Wu, X., Li, Y., Kurths, J. (2015): A new color image encryption scheme using CML and a fractional-order chaotic system. - PloS One, 10, 0119660

DOI: 10.1371/journal.pone.0119660

RESEARCH ARTICLE

A New Color Image Encryption Scheme Using CML and a Fractional-Order Chaotic System

Xiangjun Wu[1,2,3]*, **Yang Li**[1], **Jürgen Kurths**[2,3]

1 College of Software, Henan University, Kaifeng, China, 2 Potsdam Institute for Climate Impact Research (PIK), Potsdam, Germany, 3 Department of Physics, Humboldt University zu Berlin, Berlin, Germany

* xiangjung.wu@gmail.com

Abstract

The chaos-based image cryptosystems have been widely investigated in recent years to provide real-time encryption and transmission. In this paper, a novel color image encryption algorithm by using coupled-map lattices (CML) and a fractional-order chaotic system is proposed to enhance the security and robustness of the encryption algorithms with a permutation-diffusion structure. To make the encryption procedure more confusing and complex, an image division-shuffling process is put forward, where the plain-image is first divided into four sub-images, and then the position of the pixels in the whole image is shuffled. In order to generate initial conditions and parameters of two chaotic systems, a 280-bit long external secret key is employed. The key space analysis, various statistical analysis, information entropy analysis, differential analysis and key sensitivity analysis are introduced to test the security of the new image encryption algorithm. The cryptosystem speed is analyzed and tested as well. Experimental results confirm that, in comparison to other image encryption schemes, the new algorithm has higher security and is fast for practical image encryption. Moreover, an extensive tolerance analysis of some common image processing operations such as noise adding, cropping, JPEG compression, rotation, brightening and darkening, has been performed on the proposed image encryption technique. Corresponding results reveal that the proposed image encryption method has good robustness against some image processing operations and geometric attacks.

Citation: Wu X, Li Y, Kurths J (2015) A New Color Image Encryption Scheme Using CML and a Fractional-Order Chaotic System. PLoS ONE 10(3): e0119660. doi:10.1371/journal.pone.0119660

Academic Editor: Helmut Ahammer, Medical University of Graz, AUSTRIA

Received: May 31, 2014

Accepted: January 16, 2015

Published: March 31, 2015

Data Availability Statement: All relevant data are within the paper and its Supporting Information files.

Funding: This research was jointly supported by the National Natural Science Foundation of China (No. 61004006, http://www.nsfc.gov.cn), China Postdoctoral Science Foundation(No. 2013M530181, http://res.chinapostdoctor.org.cn/BshWeb/index.shtml), the Natural Science Foundation of Henan Province, China (No. 13230010254, http://www.hnkjt.gov.cn/, Program for Science & Technology Innovation Talents in Universities of Henan Province, China (Grant No 14HASTIT042, http://rcloud.edu.cn), the Foundation for University Young Key Teacher Program of Henan Province, China (No. 2011GGJS-025, http://www.

Introduction

Widespread transmission of digital images over various communication media challenges to build credible security methods for the protection of confidential and sensitive information to be transmitted. Hence the security of digital information has become a hot recent topic. Different from text encryption, most conventional ciphers such as Data Encryption Standard (DES), International Data Encryption Algorithm (IDEA), Advanced Encryption Standard (AES), RSA (developed by Rivest, Shamir and Adleman), etc., are not suitable to build cryptosystems for digital images due to inherent features of image data, e.g. bulk data capacity, high redundancy, strong correlation among adjacent pixels, etc. The implementation of these traditional

haedu.gov.cn/), Shanghai Postdoctoral Scientific Program (No. 13R21410600, http://www.21cnhr.gov.cn/doctorarea/), the Science & Technology Project Plan of Archives Bureau of Henan Province (No. 2012-X-62, http://www.hada.gov.cn/) and the Natural Science Foundation of Educational Committee of Henan Province, China (No. 13A520082, http://www.haedu.gov.cn/). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The co-author Jürgen Kurths is a PLOS ONE Editorial Board member. This does not alter the authors' adherence to PLOS ONE Editorial policies and criteria.

algorithms for image encryption usually requires more computation time and power, and causes also other problems such as in handling various data formatting.

In 1989, Matthews developed the first chaotic stream encryption algorithm [1]. After that, many studies have shown that chaos-based algorithms have advantages in applications of bulk data encryption, because the chaotic signals have cryptographically desirable properties such as extreme sensitivity to initial conditions and parameters, long periodicity, ergodicity, high randomness and mixing [2]. In contrast to traditional cryptographic techniques, it has been found that chaos-based image encryption schemes have superior performance with respect to the trade-offs between the security and efficiency.

Since Fridrich presented a symmetric image encryption algorithm using the two-dimensional standard Baker map in 1998 [3], many image cryptosystems have been developed during the past decades. In practical applications, three types of methods, i.e., permutation, diffusion, and their combined form, are usually employed to design image encryption algorithms [4–9]. The aim of permutation is to transform a meaningful image into a meaningless, disordered and unsystematic image through scrambling the positions of the plain-image pixels, which will enhance the computational complexity of a potential chosen-plaintext attack. In the diffusion process, the values of the original image pixels are changed sequentially so that a tiny change for one pixel can spread out to almost all pixels in the whole image. The image encryption methods using either the permutation-only method or the diffusion-only method have some shortcomings in both security and speed [6, 9].

So far, most image encryption algorithms combine scrambling the pixels positions and modifying the grey values of image pixels to achieve required cryptographic properties. In [10], a new color image encryption method was introduced based on chaotic logistic maps. An external secret key of 80-bit length and two chaotic logistic maps are used. The initial conditions for the logistic maps are obtained using the external secret key. Eight different types of operations are applied to encrypt the image pixels. Wang et al. [11] proposed a new color image encryption algorithm based on the logistic map. Some schemes have been suggested to achieve secure image scrambling based on the Baker map [3], Arnold cat map [12], Standard map [5, 13] etc. Note that the above-mentioned chaos-based image cryptosystems are usually based on the low-dimensional and single chaotic systems, which results in fundamental drawbacks such as insufficient key space, slow speed and weak security function.

In order to improve the security and efficiency performance, many image encryption methods based on three-dimensional chaotic systems, hyperchaos and even spatiotemporal chaos have been presented in recent years [4, 14–23]. For example, Chen et al. [4] generalized a two-dimensional chaotic cat map to a three-dimensional one for constructing a real-time secure symmetric encryption scheme, where the three-dimensional cat map is utilized to create confusion in the relationship between the cipher-image and the plain-image. Mao et al. [14] extended the same idea with the three-dimensional chaotic Baker map. In [15], a digital image encryption scheme based on the mixture of chaotic systems was proposed, where a typical coupled map was mixed with a one-dimensional chaotic map and used for high degree security image encryption. Gao and Chen [16] presented a new image encryption scheme, which used an image total shuffling matrix to shuffle the positions of image pixels and then employed a hyper-chaotic system to confuse the relationship between the plain-image and the cipher-image. The authors in [17] pointed out that this method is very weak to a chosen plain-text attack and a chosen cipher-text attack. In [18], a new image authentication scheme based on a cell neural network with hyper-chaos characteristics (HCCNN) was introduced. Zhu [19] proposed a novel image encryption scheme based on improved hyperchaotic sequences. In this algorithm, the hyperchaotic sequences were firstly modified to generate chaotic key stream that is more suitable for image encryption. Then the final encryption key stream was generated by correlating the chaotic key stream with plain-text. Özkaynak et al. [20] argued that this method is not secure enough, and obtained the secret parameters of the cryptosystem by using chosen plain-text attacks. In [21–23], spatial chaos systems were applied for image encryption in order to overcome the drawbacks of small key space and weak security in widely used one-dimensional chaotic system. However, the previously mentioned algorithms are restricted to grayscale images. Though some of them can be easily extended to handle color images, this extension comes with a cost of substantially increased computation time as a result of additional information required to represent color components.

As is well known, the color images can provide more abundant information than the grayscale ones and are frequently used in many areas. So how to develop a secure encryption algorithm for color images has attracted growing attentions in recent years. Patidar et al. [24] designed a fast loss-less symmetric color image cipher based on the widely used substitution-diffusion architecture which utilized chaotic standard and logistic maps. However, the analysis and simulation results in [25] showed that only a pair of (plain-text/cipher-text) was needed to totally break this cryptosystem. In [26], Huang and Nien proposed a color image cryptosystem using multi-chaotic systems, which is composed of two shuffling stages parameterized by chaotically generated sequences. But this scheme cannot resist known-plaintext attack and chosen-plaintext attack [27]. Rhouma et al. [28] have devised an approach for color image encryption based on one-way coupled-map lattices (OCML). An external secret key of 192-bit length was used to generate the initial conditions and parameters of the OCML by making some algebraic transformations to the secret key. Liu and Wang [29] applied a bit-level permutation and high-dimension chaotic map to encrypt color image. Firstly, a plain color image of size $M \times N$ was converted into a grayscale image of size $M \times 3N$ and the grayscale image was transformed into a binary matrix. Then the matrix was permuted at bit-level by the scrambling mapping generated by a piecewise linear chaotic map (PWLCM). Secondly, the chaotic Chen system was employed to confuse and diffuse the red, green and blue components simultaneously. More recently, several schemes combined DNA computing with chaotic systems to encrypt color images [30, 31]. Such experiments can only be done in a well equipped laboratory using current technology, and it needs higher cost. For these reasons, the studies of DNA cryptography are still focusing on affordable methods in terms of practicality.

Due to the finite precision of digital computers, the most serious defect in single chaotic systems is that the chaotic dynamics degrade fast as they are implemented in computers. Different from this, it has been revealed that spatiotemporal chaotic systems maintain much longer periodicity in digitalization and gain excellent performance in cryptography. On the other hand, chaotic attractors have been discovered in fractional-order systems in the past decade [32–37]. Compared to integer-order systems, the fractional-order systems are found to have more complex dynamics because the fractional derivatives have complex geometrical interpretation due to their nonlocal character and high nonlinearity [38]. In addition, the derivative orders can be also used as secret keys as well, which will increase the key space of the cryptosystem. To our best knowledge, there are few encryption techniques using the fractional-order chaotic systems. Therefore, for the purpose of high security, it is very promising to employ CML and fractional-order chaotic systems in color images encryption.

