaotic state when the control parameter $\omega \in [0,\ 2)$, we change the parameter μ to $(\mu + 2)$ to skip the blank window. Thus, the compound LCCM map can be rewritten as:

$$x_{n+1} = cos((\mu + 2) \cdot arccos(1 - 2x_n^2)) \tag{4}$$

where $x_n \in [-1, 1]$, and the new LCCM chaotic system stays in the chaotic state when $\mu \in [0, \infty)$.

2.3. Chaotic Behaviors

The bifurcation diagram depicts the process of a series of sudden changes in the state of a nonlinear system when it changes with the control parameters. Figure 1 shows the bifurcation diagrams of Logistic, Chebyshev, and LCCM.

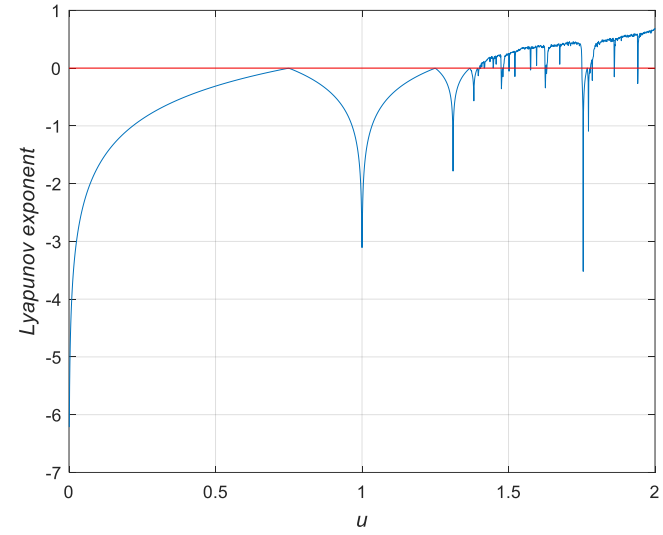
For the traditional Logistic map, as shown in Figure 1a, the iterative values are distributed throughout the entire range with $(-1, 1)$ only when $1.401155 \leq \mu \leq 2$. When $\mu < 1.401155$, the distribution of iterative results is concentrated, and the stable window range is less than $(-1, 1)$.

For Chebyshev maps, as shown in Figure 1b, the distribution of the generated sequences is relatively concentrated when the control parameter $\mu < 2$.

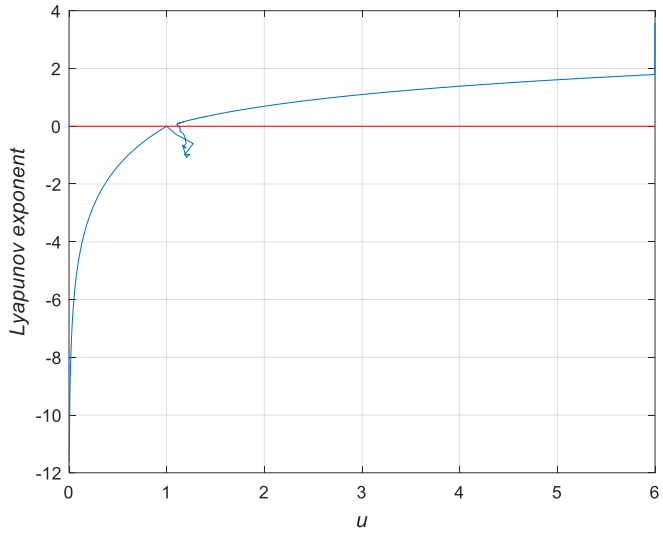
Compared with the above two maps, the proposed LCCM map has a larger parameter space and is uniformly distributed for any $\mu \in [0,\ \infty)$. Moreover, it contains no blank windows. Its bifurcation diagram is illustrated in Figure 1c.

The Lyapunov Exponent (LE) represents the average exponential divergence rate of adjacent trajectories in the phase space, which is an important numerical feature characterizing the stability of dynamic systems. The chaotic maps with positive Lyapunov exponent values for all control parameter values are chaotic, and a larger LE means better chaotic behaviors. Figure 2 shows the comparison among the different chaotic maps for the parameter LE.

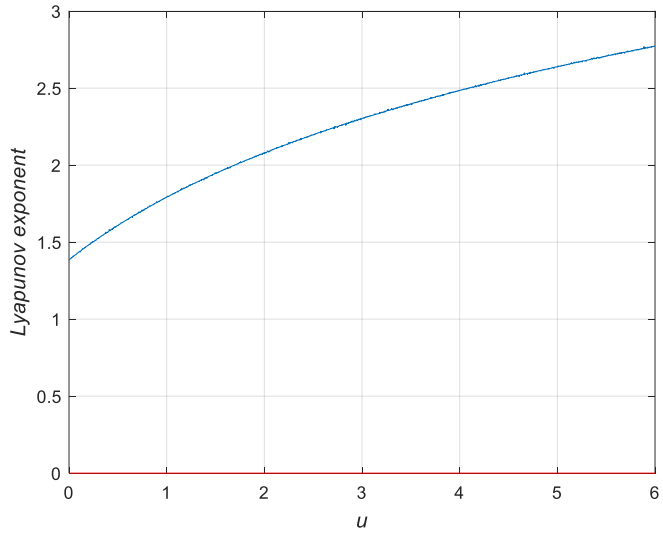
According to Figure 2a, the LE values of the 1D Logistic map are positive in a narrow range of the control parameter, and the Chebyshev map, as shown in Figure 2b, has more positive LE values than the Logistic map. As to the proposed LCCM map, it has positive LE values for all $\mu \in [0,\ \infty)$. Thus, the outputs of the LCCM are more unpredictable.



(**a**)



(**b**)



(**c**)

Figure 2. Bifurcation diagrams. (**a**) Logistic, (**b**) Chebyshev, and (**c**) LCCM.

2.4. Randomness Test

In this subsection, we employ the NIST SP 800-22 tests to evaluate the randomness performances of the LCCM. In the random tests, each test produces a p-value which is a real number in [0, 1]. If the p-value is greater than a predefined significance level a = 0.01, we can say that the random sequence can pass the test successfully. All test results are listed in Table 1. From Table 1, the random sequences produced via the LCCM pass all NIST SP 800-22 tests and have satisfactory statistical properties. Thus, the LCCM map is suitable for image encryption algorithms.

Table 1. Test results of NIST for the LCCM.

Test Name	p-Value	Results
Frequency test	0.6245	pass
Block Frequency test	0.7148	pass
Cusum-Forward test	0.8653	pass
Cusum-Reverse test	0.3261	pass
Runs test	0.3984	pass
Long Runs test of Ones	0.7215	pass
Binary Matrix Rank Test	0.6279	pass
Spectral DFT test	0.5462	pass
Non-overlapping test Templates	0.7906	pass
Overlapping test Templates	0.9270	pass
Maurer's Universal test	0.3844	pass
Approximate Entropy test	0.8568	pass
Random Excursions test	03227	pass
Lempel Ziv complexity test	0.6403	pass
Linear complexity test	0.3042	pass
Random Excursions Variant test	0.5318	pass
Serial test	0.8471	pass

3. Image Encryption Based on Improved Logistic–Chebyshev Map

To enhance the security of encryption algorithms, this paper proposes an image encryption algorithm based on an improved Logistic–Chebyshev chaotic map (as shown in Figure 3). It consists of three phases, viz. the key generation phase, multi-level permutation phase, and image diffusion phase.

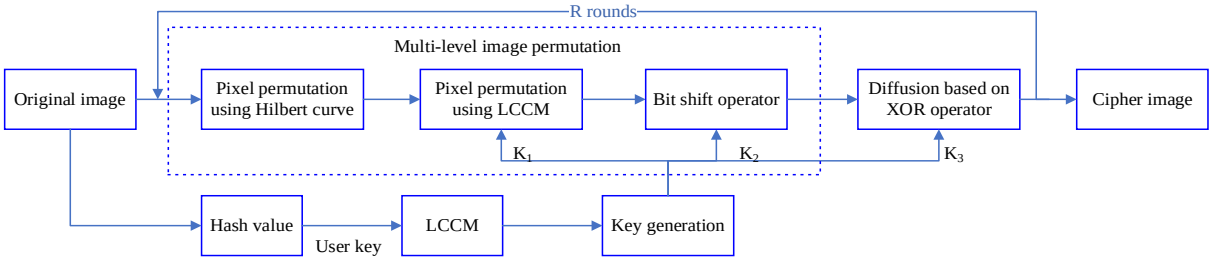


Figure 3. The image encryption process.

3.1. Key Stream Generation

First, we generate three key streams K_1, K_2, and K_3 for multi-level permutation and image diffusion. Given the secret keys u_0 and x_0, the plaintext image P with the size of $M \times N$, the key stream generation process is described as follows.

Step 1: Compute the hash code K of 32 bytes (it will be used as the key in the decryption phase) from the image matrix P with the SHA256 function, where $K = \{k(0), k(1), k(2), \ldots, k(31)\}$, and $k(i)$ is an integer in the interval $[0, 255]$, where $i = 0, 1, 2, \ldots, 31$.