Motivated by the above discussions, in this paper, we propose a new color image cryptosystem using CML and a fractional-order chaotic system. For the purpose of reaching higher security, higher complexity and higher sensitivity, the present work employs an image division-shuffling process which firstly divides the plain-image into four sub-images, and then shuffles the position of pixels in the whole image. This procedure will significantly enhance the resistance of the proposed cryptosystem against known/chosen-plaintext attacks. In order to increase the security of the presented algorithm, an external secret key of 280-bit length is utilized to generate initial conditions and parameters of the CML and the fractional-order chaotic system by making some algebraic transformations to the key so that the proposed encryption scheme is greatly sensitive to changes in even a single bit of the key. Moreover, to further strengthen the security and sensitivity of the cryptosystem, the CML is used to shuffle the positions of pixels totally, and the fractional-order Chen chaotic system and the plain-image are employed to change the values of the pixels. A simultaneous generation of the key streams and the parallel image division-shuffling process can improve the efficiency of the proposed encryption algorithm. Both theoretical analyses and computer simulations verify the feasibility and superiority of the proposed image cryptosystem. In addition, an extensive tolerance analysis of some common image processing operations such as noise addition, cropping, JPEG compression, rotation, brightening and darkening, is performed on the proposed image encryption technique. The experimental results demonstrate that our method is highly robust against some common image processing operations and geometric attacks.

The Proposed Chaotic Cryptosystem for Color Images

The proposed chaos-based cryptosystem for color images consists of the following four parts: i) an image division-shuffling process, ii) a key streams generation process, iii) an image permutation process and iv) an image diffusion process. The flowchart of the image encryption procedure using the proposed scheme is displayed in Fig. 1. Firstly, the plain-image is divided into four sub-images, and then these blocks are shuffled to obtain a disordered image. This process can enhance the resistance of the cipher-image against plaintext attack. Secondly, a 280-bit external secret key is used to generate initial conditions and parameters of the CML and the fractional-order chaotic system. The key streams can be generated by using the obtained initial conditions and parameters to iterate the CML and the fractional-order chaotic system. Thirdly, the positions of the image pixels are permuted by the pseudo-random key stream generated from the CML. In the last stage, the pixel values are modified by the pseudo-random key stream generated from the fractional-order chaotic system. After this, the cipher-image is finally achieved.

0.1 The image division-shuffling process

Without loss of generality, we assume that the size of the color plain-image I is $M \times N$, where M and N are the width and height of the image, respectively. Converting the image I into its red, green and blue components, one can get three color matrices I_R, I_G, I_B with size $M \times N$.

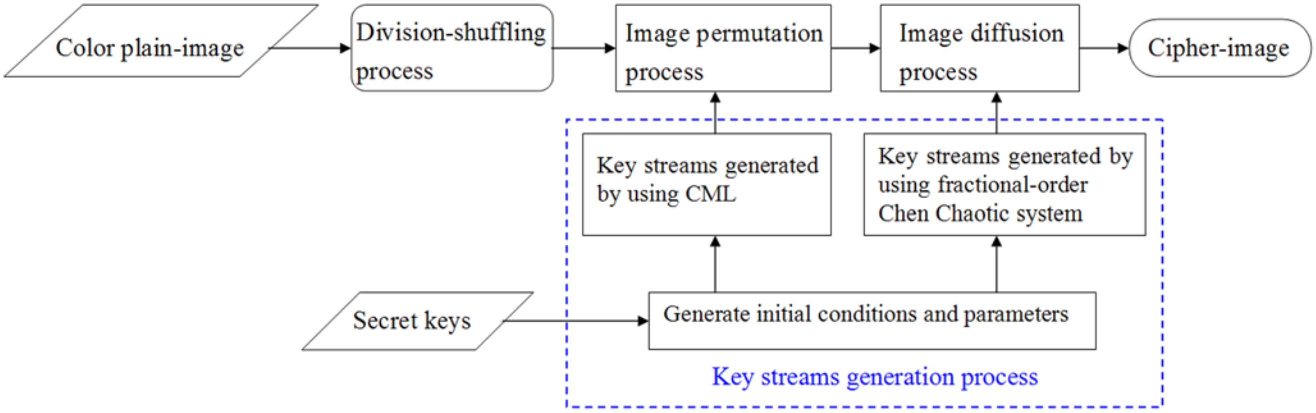
Color plain-image
Division-shuffling process
Image permutation process
Image diffusion process
Cipher-image
Key streams generated by using CML
Key streams generated by using fractional-order Chen Chaotic system
Secret keys
Generate initial conditions and parameters
Key streams generation process

Fig 1. Flowchart of the proposed image encryption algorithm.

doi:10.1371/journal.pone.0119660.g001

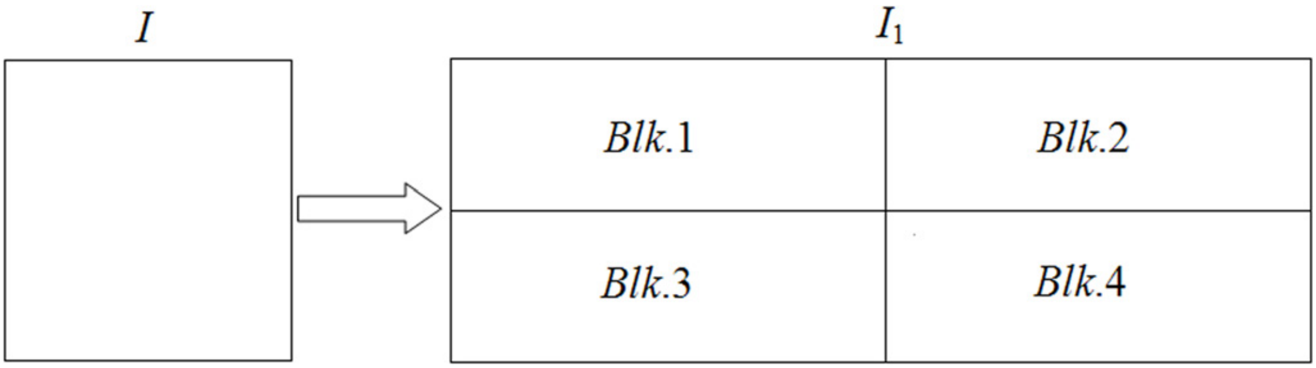

Fig 2. Division of the plain-image.

doi:10.1371/journal.pone.0119660.g002

Then combine the red, green and blue matrices horizontally by the following formula (1) and obtain a matrix I_1 with M rows and $3N$ columns:

$$\begin{cases} I_1(1:M, 1:N) = I_R(1:M, 1:N) \\ I_1(1:M, (N+1):2N) = I_G(1:M, 1:N) \\ I_1(1:M, (2N+1):3N) = I_B(1:M, 1:N) \end{cases} . \tag{1}$$

The image I_1 is decomposed equally into four blocks and each block can be labeled in the form of $Blk.k$, $k = 1, 2, 3, 4$, as shown in Fig. 2. The size of each block are given as follows: size (Blk. 1) = $\lfloor M/2 \rfloor \times \lfloor 3N/2 \rfloor$, size(Blk.2) = $\lfloor M/2 \rfloor \times (3N- \lfloor 3N/2 \rfloor)$, size($Blk$.3) = $(M- \lfloor M/2 \rfloor) \times \lfloor 3N/2 \rfloor$ and size(Blk.4) = $(M- \lfloor M/2 \rfloor) \times (3N- \lfloor 3N/2 \rfloor)$. If M is an odd number, append the first row of Blk.4 to the end of Blk.2 and delete the first row of Blk.4. Then the sizes of blocks Blk.2 and Blk.4 are obtained as follows: size(Blk.2) = $(M- \lfloor M/2 \rfloor) \times (3N- \lfloor 3N/2 \rfloor)$ and size (Blk.4) = $\lfloor M/2 \rfloor \times (3N- \lfloor 3N/2 \rfloor)$. And the sizes of the other two blocks keep unchanged. In the following, we will further shuffle the plain-image, which makes the encryption operation more confusing and complex as it adds one extra step to the encryption process.

We firstly create three empty matrices, i.e., P_1 with size $\lfloor M/2 \rfloor \times 3N$, P_2 with size $(M- \lfloor M/2 \rfloor) \times 3N$ and I_2 with size $M \times 3N$. Then insert each column of Blk.4 in turn into the odd columns of matrix P_1 and insert each column of Blk.1 sequentially into the even columns of matrix P_1 as illustrated in Fig. 3. Similarly, insert each column of Blk.2 in turn into the odd columns of matrix P_2 and insert each column of Blk.3 sequentially into the even columns of matrix P_2. Finally, insert each row of P_2 in turn into the odd rows of matrix I_2 and insert each row of P_1 sequentially into the even rows of matrix I_2. Thus a disordered image matrix I_2 with size $M \times 3N$ is obtained.

Implementing the above procedure, we apply the first complexity to our encryption approach, which significantly enhance the resistance against known/chosen-plaintext attacks.

0.2 CML and the fractional-order Chen chaotic system

The cryptosystems based on widely used one-dimensional discrete chaotic maps suffer from fundamental drawbacks such as small key space, slow performance speed and weak security function [39]. To overcome these limitations in the proposed encryption scheme, a 2D coupled map lattice (CML) and the fractional-order Chen chaotic system are utilized to generate the key streams.