Step 2: Get the image features according to the hash code K. The procedure is described in Equation (5) below:

$$\begin{cases} u_i = k(5(i-1)) \otimes k(5(i-1)+1) \otimes \ldots \otimes k(5(i-1)+4) \otimes k(15) \\ x_i = \frac{k(5(i-1)+16) \otimes k(5(i-1)+17) \otimes \ldots \otimes k(5(i-1)+20) \otimes k(31)}{256} \end{cases}, \quad i = 1,2,3 \quad (5)$$

Step 3: Generate the initial values and the control parameters for the LCCM map shown in Equation (4) by combining the secret keys and the feature values u_i and x_i. It can be written as follows:

$$\begin{cases} \widetilde{\mu}_i = mod(u_i + u_0, 256) \\ \widetilde{x}_i = mod(x_i + x_0, 1) \end{cases}, \; i = 1,2,3 \quad (6)$$

Step 4: Given the different initial values and parameters $\widetilde{\mu}_i$, $\widetilde{x}_i$, and $i = 1,2,3$, we can generate three chaotic sequences y_1, y_2, and y_3 of length $M \times N$ by iterating the proposed LCCM map for $n_0 + M \times N$ times. To obtain a good chaotic effect, the first n_0 values are discarded.

Step 5: To eliminate the local monotonicity of the sequences, a sequence monotony suppression model is developed to enhance their randomness. After doing this, three key streams K_1, K_2, and K_3 of length $M \times N$ are produced, as shown below:

$$K_1 = mod(round(y_1 \times 10^{15}), M \times N) \quad (7)$$

$$K_2 = mod(round(y_2 \times 10^{15}), 8) \quad (8)$$

$$K_3 = mod(round(y_3 \times 10^{15}), 256) \quad (9)$$

3.2. Multi-Level Permutation

The image permutation phase includes pixel permutation based on the Hilbert curve, permutation using the LCCM map, and a bit-shift operator based on the chaotic map. Given the plaintext image P of size $M \times N$, key streams K_1, K_2, and K_3, the detailed procedure of the multi-level permutation is explained below.

Step 1: Pixel permutation based on the Hilbert curve. Divide the image matrix P of size $M \times N$ into four sub-blocks' size of $\frac{M}{2} \times \frac{N}{2}$, then perform image scrambling based on the Hilbert curve on each image block. Repeat this process for each image block until each block is a pixel block of size 2×2 and the scrambled image P_H of size $M \times N$ is obtained.

Step 2: Pixel permutation using the LCCM map. Convert the scrambled image P_H into a one-dimensional sequence P_{H1} of length $M \times N$. Use the key stream K_1 produced via Equation (7) to scramble the sequence P_{H1} and generate the scrambled sequence P_1. The scrambling operation is shown in Equation (10) below:

$$P_1 = P_{H1}(K_1) \quad (10)$$

Step 3: Bit-level circular shift operation. Each element in the sequence K_2 generated via Equation (8) is an integer between 0 and 7, and each element in the sequence P_1 is an integer between 0 and 255, that is, an 8-bit binary number. We can use the chaotic sequence K_2 to permutate each element of the sequence P_1. First, transform each element $P_1(i)$ into its binary form, and then perform a left shift $K_2(i)$ bits on it, and finally convert it into a decimal, which is recorded as $P_c(i)$, as shown below:

$$P_c(i) = CBshift(P_1(i), K_2(i)), \; i = 0,1,\ldots,M \times N - 1 \quad (11)$$

where the function $CBshift(x,y)$ performs a circular left shift on the binary representation of x by y bits.

3.3. Image Diffusion

The scrambled image only changes the pixel position. To fully confuse the gray values of the image, a chaotic sequence generated via Equation (9) is used to diffuse the scrambled image. Finally, the ciphertext image E is obtained. The diffusion process is shown in Equation (12) below:

$$\begin{cases} E(0) = P_c(0) \otimes P_c(MN-1) \otimes K_3(0), & i = 0 \\ E(i) = P_c(i) \otimes E(i-1) \otimes K_3(i), & i \neq 0 \end{cases} \tag{12}$$

3.4. Image Decryption

Image decryption is the reverse process of image encryption, and the same key streams are used for both encryption and decryption processes. Given ciphertext image E of size $M \times N$, key streams K_1, K_2, and K_3, the decryption process is described as the following steps.

Step 1: Use the same key to generate key streams K_1, K_2, and K_3 using Equations (7)–(9).