A 2D CML was introduced by Kaneko and Tsuda [40] as a simple model capturing essential features of spatiotemporal dynamics of extended nonlinear systems and later was employed for

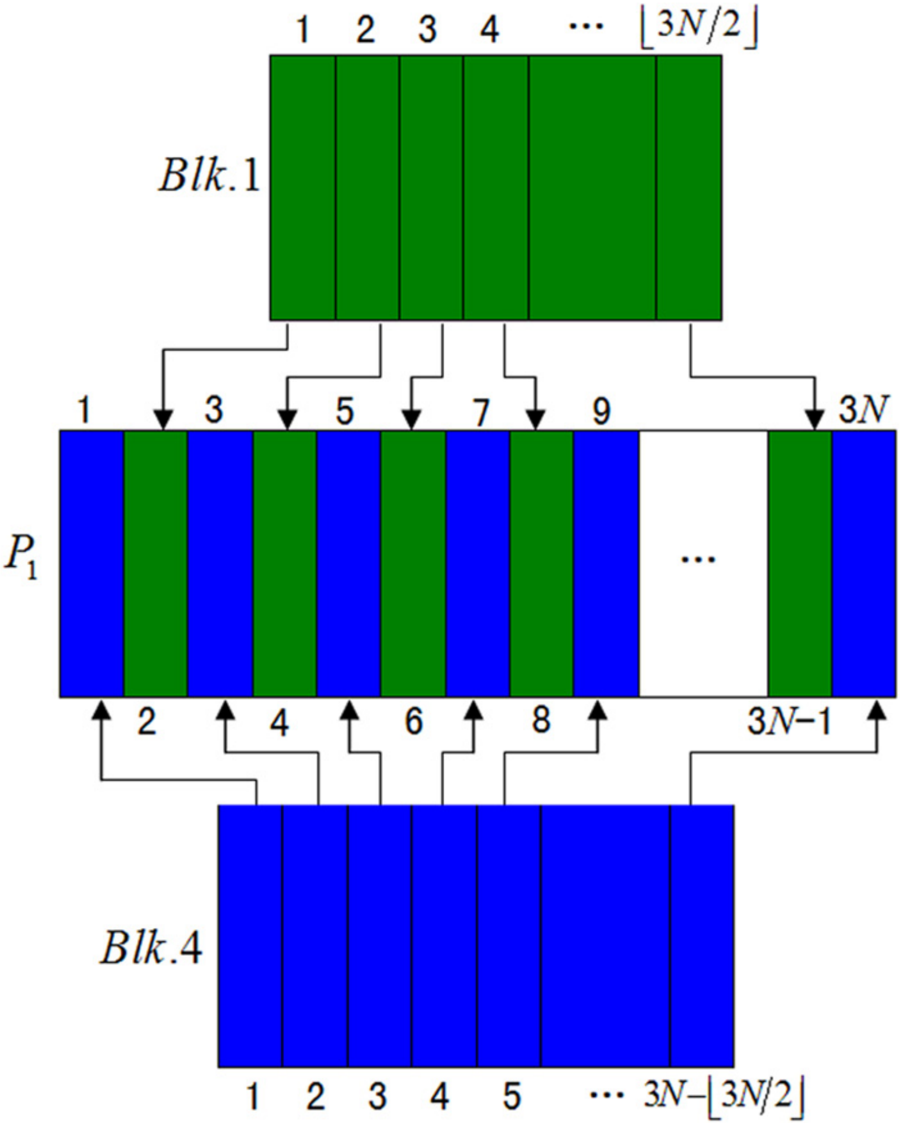

Fig 3. Shuffling of the image blocks *Blk*. 1 and *Blk*. 4.

doi:10.1371/journal.pone.0119660.g003

modeling complex spatial phenomena in diverse areas of science and engineering. Recently, CML was introduced for cryptography of a self-synchronizing stream cipher. The 2D CML is defined as follows:

$$x_{n+1}(k) = (1-\varepsilon)f(x_n(k)) + \frac{\varepsilon}{2}[f(x_n(k-1)) + f(x_n(k+1))], \tag{2}$$

where $f(x)$ is the mapping function, n is the discrete time index, $n = 0, 1, 2.. . .,L-1$, with L being the system size, $k = 1, 2, . . .,S$ is the lattice site index, and $\varepsilon \in (0, 1)$ is the coupling constant. In general, periodic boundary conditions $x_n(k+L) = x_n(k)$ are assumed, and $f(x) = 1-\mu x^2$ is chosen where μ is a constant parameter and $\mu \in (0, 2)$. The chaotic behavior of 2D CML (2) is demonstrated in Fig. 4(a).

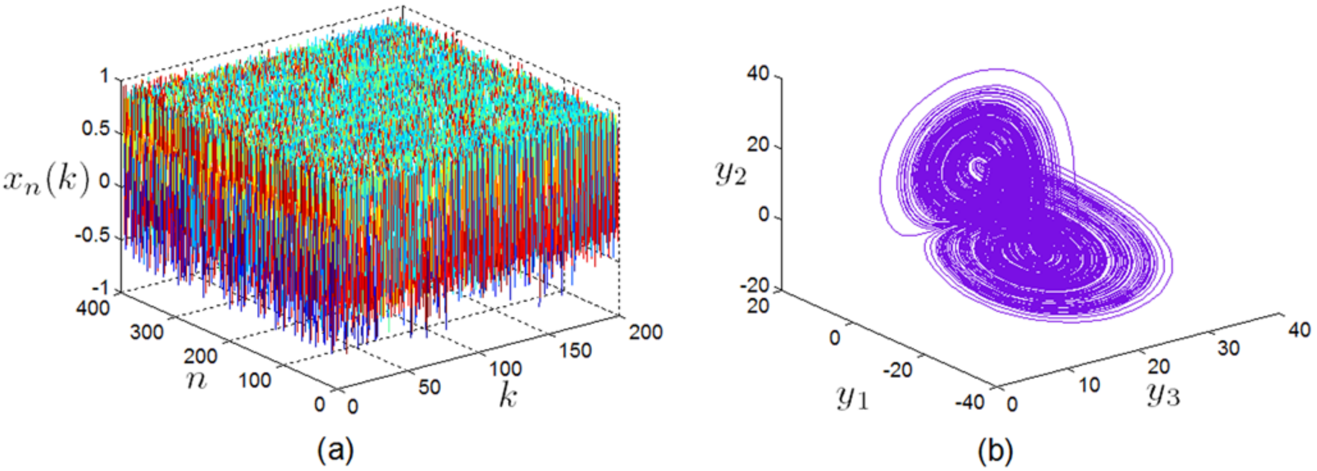

Fig 4. The chaotic behaviors of systems (2) and (3). (a) The spatiotemporal attractor of the 2D CML (2) with $\mu = 1.8$ and $\varepsilon = 0.1$, (b) the chaotic attractor of the fractional-order Chen system (3) with $\alpha_1 = \alpha_2 = \alpha_3 = 0.9$.

doi:10.1371/journal.pone.0119660.g004

Another chaotic system in our scheme is the fractional-order Chen chaotic system, which is described by

$$\begin{cases} D_*^{\alpha_1} y_1 = a(y_2 - y_1) \\ D_*^{\alpha_2} y_2 = -by_1 - y_1 y_3 + cy_2\,, \\ D_*^{\alpha_3} y_3 = y_1 y_2 - dy_3 \end{cases} \tag{3}$$

where a, b, c and d are positive system parameters. When $a = 35$, $b = 7$, $c = 28$, $d = 3$, and $\alpha_j \in [0.8,1]$ ($j = 1, 2, 3$), the fractional-order Chen system behave chaotically [41], as displayed in Fig. 4(b).

0.3 Generation of the initial conditions and parameters

In the proposed scheme, let $L = M \times N$ and $S = 3$ for the CML, i.e., $n = 0, 1, 2, \ldots, M \times N - 1$, $k = 1, 2, 3$. In view of the basic need of cryptology, the cipher-text should have a close correlation with the key. There are two ways to accomplish this requirement: one is to mix the key thoroughly into the plain-text through the encryption process, another is to use a good key generation mechanism. Here we use a 280-bit long external secret key (K), which is divided into blocks (K_i) of 8-bit, to derive the parameters ε, μ, a_j ($j = 1, 2, 3$) and the initial conditions x_0 (1), x_0 (2), x_0 (3), y_1 (0), y_2 (0) and y_3 (0) in systems (2) and (3). The 280-bit long secret key (K) is given by:

$$K = K_1 K_2 K_3 \cdots K_{35}. \tag{4}$$

The initial conditions and parameters are obtained as follows:

$$x_0(1) = \left(\left((K_1|K_3) \oplus (K_5|K_7) + \sum_{i=1}^{35} K_i\right)/257\right) \bmod 1, \tag{5}$$

$$x_0(2) = \left(\left((K_2|K_4) \oplus (K_6|K_9) + \sum_{i=1}^{35} K_i\right)/257\right) \bmod 1, \tag{6}$$

$$x_0(3) = \left(\left((K_8|K_{11}) \oplus (K_{13}|K_{15}) + \sum_{i=1}^{35} K_i\right)/257\right) \bmod 1, \tag{7}$$

$$y_1(0) = \left(\left((K_{10}\&K_{12}) \oplus (K_{14}\&K_{16}) + \sum_{i=1}^{35} K_i\right)/257\right) \bmod 3, \tag{8}$$

$$y_2(0) = \left(\left((K_{17}\&K_{19}) \oplus (K_{21}\&K_{23}) + \sum_{i=1}^{35} K_i\right)/257\right) \bmod 4, \tag{9}$$

$$y_3(0) = \left(\left((K_{18}\&K_{20}) \oplus (K_{22}\&K_{24}) + \sum_{i=1}^{35} K_i\right)/257\right) \bmod 5, \tag{10}$$

$$\varepsilon = \left(\left(\left(K_{25} \oplus K_{27} \oplus K_{29} + \sum_{i=1}^{35} K_i\right)/257\right) \bmod 1\right)/10, \tag{11}$$

$$\mu = 1.8 + \left(\left(\left(K_{26} \oplus K_{28} \oplus K_{30} + \sum_{i=1}^{35} K_i\right)/257\right) \bmod 1\right)/10, \tag{12}$$

$$\alpha_1 = 0.92 + \left(\left(\left(K_{31} \oplus K_{33} \oplus K_{35} + \sum_{i=1}^{35} K_i\right)/257\right) \bmod 1\right)/15, \tag{13}$$

$$\alpha_2 = 0.92 + \left(\left(\left(K_{30} \oplus K_{32} \oplus K_{34} + \sum_{i=1}^{35} K_i\right)/257\right) \bmod 1\right)/15, \tag{14}$$

$$\alpha_3 = 0.92 + \left(\left(\left(K_{10} \oplus K_{15} \oplus K_{25} + \sum_{i=1}^{35} K_i\right)/257\right) \bmod 1\right)/15, \tag{15}$$

where the | operator is the bitwise OR; the & operator is the bitwise AND; $\oplus$ is the bitwise XOR operator.

Clearly, from Equations (5)–(15), one can see that the initial conditions and parameters of the CML and the fractional-order Chen system are greatly sensitive to changes in even a single bit of the 280-bit secret key. Therefore, the proposed cryptosystem with a key space of 2^{280} can resist any brute-force attack.