Step 2: Perform the XOR operation on the ciphertext image E to recover the diffused image into its original value, and then obtain image D., as shown below:

$$\begin{cases} D(0) = E(0) \otimes P_c(MN-1) \otimes K_3(0), & i = 0 \\ D(i) = E(i) \otimes E(i-1) \otimes K_3(i), & i \neq 0 \end{cases} \tag{13}$$

Step 3: The plaintext image P will be recovered without distortion by performing inverse processes of pixel cyclic shift, chaotic scrambling, and Hilbert scrambling on image D, respectively.

4. Experimental Results

In the experiment, standard grayscale images with the size of 256 × 256 are selected for experimental testing. The proposed algorithm is implemented in MATLAB R2018a. All experiments are done on one single core of an Intel Core (TM) i7-12700H with 2.30 GHz and 16 GB of memory. Figure 4 lists the four test images (couple, Goldhill, lighthouse, and peppers), the ciphertext images after one round of encryption, and the corresponding deciphertext images. The original plaintext image information cannot be seen at all from the ciphertext images in the center column Figure 4, indicating that the encryption algorithm has a good encryption effect. One can see that users with the correct encryption key can completely restore the original images from the ciphertext images in the right column of Figure 4.

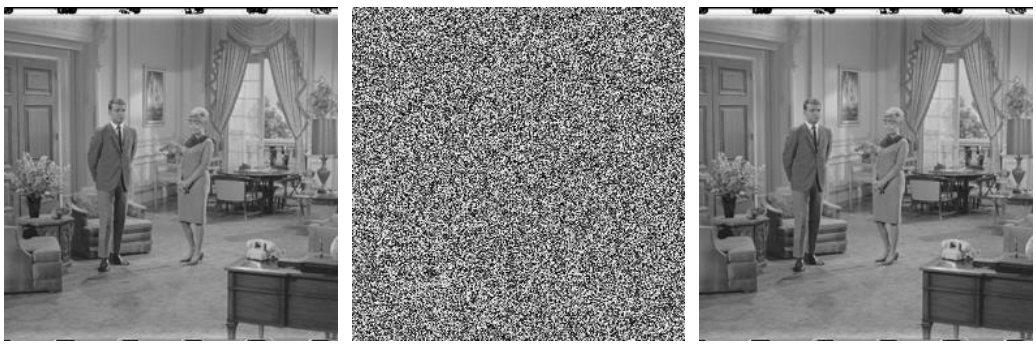


Figure 4. *Cont.*

Figure 4. Some examples of image encryption. (**left**) column: Plaintext image, (**center**) column: ciphertext image, and (**right**) column: decipher image.

4.1. Histogram Analysis

Image histogram reflects the statistical characteristics of grayscale images. The comparison of image histograms before and after encryption is shown in Figure 5. From Figure 5, it can be seen that the histogram before encryption has great fluctuation, while the histograms after encryption are more uniformly distributed. It is hard to obtain any meaningful content of the original image from the encrypted image. This indicates that this encryption algorithm can resist statistical attacks well.

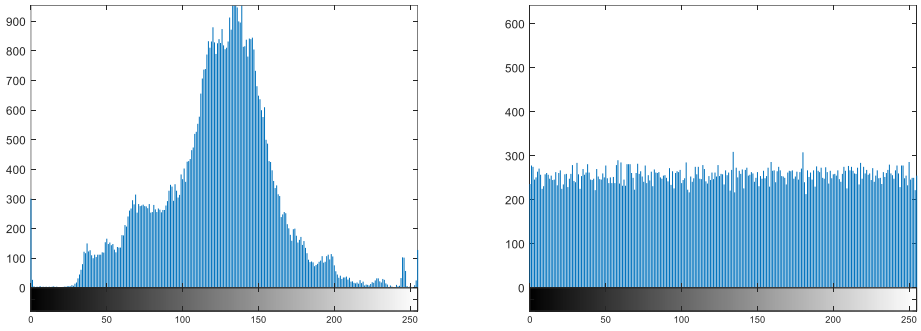


Figure 5. *Cont.*

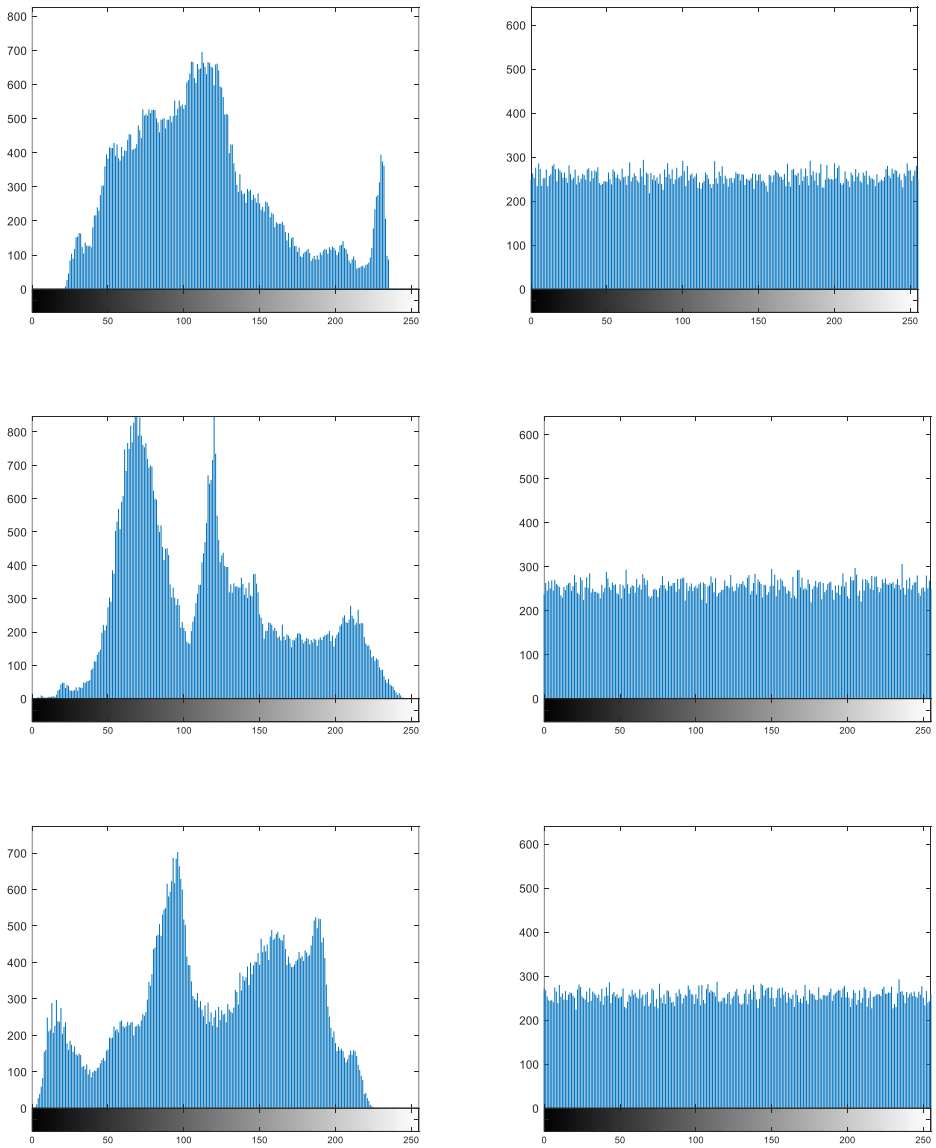


Figure 5. Histograms of plaintext images (**Left**) and ciphertext images (**Right**). Up to down: couple, Goldhill, lighthouse, and peppers.

4.2. Correlation Analysis

The correlation between the adjacent pixels is an important indicator for the evaluation of image encryption. The correlation coefficient between two adjacent pixels of an image is calculated in horizontal, vertical, and diagonal directions.

Take the 'couple' image as an example; Figure 6 illustrates the distributions of the correlation of a plaintext image and ciphertext image in the three directions. The results of the correlation coefficients of the original images and encrypted images are shown in Table 2. It can be seen that from Table 2, the values of the plaintext images nearer to 1 mean a high adjacent pixel correlation, while that of the ciphertext images nearer to 0 indicates a low correlation. So, the proposed scheme shows good resistance against attacks.