0.4 Image permutation based on CML

Image data have strong correlations among adjacent pixels in horizontal, vertical, and also diagonal directions for both natural and computer-graphical images. In order to weaken the strong relationship among adjacent pixels, a CML is used to scramble the pixel positions of the image I_2.

To permute the positions of image pixels, we perform the following 11 steps:

Step 1. Use the initial conditions $x_0(1)$, $x_0(2)$, $x_0(3)$ and the parameters ε, μ obtained in Section 0.3 to iterate Equation (2) for $l_1 + MN$ times, and discard the former l_1 state values to get rid of harmful effects. One can get three chaotic sequences, i.e.,

$\mathrm{D}_1 = \{x_{l_1+1}(1), x_{l_1+2}(1), \cdots, x_{l_1+MN}(1)\}, \mathrm{D}_2 = \{x_{l_1+1}(2), x_{l_1+2}(2), \cdots, x_{l_1+MN}(2)\}$ and $\mathrm{D}_3 = \{x_{l_1+1}(3), x_{l_1+2}(3), \cdots, x_{l_1+MN}(3)\}$.

Step 2. Preprocess the above three chaotic sequences by the following formula (16):

$$x'_{l_1+i}(j) = 10^3 \times x_{l_1+i}(j) - \mathrm{Round}(10^3 \times x_{l_1+i}(j)), \tag{16}$$

where $x_{l_1+i}(j) \in \mathrm{D}_j$, $x'_{l_1+i}(j) \in \mathrm{D}'_j$, $i = 1, 2, \ldots, MN$, $j = 1, 2, 3$, and Round(x) represents the integer function which rounds the real number x to the nearest integer. Thus we get three chaotic analogue sequences D'_1, D'_2 and D'_3 with very well expressed random-like properties. Further, transform the vectors D'_j into the matrices MD_j with size $M \times N$, where the value in row ($\mathbf{mod}(i-1, M) + 1$) and column $\lceil i/M \rceil$ of MD, denotes the i-th value of D'_j, i.e., $\mathrm{MD}_j((\mathbf{mod}(i-1, M)+1), \lceil i/M \rceil) = x'_{l_1+i}(j)$, $i = 1, 2, \ldots, MN$, $j = 1, 2, 3$.

Step 3. Divide the image matrix I_2 equally into three matrices from left to right, i.e., $I_{21} = I_2(1{:}\ M, 1{:}\ N)$, $I_{22} = I_2(1{:}\ M, (N+1){:}\ 2N)$ and $I_{23} = I_2(1{:}\ M, (2N+1){:}\ 3N)$. Transform the matrices $I_{21,}$ $I_{22,}$ and I_{23} into three one-dimensional vectors VP_1, VP_2 and VP_3 with length MN, respectively. For example, $VP_1 = \{g_{1,1}, g_{1,\ 2}, \ldots, g_{1,MN}\}^{\mathrm{T}}$ where $g_{1,i}$ denotes the value of the image pixel in row ($\mathbf{mod}(i - 1, M) + 1$) column $\lceil i/M \rceil$ of I_{21}, i.e., $g_{1,i} = I_{21}((\mathbf{mod}(i - 1, M) + 1), \lceil i/M \rceil)$, $i = 1, 2, \ldots, MN$. Sort MN values in VP_{i} and attain the sorted vectors $SP_j = \{\bar{g}_{j,1}, \bar{g}_{j,2}, \cdots, \bar{g}_{j,MN}\}^{\mathrm{T}} (j = 1, 2, 3)$. Find the positions of values $\{\bar{g}_{j,1}, \bar{g}_{j,2}, \cdots, \bar{g}_{j,MN}\}^{\mathrm{T}}$ in $\{g_{\mathrm{i},1}, g_{\mathrm{i},2,} \ldots, g_{\mathrm{i},MN}\}^{\mathrm{T}}$ and mark the transformation positions $TP_{\mathrm{j}} = \{p_{\mathrm{i},1}, p_{\mathrm{i},2,} \ldots, p_{\mathrm{i},MN}\}^{\mathrm{T}}$ where $\bar{g}_{j,i}$ is exactly the value of $g_{j,p_{ii}}$.

Step 4. Shuffle the values in SP_j by TP_j ($j = 1, 2, 3$), getting $SP'_j = \{\bar{g}'_{j,1}, \bar{g}'_{j,2}, \cdots, \bar{g}'_{j,MN}\}^{\mathrm{T}}$ where $\bar{g}'_{j,i} = \bar{g}_{j,p_{j,i}}$. Then transform the three one-dimensional vectors SP'_1, SP'_2 and SP'_3 into the three image matrices I_{31}, I_{32} and I_{33} with size $M \times N$, respectively, where the pixel value of $I_{3,j}$ in row ($\mathbf{mod}(i - 1, M) + 1$) and column $\lceil i/M \rceil$ is equal to the i-th value of SP'_j, i.e., $I_{3,j}((\mathbf{mod}(i-1, M)+1), \lceil i/M \rceil) = \bar{g}'_{j,i}$, $i = 1, 2, \ldots, MN$.

Step 5. Repeat Steps 6 to 11 n rounds ($n \leq \min(M, N)$).

Step 6. Let $i \leftarrow 1$;

Step 7. Sort N values of the i-th row of MD_j and obtain the transform positions $TM_{j,i} = \{pm_{j,i}(1), pm_{j,i}(2), \ldots pm_{j,i}(N),\}$ ($j = 1, 2, 3$). Scramble the pixel positions of the i-th row of image $I_{3,j}$ according to $TM_{j,i}$, i.e., move the $pm_{j,i}(1)$ column of the i-th row to the first column, the $pm_{j,i}(2)$ column of the i-th row to the second column etc., until all columns have been moved; thus a new column transformation of the i-th row is generated.

Step 8. Let $i \leftarrow i + 1$, return to Step 7 until i reaches M. Thus we get three row-permuted matrices I_{41}, I_{42} and I_{43}.

Step 9. Let $i \leftarrow 1$;

Step 10. Sort M values of the i-th column of MD_j and obtain the transform positions $TD_{j,i} = \{pd_{j,i}(1), pd_{j,i}(2), \ldots, pd_{j,i}(N)\}^{\mathrm{T}}$ ($j = 1, 2, 3$). Shuffle the pixel positions of the i-th column of image I_{4j} according to $TD_{j,i}$, i.e., move the $pd_{j,i}(1)$ row of the i-th column to the first row, the $pd_{j,i}(2)$ row of the i-th column to the second row etc., until all rows have been moved; thus a new row transformation of the i-th column is obtained.

Step 11. Let $i \leftarrow i + 1$, return to Step 10 until i reaches N. Thus we obtain three new total shuffled matrices $I_{51,}$ I_{52} and I_{53}, which are respectively the R, G and B color matrices of a new permutation image denoted by I_5.

Obviously, the above permutation-only process just rearranges the pixel positions without changing the pixel's value. Different from conventional block encryption methods such as DES and AES, the proposed algorithm shuffles the positions of image pixels totally by using the transform positions. Compared with the permutations based on one-dimensional or two-dimensional chaotic maps, the permutation scheme proposed overcomes the drawback of short periodicity, since the relationship between the original and shuffled position of one pixel is not directly related to the chaotic map. However, there are still some potential weak points in these permutation-only algorithms [42], which are weak against statistical attack and known-text attack. To deal with the weakness of pure position permutation method, in the following, a diffusion process is further employed to modify the pixel's gray value to enhance the security of the encryption algorithm.

0.5 Image diffusion based on the fractional-order Chen chaotic system

The encryption algorithm proposed in this paper is based on a permutation-diffusion architecture. In the diffusion stage, the fractional-order Chen chaotic system is employed to generate the key stream for diffusion, and the pixel values are modified sequentially to confuse the relationship between the cipher-image and the plain-image. In some existing chaos-based image ciphers, the key stream used in the diffusion process is solely determined by the key. The same key stream is applied to encrypt different plain-images if the key remains unchanged. An opponent may derive the key stream by the plain-text attack, i.e., by ciphering some special plain-text sequences and then comparing them with the corresponding cipher-text sequences. In order to make the cryptosystem secure against a differential attack, the modification made to a particular pixel depends not only on the corresponding key stream element, but also on the accumulated effect of all previous pixel values. The diffusion process is decomposed into the following 5 steps:

Step 1. Arrange the pixels of permuted color matrices $I_{51,}$ I_{52} and I_{53} obtained in Section 0.3 from left to right and then from top to bottom, respectively, we get three one-dimensional vectors $PR = \{pr_1, pr_2, \ldots, pr_{MN},\}$, $PG = \{pg_1, pg_2, \ldots, pg_{MN},\}$ and $PB = \{pb_1, pb_2, \ldots, pb_{MN},\}$.

Step 2. Use the initial conditions $y_1(0)$, $y_2(0)$, $y_3(0)$, and the fractional orders α_j ($j = 1, 2, 3$) generated in Section 0.3 to iterate Equation (3) for $l_2 + MN$ times, and discard the former l_2 state values to avoid transient effects. We then obtain three chaotic sequences, i.e., $\mathrm{L}_1 = \{y_1(l_2{+}1), y_1(l_2{+}2), \ldots, y_1(l_2{+}MN),\}$, $\mathrm{L}_2 = \{y_2(l_2{+}1), y_2(l_2{+}2), \ldots, y_2(l_2{+}MN),\}$ and $\mathrm{L}_3 = \{y_3(l_2{+}1), y_3(l_2{+}2), \ldots, y_3(l_2{+}MN)\}$.

Step 3. To get the key streams, preprocess the above three chaotic sequences according to the following formula (17):

$$y'_j(l_2 + i) = \mathrm{Round}(\mathrm{Abs}(y_j(l_2 + i) - \mathrm{Fix}(y_j(l_2 + i)))) \times 10^{14} \mathrm{mod}\ 256, \tag{17}$$

where $y_j(l_2 + i) \in \mathrm{L}_j$, $y_j'(l_2 + i) \in \mathrm{L}_j'$, $i = 1, 2, \ldots, MN$, $j = 1, 2, 3$, and Round(x) represents the integer function which has the same meaning as that in Equation (16), the function Abs(x) returns the absolute value of x and the function Fix(x) returns the value of x to the nearest integer towards zero. Thus we obtain three key streams L_1', L_2' and L_3'.