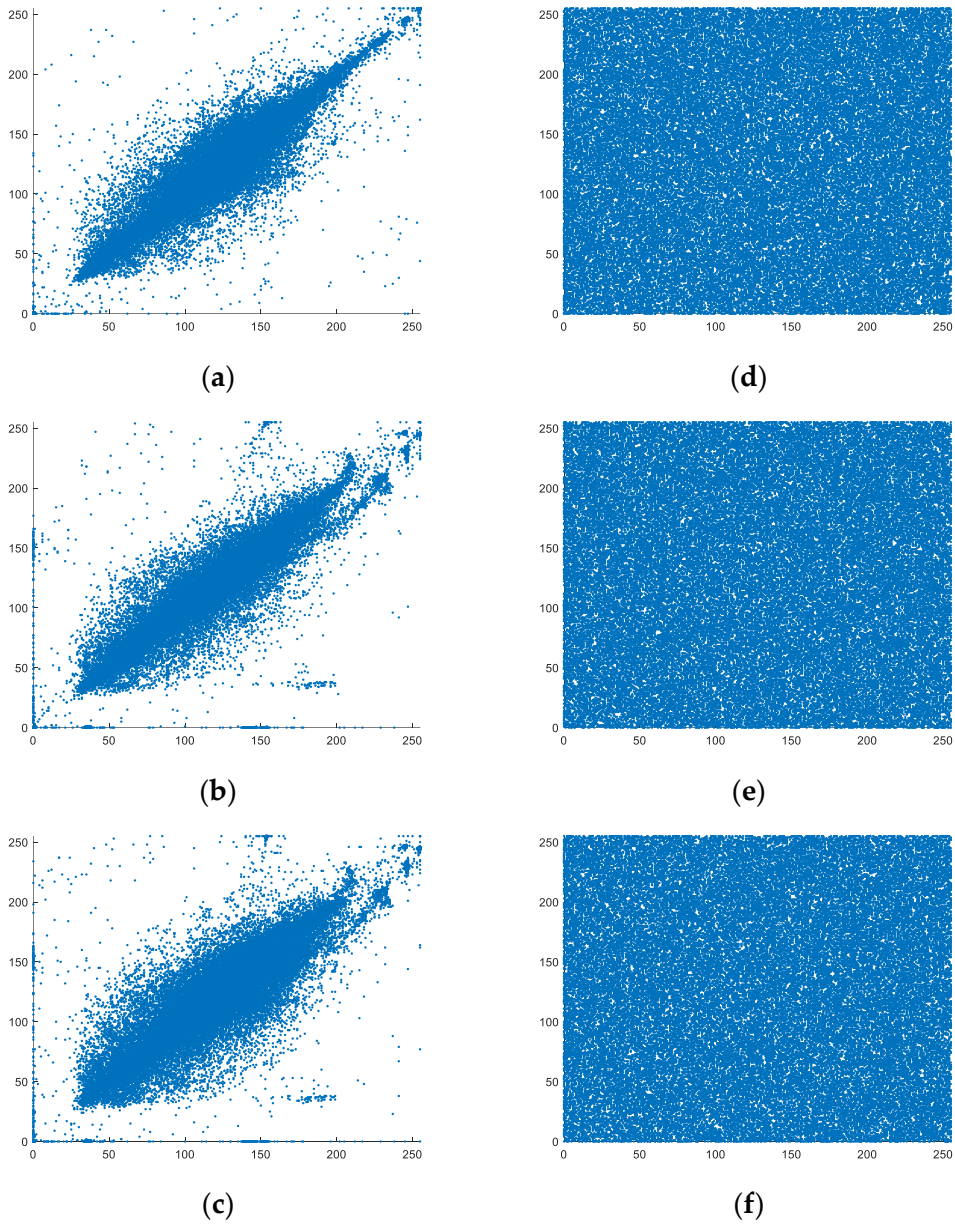


Figure 6. The correlation distribution in the (**a**) horizontal direction, (**b**) vertical direction, and (**c**) diagonal direction of the original couple image; in the (**d**) horizontal direction, (**e**) vertical direction, and (**f**) diagonal direction of the ciphered image of couple.

Table 2. Correlation coefficient of plaintext and ciphertext images.

Images	Correlation of Plaintext Image			Correlation of Ciphertext image		
	Horizontal Correlation	Vertical Correlation	Diagonal Correlation	Horizontal Correlation	Vertical Correlation	Diagonal Correlation
couple	0.9263	0.8766	0.8359	0.0086	0.0032	0.0042
Goldhill	0.9646	0.9531	0.9245	0.0026	0.0070	0.0009
lighthouse	0.8970	0.9298	0.8435	0.0010	0.0000	0.0029
peppers	0.9508	0.9574	0.9212	0.0020	0.0021	0.0002

The comparison results of the correlation coefficients with the other four encryption algorithms are listed in Table 3. From Table 3, one can see that the proposed algorithm has

the lowest correlation on average. The new encryption scheme can reduce the correlation of the adjacent pixels effectively.

Table 3. Comparisons of correlation coefficients of the peppers image under different image encryption algorithms.

Pepper	Horizontal 0.9508	Vertical 0.9574	Diagonal 0.9212	Average
Ref. [9]	0.0103	0.0121	0.0310	0.0178
Ref. [13]	0.0023	0.0016	0.0046	0.0028
Ref. [7]	0.0082	0.0027	0.0035	0.0048
Ref. [17]	0.0123	0.0052	0.0014	0.0063
Our scheme	0.0020	0.0021	0.0002	0.0014

4.3. Information Entropy

Information entropy is an important statistical metric that reflects the randomness of information. The more evenly distributed the grayscale values of an image are, the greater the information entropy. The ciphertext images generated via good encryption will have an entropy value very close to 8. Table 4 shows the information entropy of the proposed algorithm and the four comparable algorithms. The average entropy value of the test images of our scheme arrives at 7.9975. It is larger than that of the comparable algorithms. So, our encryption algorithm is robust against entropy attacks.

Table 4. Information entropy of different algorithms.

Images	Plaintext Image	Ciphertext Image	Ref. [9]	Ref. [13]	Ref. [7]
couple	7.1662	7.9971	7.9982	7.9918	7.9962
Goldhill	7.4452	7.9976	7.9973	7.9925	7.9839
lighthouse	7.4557	7.9973	7.9926	7.9964	7.9952
peppers	7.5897	7.9979	7.9946	7.9376	7.9913
Average	7.4142	7.9975	7.9957	7.9796	7.9917

4.4. Differential Attack Analysis

The number of pixel change rates (NPCR) and the unified average change intensity (UACI) are widely used to measure the resistance of encryption schemes against differential attacks.

In the experiment, five pixels are randomly selected, and their grayscale values change by one. The corresponding NPCR and UACI were calculated. The average values are shown in Table 5. It can be seen that the NCPR and UACI values of the proposed scheme are close to the ideal value of 0.996 and 0.334, respectively. So, it can meet the security requirements of the encryption algorithm.

Table 5. NPCR and UACI values of the different images.

Images	Couple	Goldhill	Lighthouse	Peppers
NPCR (%)	99.6262	99.5956	99.6185	99.6155
UCAI (%)	33.4716	33.4873	33.4851	33.4718

Table 6 compares the average NPCR and UACI of the ciphertext image generated via our algorithm with other algorithms. From Table 6, it can be seen that the new encryption scheme has a greater value of NPCR and UACI than some previous encryption algorithms, indicating that the proposed algorithm can effectively resist differential attacks.

Table 6. Average NPCR and UACI of various image encryption schemes.

Algorithms	Proposed	Ref. [9]	Ref. [13]	Ref. [7]	Ref. [17]
NPCR (%)	99.6262	99.6243	99.5993	99.5846	99.56
UCAI (%)	33.4716	14.6022	33.4600	33.4510	33.4578

4.5. Resistance against Chosen Plaintext

Due to the use of the original image hash value in the key generation process, the chaotic encryption system will depend on different original plain images. In this way, different images generate different encryption keys because of their different hash values. Thus, this algorithm has high sensitivity to initial values, that is, different initial values correspond to different keys and encryption results. Suppose that the key stream group $(K_1, K_2,$ and $K_3)$ generated from each plain image in Figure 4 is recorded as KG_1. A tiny modified plain image P' is obtained by changing only one bit of the image P, and the key stream associated with the image P' is denoted as KG_2. Here, the difference between the two key streams KG_1 and KG_2 is evaluated in terms of the NPCR and UACI, as listed in Table 7. It can be seen that the key streams generated for two slightly different images are completely distinct and are dependent on the plain images. This means that the proposed scheme can resist chosen plaintext attacks.

Table 7. The difference between the two key streams KG_1 and KG_2.

Image	Differences in Key Streams	
	NPCR (%)	UACI (%)
couple	99.2372	33.0617
Goldhill	99.2693	33.1732
lighthouse	99.2514	33.0115
peppers	99.1656	33.0248

5. Conclusions

In this paper, a new chaotic map, called LCCM, is designed based on the Logistic map and the Chebyshev map. First, the Chebyshev map is dynamically modulated using the output of the Logistic map, and then the parameter space of the Chebyshev map is enlarged by expanding its control parameter. The chaotic performance of the LCCM is analyzed using a bifurcation diagram, Lyapunov exponents, and so on. The analysis results show that the proposed LCCM map has excellent ergodicity and unpredictability. Moreover, a new image encryption algorithm is presented using the LCCM and multi-level permutation. The LCCM is employed to produce key streams to be used in image scrambling and image diffusion phases. Extensive experiments and security performance analyses including histogram analysis, information entropy analysis, and correlation analysis evaluate the scheme's statistical attack resistance. High values of NPCR and NCAI indicate that it is robust against differential attacks. As a result, our encryption method has excellent security and can be used in secure image communication.