Step 4. Calculate the corresponding pixel data of the cipher-image by using the values of the currently operated pixel and the previously operated pixels, according to the following formula:

$$C\_R(i) = ((PR(i) \oplus \mathrm{L}_1'(i)) \oplus (C\_R(i-1) \oplus \mathrm{L}_1'(i-1))) \oplus (C\_G(i-1) \oplus C\_B(i-1)), \tag{18}$$

$$C\_G(i) = ((PG(i) \oplus \mathrm{L}'_2(i)) \oplus (C\_G(i-1) \oplus \mathrm{L}'_2(i-1))) \oplus (C\_R(i-1) \oplus C\_B(i-1)), \tag{19}$$

$$C\_B(i) = ((PB(i) \oplus \mathrm{L}'_3(i)) \oplus (C\_B(i-1) \oplus \mathrm{L}'_3(i-1))) \oplus (C\_R(i-1) \oplus C\_G(i-1)), \tag{20}$$

where $i = 1, 2, \ldots, MN$, $\oplus$ is the bitwise XOR operator, $PR(i)$, $PG(i)$ and $PB(i)$ are the current plain pixel values, $C\_R(i)$, $C\_G(i)$ and $C\_B(i)$ are the current cipher pixel values, $\mathrm{L}'_1(i)$, $\mathrm{L}'_2(i)$ and $\mathrm{L}'_3(i)$ are the current key stream elements, $C\_R(i-1)$, $C\_G(i-1)$ and $C\_B(i-1)$ are the previous cipher pixel values, and $L'_1(i-1)$, $L'_2(i-1)$ and $L'_3(i-1)$ are the previous key stream elements. The first cipher pixel values *C_R*(*1*), *C_G*(*1*) and *C_B*(*1*) are set as follows:

$$C\_R(1) = ((PR(1) \oplus L'_1(1)) \oplus (PR(MN) \oplus L'_1(MN))) \oplus (PG(MN) \oplus PB(MN)), \tag{21}$$

$$C\_G(1) = ((PG(1) \oplus L'_2(1)) \oplus (PG(MN) \oplus L'_2(MN))) \oplus (PR(MN) \oplus PB(MN)), \tag{22}$$

$$C\_B(1) = ((PB(1) \oplus L'_3(1)) \oplus (PB(MN) \oplus L'_3(MN))) \oplus (PR(MN) \oplus PG(MN)). \tag{23}$$

Step 5. Transform three encrypted vectors *C_R*, *C_G* and *C_B* with length *MN* into three matrices with size *M* × *N*, i.e., *CR*, *CG* and *CB*, respectively, which are the *R*, *G*, *B* components of the ciphered image *C*. Thus we finally obtain the encrypted image.

Since the CML and the fractional-order chaotic system have a nonlinear structure and more complex dynamics than low-dimensional ones, the proposed chaos-based cryptosystem is greatly sensitive to a change in even a single bit of the 280-bit long secret key. Moreover, using the division-shuffling process and the permutation-diffusion process, a slight change of plain-image pixel causes a significant change in the cipher-image, which makes a differential analysis inefficient and practically useless. These features will in turn strengthen the security and sensitivity of the cryptosystem. In addition, the generation of pseudo-random numbers by the CML and the fractional-order Chen chaotic system and the division-shuffling process on the plain-image can be carried out simultaneously, i.e., in a parallel manner, which promotes the speed performance of the proposed image encryption algorithm. Therefore, our proposed scheme has higher security and overcomes the limitations in the image cryptosystem based on one-dimensional or two-dimensional chaotic maps.

0.6 Design of image decryption algorithm

Because the presented color image encryption algorithm is a symmetric cryptosystem, the decryption procedure is similar to that of the encryption process but just in the reversed order. However, some remarks should be considered in the decryption process, which are summarized as follows:

Remark 1. We can rewrite Equations (18)–(23) to give the pixel values in the RGB components:

$$PR(i) = ((C\_R(i) \oplus L'_1(i)) \oplus (C\_R(i-1) \oplus L'_1(i-1))) \oplus (C\_G(i-1) \oplus C\_B(i-1)), \tag{24}$$

$$PG(i) = ((C\_G(i) \oplus L'_2(i)) \oplus (C\_G(i-1) \oplus L'_2(i-1))) \oplus (C\_R(i-1) \oplus C\_B(i-1)), \tag{25}$$

$$PB(i) = ((C\_B(i) \oplus L'_3(i)) \oplus (C\_B(i-1) \oplus L'_3(i-1))) \oplus (C\_R(i-1) \oplus C\_G(i-1)), \tag{26}$$

$$PR(1) = ((C\_R(1) \oplus L'_1(1)) \oplus (PR(MN) \oplus L'_1(MN))) \oplus (PG(MN) \oplus PB(MN)), \tag{27}$$

$$PG(1) = ((C\_G(1) \oplus L'_2(1)) \oplus (PG(MN) \oplus L'_2(MN))) \oplus (PR(MN) \oplus PB(MN)), \tag{28}$$

$$PB(1) = ((C\_B(1) \oplus L'_3(1)) \oplus (PB(MN) \oplus L'_3(MN))) \oplus (PR(MN) \oplus PG(MN)). \tag{29}$$

where *i* = *MN*,(*MN* − *1*), (*MN* − *2*),. . .,3,2.

Remark 2. Perform the reverse operations to remove the effect of permutation. All operations are the same as steps 3–11 in the image permutation process.

Remark 3. Use the same method in the image division-shuffling process but in the reversed order to recover the original color image.

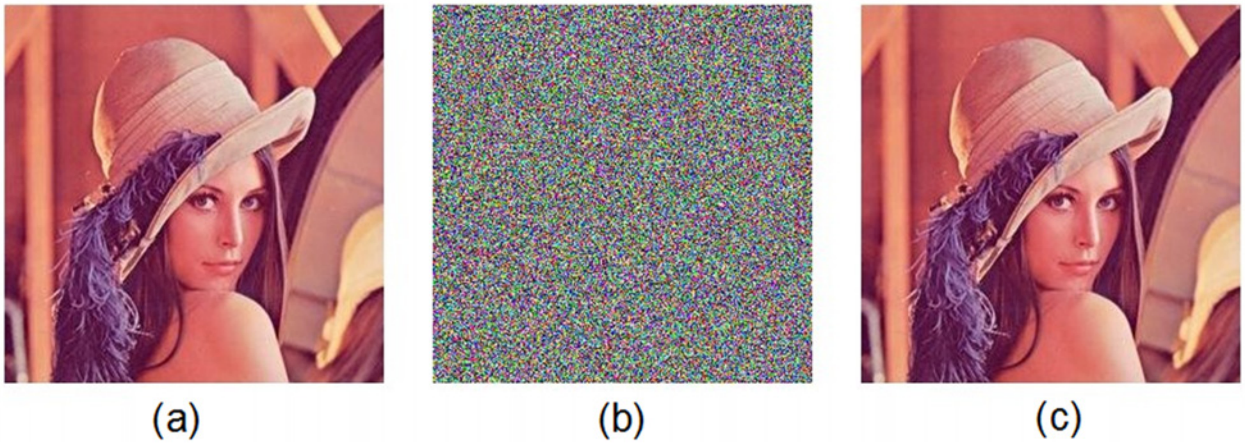

Fig 5. Experimental results for Lena image. (a) Original image of Lena, (b) encrypted image of Lena, (c) decrypted image of Lena.

doi:10.1371/journal.pone.0119660.g005

Remark 4. Note that since the decryption process requires the same key streams for decrypting the cipher-image, the same 280-bit long external secret key $K = K_1K_2K_3...K_{35}$ should be applied for decryption. Hence, according to Section 0.2, it is possible to set the same initial conditions $x_0(1)$, $x_0(2)$, $x_0(3)$, $y_1(0)$, $y_2(0)$, and $y_3(0)$, and the parameters ε, μ, α_j ($j = 1, 2, 3$).

Figs. 5 and 6 show the encryption and decryption of two color images of Lena and Vegetables with size 256 × 256, respectively, where the secret key is chosen as "2eea2814e6087660406d59f82f740bfd9c2e5e463d96bdafe482c8054f457bab5cd180" (in hexadecimal). Throughout this paper, the original image of Lena is freely available at the USC-SIPI image database [43].

Performance and Security Analysis

In this section, the performance of the proposed image cryptosystem is analyzed by using different security test measures. These measures are taken as follows: key space analysis, statistical analysis including histogram analysis and computing the correlation coefficients of adjacent pixels, information entropy analysis, test security against differential attack including

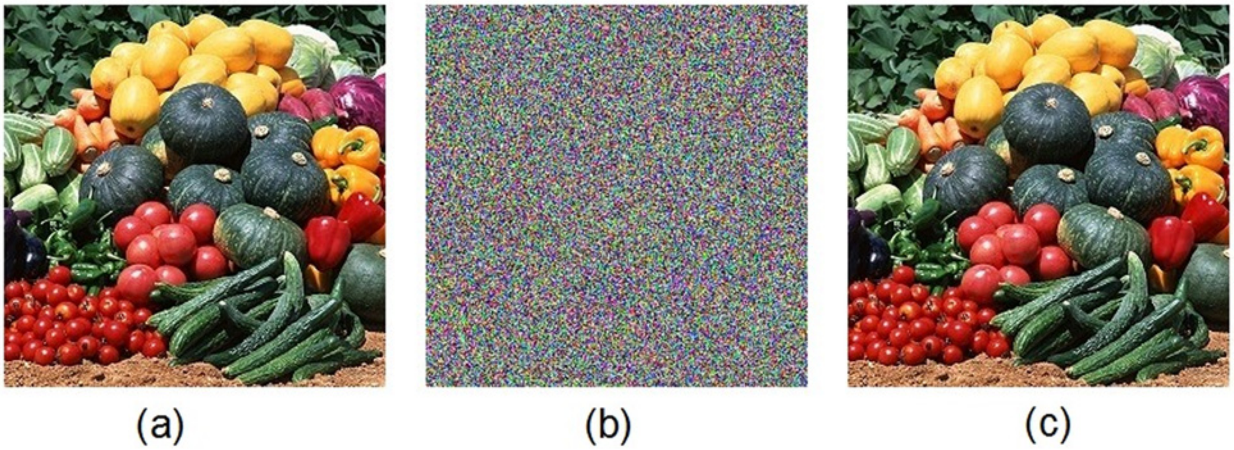

Fig 6. Experimental results for Vegetables image. (a) Original image of Vegetables, (b) encrypted image of Vegetables, (c) decrypted image of Vegetables.

doi:10.1371/journal.pone.0119660.g006

calculating the number of pixel change rate (NPCR) and unified average changing intensity (UACI), and key sensitivity analysis.

1.1 Key space analysis

The size of the key space is the total number of different keys that can be applied in the encryption/decryption process. The key space should be large enough to make brute-force attacks infeasible. From the cryptographic point of view, the size of the key space should not be smaller than 2^{100} to ensure a high level of security [44]. Since the secret key of the proposed scheme is 280-bit long, the key space is 2^{280}, which is sufficiently large enough to resist a brute-force attack.

1.2 Statistical analysis

Shannon suggested that diffusion and confusion should be employed in a cryptosystem [8] for the purpose of frustrating a powerful statistical analysis. An ideal cipher should be robust against any statistical attack. In order to demonstrate the robustness of the proposed image encryption scheme, we have performed some statistical tests on the histograms of the ciphered images and on the correlations of adjacent pixels in the ciphered image.

A. Histogram analysis. Image histogram is a significant feature in image analysis. Indeed, one can see the frequency of each gray level from the histograms, which can leak image information. For a good encryption algorithm, the distribution of cipher-text should hide the redundancy of plain-text and not leak any information about the plain-text or the relationship between the plain-text and the cipher-text.

Figs. 7 and 8 display the histograms of the color plain-image "Lena" (Fig. 5(a)) and the corresponding cipher-image (Fig. 5(b)), respectively. From these figures, one can clearly see that the histograms of the encrypted image are fairly uniform and significantly different from those of the plain-image. The statistical feature of the plain-image is enhanced in such a manner that the cipher-image has a uniform gray level distribution and good balance property. Hence it does not offer any clue to be used in a statistical analysis attack on the encrypted image.

B. Correlation analysis of two adjacent pixels. It is well known that the adjacent pixels of the plain images are highly correlated either in horizontal, vertical or diagonal directions. An efficient image encryption algorithm should decrease the correlation of two adjacent pixels in the ciphered image as low as possible.

In the following, the correlation between all pairs of two horizontally adjacent pixels, all pairs of two vertically adjacent pixels, and 2000 pairs randomly selected of two diagonally adjacent pixels in the plain-image and cipher-image is tested. The correlation coefficients between two adjacent pixels in an image are calculated using the following equations:

$$r_{xy} = \frac{\mathrm{cov}(x, y)}{\sqrt{D(x)D(y)}}, \tag{30}$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i, \tag{31}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))^2, \tag{32}$$

$$\mathrm{cov}(x, y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)), \tag{33}$$

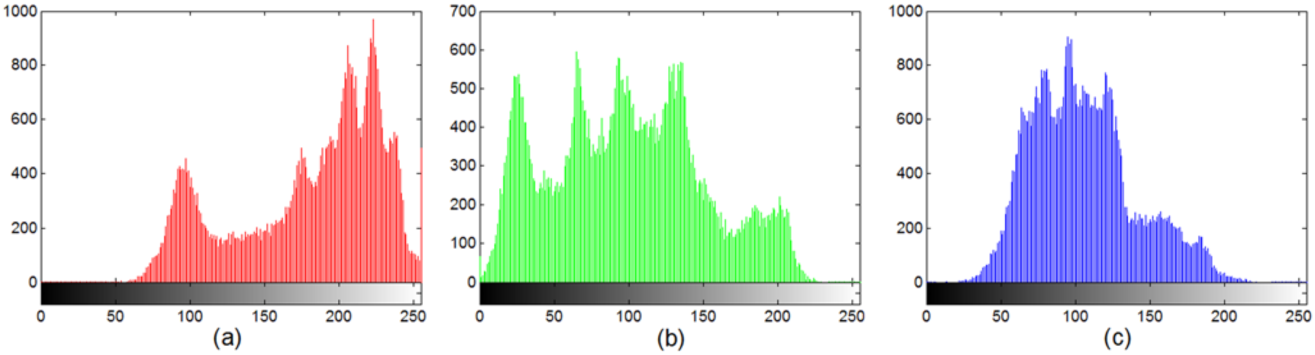

Fig 7. Histogram of the original image of Lena in the (a) red, (b) green, (c) blue, components.

doi:10.1371/journal.pone.0119660.g007

where x and y are gray level values of two adjacent pixels in the image, N is the total number of pixels selected from the image, $E(x)$ and $E(y)$ are the mean values of x_i and y_i, respectively.

Fig. 9 shows the vertical relevance of adjacent pixels in the plain-image of Lena (Fig. 5(a)) and the encrypted one (Fig. 5(b)). The detailed results of the correlation coefficients for two horizontally (vertically and diagonally) adjacent pixels in the red, green and blue components of the original plain-image and the encrypted one are given in Table 1. These results clearly show that the correlation coefficients of the plain-image are close to 1, while those of the cipher-image are nearly 0 and the distribution of adjacent pixels is fairly uniform. It indicates that the proposed algorithm has successfully reduced the correlation of adjacent pixels in the plain-image so that neighboring pixels in the cipher-image virtually have no correlation. So the proposed algorithm can resist statistical attacks. Furthermore, the comparison performed in Table 2 demonstrates that the proposed scheme in this paper is superior to other methods reported in the literature. The cipher-image using our proposed algorithm has the highest performance in the horizontal, vertical and diagonal directions.

1.3 Information entropy analysis

There are various kinds of entropy, such as K-S entropy, K2 entropy, Tsallis entropy and information entropy *etc*. Information entropy, also called Shannon entropy, which was introduced by Shannon in 1948, is described by

$$H(s) = \sum_{i=0}^{2^N-1} p(s_i)\log_2 \frac{1}{p(s_i)}, \tag{34}$$

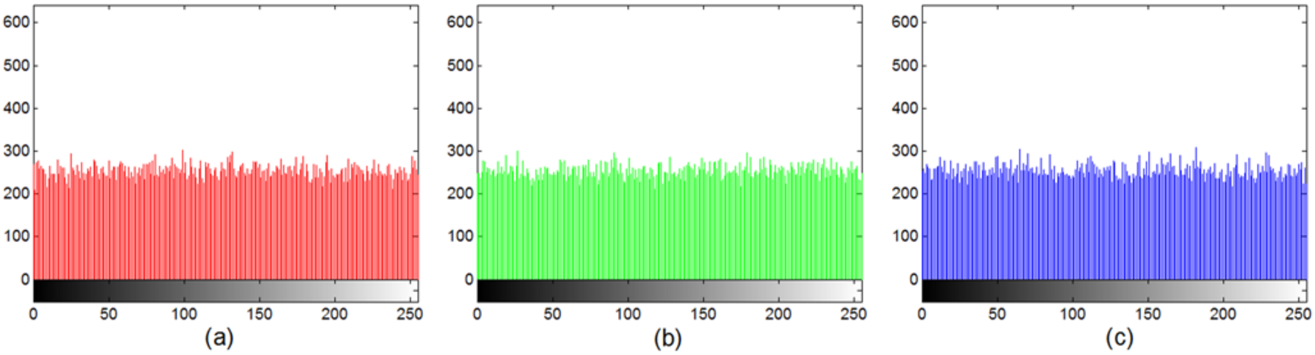

Fig 8. Histogram of the encrypted image of Lena in the (a) red, (b) green, (c) blue, components.