Hyperchaotic systems have more excellent dynamical behavior and are preferable for use in chaotic data encryption systems. The LCCM has a smaller parameter space compared to hyperchaotic systems. For future work, compound high-dimension hyperchaotic systems based on Chebyshev will be studied to enlarge the parameter space of chaotic systems and improve their randomness.

Author Contributions: Conceptualization, M.J. and H.Y.; Methodology, M.J. and H.Y.; Investigation, M.J.; Writing—original draft, M.J.; Writing—review and editing, H.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China under Grant 61872408, the Natural Science Foundation of Hunan Province under Grant 2020JJ4238, and the Social Science Foundation of Hunan Province under Grant 19YBA098.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: These datasets were derived from the following public domain resources: https://sipi.usc.edu/database/ (accessed on 10 May 2023).

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

References

1. Fridrich, J. Symmetric ciphers based on two-dimensional chaotic maps. *Int. J. Bifurc. Chaos* **1998**, *8*, 1259–1284. [CrossRef]
2. Sabery, M.K.; Yaghoobi, M. A new approach for image encryption using chaotic Logistic map. In Proceedings of the 2008 International Conference on Advanced Computer Theory and Engineering, Phuket, Thailand, 20–22 December 2008; pp. 585–590.
3. Ai-hong, Z.; Lian, L. Improving for chaotic image encryption algorithm based on logistic map. In Proceedings of the 2010 the 2nd Conference on Environmental Science and Information Application Technology, Wuhan, China, 17–18 July 2010; pp. 211–214.
4. Hazarika, N.; Saikia, M. A novel partial image encryption using chaotic logistic map. In Proceedings of the 2014 International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 20–21 February 2014; pp. 231–236.
5. Hua, Z.; Zhou, Y.; Pun, C.-M.; Chen, C.L.P. 2D Sine Logistic modulation map for image encryption. *Inf. Sci.* **2015**, *297*, 80–94. [CrossRef]
6. Hua, Z.; Zhou, Y. Image encryption using 2D Logistic-adjusted-Sine map. *Inf. Sci.* **2016**, *339*, 237–253. [CrossRef]
7. Wei, D.; Jiang, M. A fast image encryption algorithm based on parallel compressive sensing and DNA sequence. *Optik* **2021**, *238*, 166748. [CrossRef]
8. Kumar, K.; Roy, S.; Rawat, U.; Mishra, I. An efficient image encryption technique using Logistic map and 2D-TCLM. In Proceedings of the Cyber Warfare, Security and Space Research, 28 August 2022; Springer International Publishing: Cham, Switzerland, 2022; Volume 1599, pp. 87–96.
9. Diab, H. An efficient chaotic image cryptosystem based on simultaneous permutation and diffusion operations. *IEEE Access* **2018**, *6*, 42227–42244. [CrossRef]
10. Huang, X. Image encryption algorithm using chaotic Chebyshev generator. *Nonlinear Dyn.* **2012**, *67*, 2411–2417. [CrossRef]
11. Wang, X.; Luan, D.; Bao, X. Cryptanalysis of an image encryption algorithm using Chebyshev generator. *Digit. Signal Process.* **2014**, *25*, 244–247. [CrossRef]
12. Qi, F.; Huang, S.; Li, T.; Yang, H.; Kang, X. 2D Henon-Chebyshev Chaotic map for image encryption. In Proceedings of the 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Zhangjiajie, China, 10–12 August, 2019; pp. 774–781.
13. Shahna, K.U.; Mohamed, A. A novel image encryption scheme using both pixel level and bit level permutation with chaotic map. *Appl. Soft Comput.* **2020**, *90*, 106162. [CrossRef]
14. Dai, Y.; Wang, X. Medical image encryption based on a composition of Logistic Maps and Chebyshev Maps. In Proceedings of the 2012 IEEE International Conference on Information and Automation, Shenyang, China, 6–8 June 2012; pp. 210–214.
15. Liu, L.; Jiang, D.; Wang, X.; Rong, X.; Zhang, R. 2D Logistic-adjusted-Chebyshev map for visual color image encryption. *J. Inf. Secur. Appl.* **2021**, *60*, 102854. [CrossRef]
16. Wang, X.; Du, X. Pixel-level and bit-level image encryption method based on Logistic-Chebyshev dynamic coupled map lattices. *Chaos Solitons Fractals* **2022**, *155*, 111629. [CrossRef]
17. Basha, S.M.; Mathivanan, P.; Ganesh, A.B. Bit level color image encryption using Logistic-Sine-Tent-Chebyshev (LSTC) map. *Optik* **2022**, *259*, 168956. [CrossRef]