doi:10.1371/journal.pone.0119660.g008

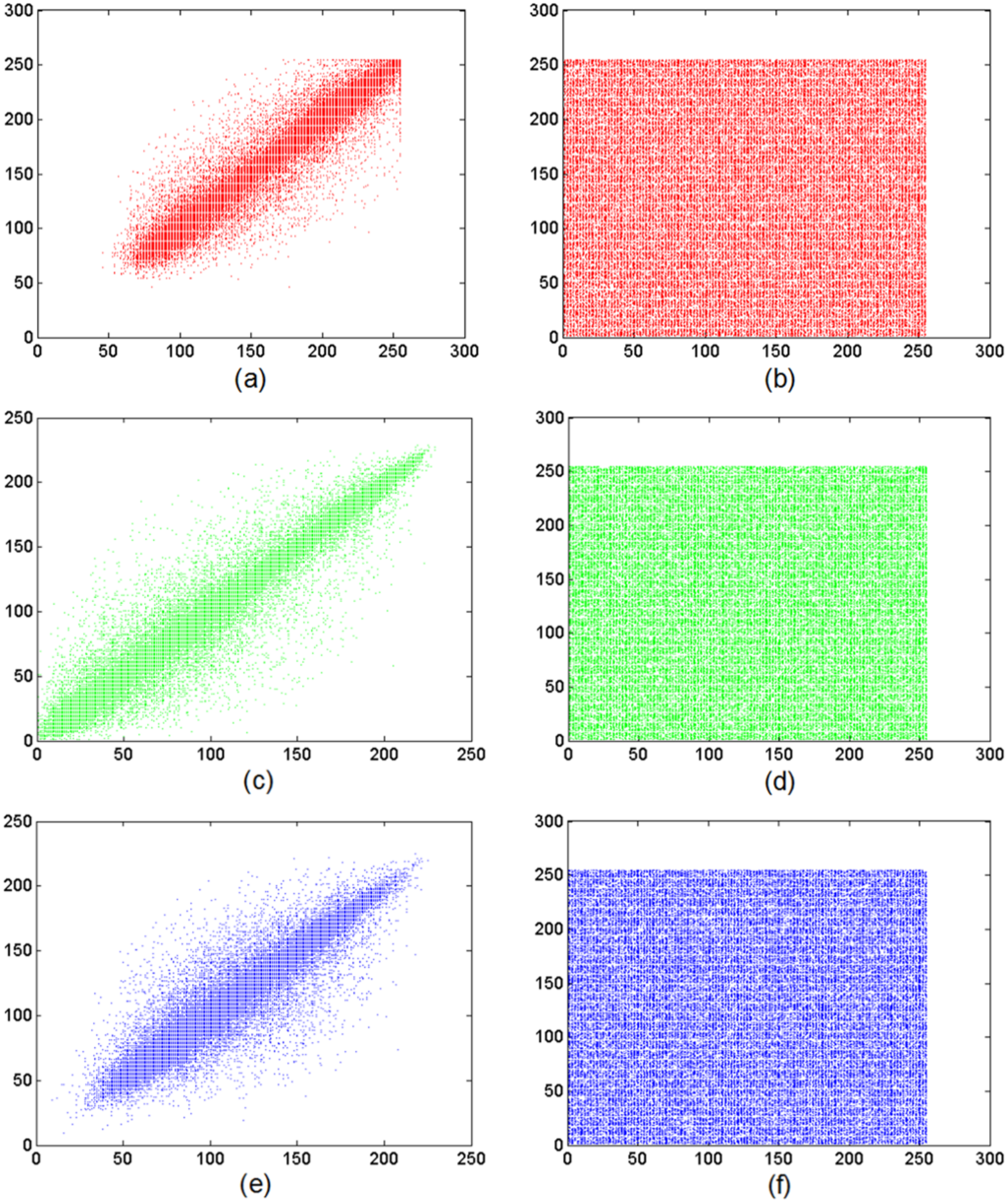

Fig 9. Vertical direction correlations of two adjacent pixels. Frames (a), (c) and (e) show the distribution of two vertically adjacent pixels in the plain-image of Lena in the (a) red, (b) green and (c) blue components, respectively. Frames (b), (d) and (f) display the distribution of two vertically adjacent pixels in the encrypted image of Lena in the (b) red, (d) green and (f) blue components, respectively.

doi:10.1371/journal.pone.0119660.g009

Table 1. Correlation coefficients of two adjacent pixels in the plain-image and cipher-image.

Correlation direction	Plain-image			Ciphered-image		
	Red	Green	Blue	Red	Green	Blue
Horizontal	0.94003	0.94082	0.89333	0.00238	-0.00563	-0.00780
Vertical	0.96795	0.97099	0.94265	0.00098	-0.00368	0.00312
Diagonal	0.88295	0.86469	0.74511	-0.01475	-0.02953	-0.02467

doi:10.1371/journal.pone.0119660.t001

Table 2. Comparison of correlation coefficients of two adjacent pixels in different directions using the proposed algorithm with some other algorithms.

	Correlation direction		
	Horizontal	Vertical	Diagonal
The original Lena image	0.92473	0.96053	0.83092
The proposed algorithm	-0.00368	0.00014	-0.02298
Ref. [26]	0.1257	0.0581	0.0504
Ref. [28]	0.0681	0.0845	-
Ref. [30]	0.0042	0.0033	0.0024
Ref. [37]	-0.00124	0.00176	0.00193
Ref. [45]	-0.0018	0.00033	0.00427

doi:10.1371/journal.pone.0119660.t002

where s denotes the information source, N is the number of bits to represent a symbol $s_i \in s$, $p(s_i)$ is the probability of symbol s_i. For a purely random source emitting 2^N symbols, i.e., $s = 2^N$, the information entropy is $H(s) = N$. In fact, a practical information source seldom generates random messages. Therefore, in general its entropy value is smaller than the maximum one. However, when the messages are encrypted, their entropy should ideally be N. If the output of such a cipher emits symbols with entropy less than N, there is a certain degree of predictability, which threatens its security. In order to design a good cryptosystem, the information entropy of the cipher-image should be as close to the ideal case as possible.

For the cipher-image (Fig. 5(b)) of the original Lena image (Fig. 5(a)) encrypted using the proposed scheme, we record the number of occurrence of each cipher-image pixel m_i and calculate the probability of occurrence for the red, green and blue components of the cipher-image, respectively. The entropy of the three color components of the cipher-image is:

$$H_R(m) = \sum_{i=0}^{2^8-1} p(R_i)\log_2\frac{1}{p(R_i)} = 7.9893 \approx 8,$$

$$H_G(m) = \sum_{i=0}^{2^8-1} p(G_i)\log_2\frac{1}{p(G_i)} = 7.9898 \approx 8,$$

$$H_B(m) = \sum_{i=0}^{2^8-1} p(B_i)\log_2\frac{1}{p(B_i)} = 7.9894 \approx 8,$$

where R_i, G_i and B_i are the color components of the pixel m_i. The values obtained are very close to the theoretical maximum value $N = 8$ for the three color components, which indicates that information leakage in the encryption process is negligible and the cryptosystem is secure against an entropy attack. Further, Table 3 compares information entropy using the proposed

Table 3. Comparison of the information entropy using the proposed algorithm with some other algorithms.

Algorithm	Entropy		
	Red	Green	Blue
The proposed algorithm	7.9893	7.9898	7.9894
Ref. [29]	7.9871	7.9881	7.9878
Ref. [46]	7.9278	7.9744	7.9705

doi:10.1371/journal.pone.0119660.t003

algorithm with those using the existing algorithms mentioned in Refs. [29, 46]. Obviously, the entropy obtained using our proposed algorithm is indeed closer to the maximum.

1.4 Differential attack

In general, an opponent may make a slight change (e.g., modify only one pixel) in the plain-image and compare the ciphered images to find out some meaningful relationship between the plain-image and the cipher-image, which can facilitate in determining the secret key. If one minor change in the plain-image can cause a significant change in the cipher-image, with respect to diffusion and confusion, then this differential attack would become very inefficient and practically useless.

As a general requirement for all the image encryption schemes, the encrypted image should be clearly different from its original form. Such a difference can be measured by means of two criteria, namely, the number of pixel change rate (NPCR) and the unified average changing intensity (UACI) [4, 28]. The more NPCR approaches 100%, the more effective for the cryptosystem to resist a plain-text attack. The larger UACI is, the more effective for the cryptosystem to resist a differential attack.

The formulas for calculating NPCR and UACI are described, respectively, as follows:

$$\mathrm{NPCR}_{R,G,B} = \frac{\sum_{i,j} D_{R,G,B}(i,j)}{M \times N} \times 100\%, \tag{35}$$

$$\mathrm{UACI}_{R,G,B} = \frac{\sum_{i,j} \frac{|C_{R,G,B}(i,j) - C'_{R,G,B}(i,j)|}{255}}{M \times N} \times 100\%, \tag{36}$$

where M and N are the width and height of the image, $C_{R,G,B}$ and $C'_{R,G,B}$ are the two encrypted images before and after only one pixel of the plain-image is changed, respectively, $C_{R,G,B}(i,j)$ and $C'_{R,G,B}(i,j)$ are the values of the corresponding red, green or blue component in the two cipher-images, respectively. The matrix $D_{R,G,B}$ is defined as follows: if $C_{R,G,B}(i,j) = C'_{R,G,B}(i,j)$, then $D_{R,G,B}(i,j) = 0$; otherwise, $D_{R,G,B}(i,j) = 1$. For instance, for two random images with 256 × 256 pixels and 24-bit true color, the expected values of $\mathrm{NPCR}_{R,G,B}$ and $\mathrm{UACI}_{R,G,B}$ are, respectively, computed as follows: $\mathrm{NPCR}_R = \mathrm{NPCR}_G = \mathrm{NPCR}_B =$ 99.6094% and $\mathrm{UACI}_R = \mathrm{UACI}_G = \mathrm{UACI}_B = 33.4635\%$.

To test the NPCR and UACI of the proposed cryptosystem, two plain images with only one bit difference are employed, i.e., the original color image of Lena, and the other one which is obtained by randomly modifying the value '213' of the pixel located at (2,129) of the red component in the original image as '214'. Their corresponding ciphered images are obtained by encrypting the two plain images with the same key after n rounds of the proposed permutation process. The $\mathrm{NPCR}_{R,G,B}$ and $\mathrm{UACI}_{R,G,B}$ versus permutation rounds n are plotted in Fig. 10(a) and (b),

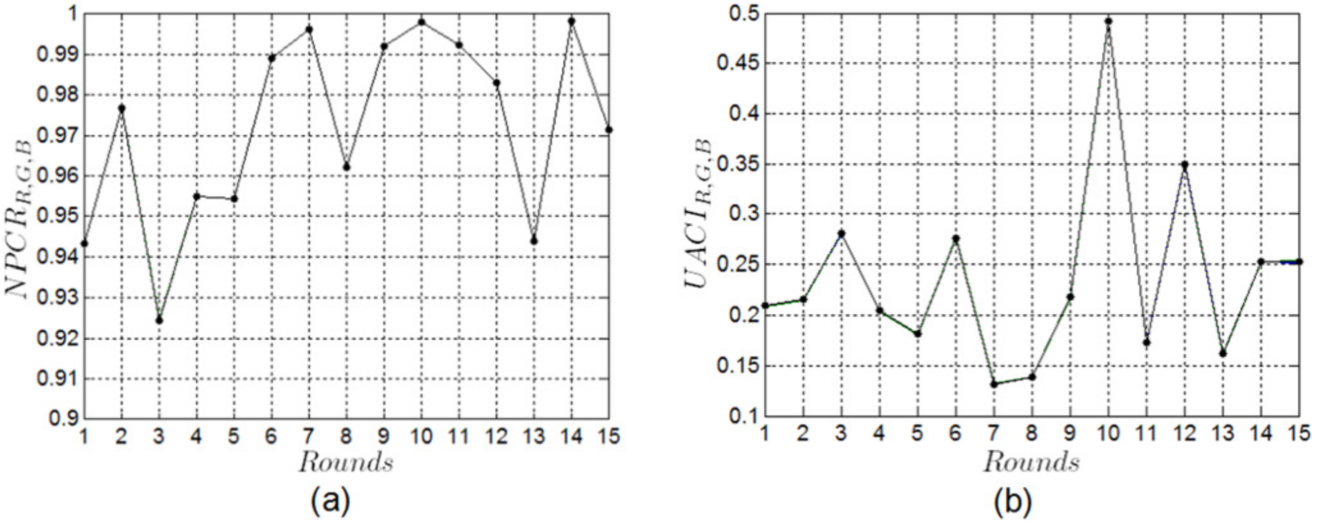

Fig 10. NPCR and UACI performance analysis of the proposed scheme. (a) NPCR versus permutation rounds, (b) UACI versus permutation rounds.

doi:10.1371/journal.pone.0119660.g010

respectively. One can find that different $NPCR_{R,G,B}$ and $UACI_{R,G,B}$ are obtained after different permutation rounds; after n = 14 permutation rounds, we get the largest value of $NPCR_{R,G,B}$, while the value of $UACI_{R,G,B}$ is small; after n = 10 permutation rounds, we obtain the largest values of $UACI_{R,G,B}$, and the values of $NPCR_{R,G,B}$ are more than 99.7%. From Fig. 10, we also find that the performance of $NPCR_{R,G,B}$ and $UACI_{R,G,B}$ are better after n = 10 permutation rounds than that obtained after other permutation rounds. The detailed results of $NPCR_{R,G,B}$ and $UACI_{R,G,B}$ with permutation rounds n = 10 are given in Table 4. One can easily see that the $NPCR_{R,G,B}$ is over 99.79% and the $UACI_{R,G,B}$ is over 49.19%, which implies that the proposed image cryptosystem is very sensitive to tiny changes in the plain-image. A slight change in the original image will result in a significant change in the ciphered image, so the proposed scheme can well resist a known/chosen plaintext attack.

Table 5 compares the $NPCR_{R,G,B}$ and $UACI_{R,G,B}$ for the proposed method and the schemes in Refs. [11, 26, 28, 30, 37, 45, 46] on the color image of Lena. As it is clear from the simulation results, our proposed cryptosystem achieves a higher performance by having $NPCR_{R,G,B} \geqslant$ 99.79% and $UACI_{R,G,B} \geqslant$ 49.19%.

1.5 Key sensitivity analysis

An efficient encryption scheme has also to be sensitive to the secret key, i.e., a very small change in the key will cause a significant change in the output. Suppose that two 280-bit secret keys are chosen randomly as: K1 =“ 2eea2814e6087660406d59f82f740bfd9c2e5e463d96bdafe482c8054f457bab5cd180” and K2 =“ 2eea2814e6087660406d59f82f740bfd9c2e5e463d96bdafe482c8054f457bab5cd181”. Obviously, two keys are different in only one bit. The key sensitivity test is carried out as follows.

The original color image of Lena is firstly encrypted by using the secret key K1 and then encrypted by using the secret key K2. We get two ciphered images by two slightly different keys. Fig. 11 displays the test results. The test shows that there is a difference up to 99.64% in terms of pixel gray-scale values between the encrypted image with K1 (Fig. 11(b)) and the encrypted one with K2 (Fig. 11(c)). Moreover, in Fig. 12, we have shown the results of some attempts to decrypt an encrypted image with slightly different secret keys. We use the color image of Lena as the plain-image. Fig. 12(a) shows the encrypted image by using the secret key K1. Fig. 12(b)

Table 4. NPCR and UACI of the color image of Lena when changing one pixel and permutation rounds *n* = 10.

	NPCR (%)	UACI (%)
Red	99.7909	49.1964
Green	99.7925	49.2234
Blue	99.7910	49.2374

doi:10.1371/journal.pone.0119660.t004

Table 5. Comparison of the average NPCR and UACI values on the color image of Lena.

Algorithm	Average NPCR (%)	Average UACI (%)
The proposed scheme	99.7915	49.2191
Ref. [11]	99.6358	33.4428
Ref. [26]	99.52	26.7933
Ref. [28]	99.5843	33.3755
Ref. [30]	99.2173	33.4055
Ref. [37]	42.7519	13.2874
Ref. [45]	99.9654	33.5720
Ref. [46]	99.6062	33.8981

doi:10.1371/journal.pone.0119660.t005

displays the decrypted image by using another trivially modified key K2. Fig. 12(c) plots the decrypted image by using the correct key K1. The encrypted image by using the secret key K2 is displayed in Fig. 12(d). The decrypted image by using the slightly different key K1 is shown in Fig. 12(e). The decrypted image by using the correct key K2 is depicted in Fig. 12(f). Obviously, the decryption with a slightly different key fails completely and hence the proposed image encryption scheme is highly key sensitive.

Furthermore, as discussed in Section 0.3, the transformations used in Equations (5)–(15) are constructed such that the initial conditions and parameters of the CML and the fractional-order Chen chaotic system are highly sensitive to a slight change even in one bit of the secret key, which will lead to undesired decryption images. The average pixel differences of some color images (Figs. 5(a), 6(a) and 13) using the random keys K1 and K2 are given in Table 6. From the results in Table 6, one can easily see that the values obtained by the proposed method are very close to the expected value of pixel difference on two randomly generated images (NPCR = 99.6094% and UACI = 33.4635%).

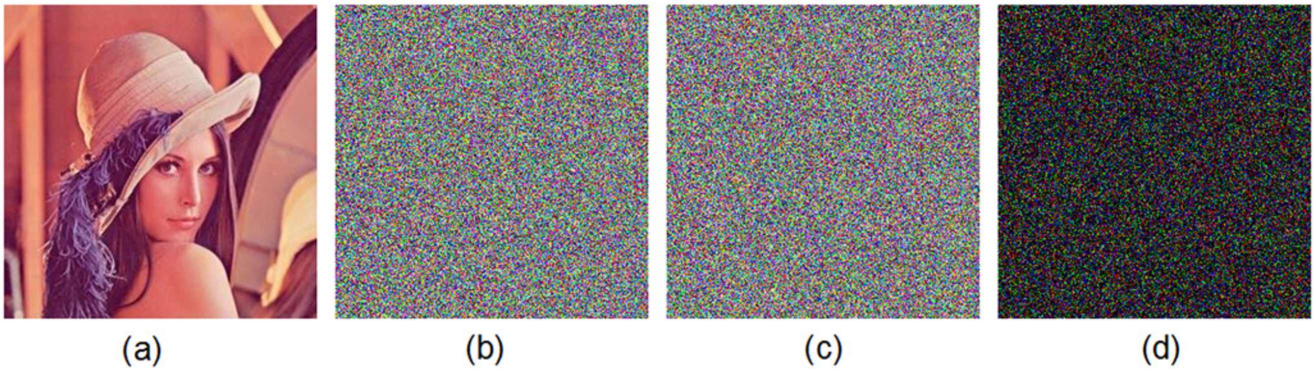

Fig 11. Key sensitivity test I. (a) Original image of Lena, (b) encrypted image of Lena with the secret key K1, (c) encrypted image of Lena with the secret key K2, (d) difference image.

doi:10.1371/journal.pone.0119660.g011

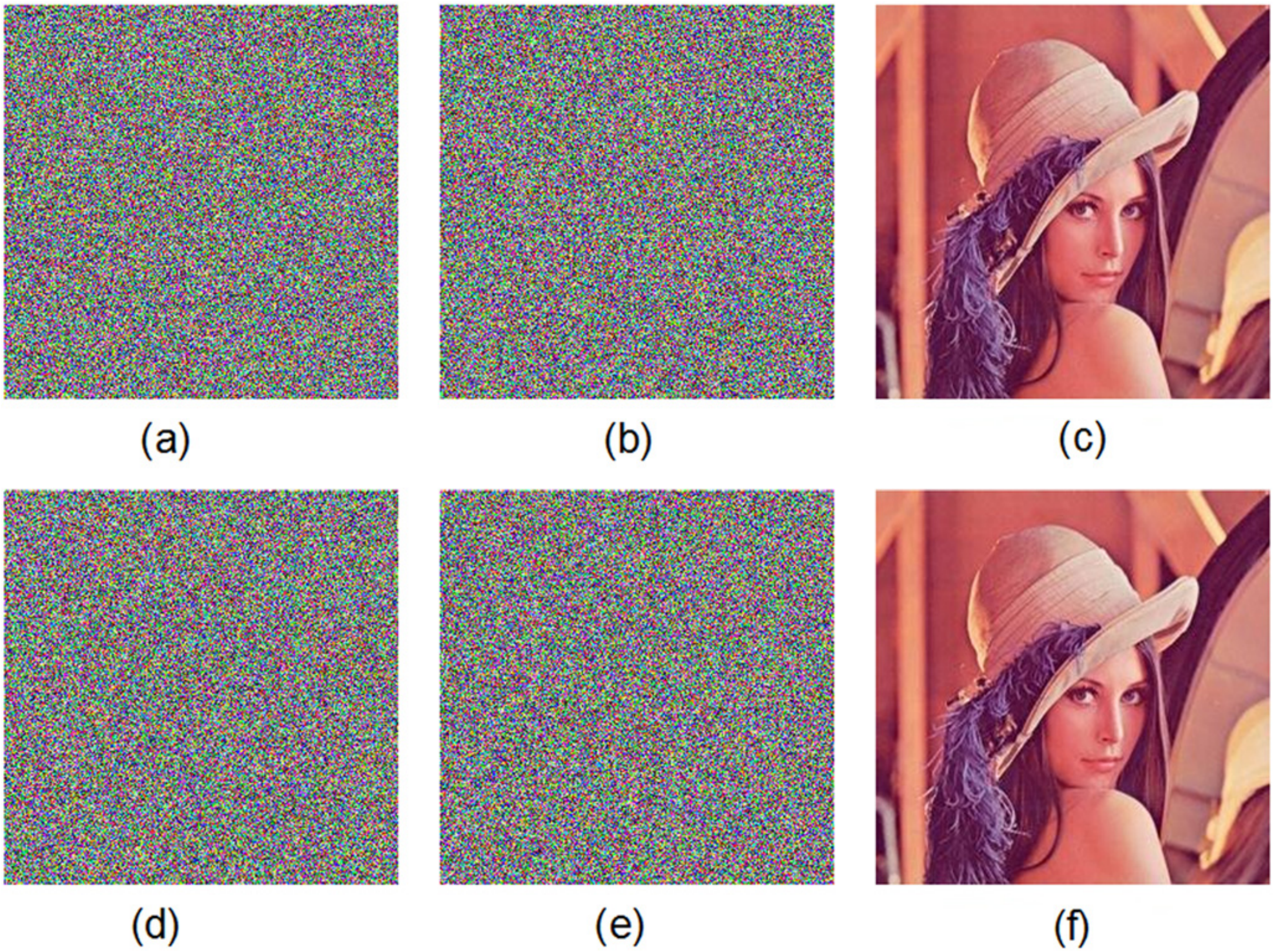

Fig 12. Key sensitivity test II. (a) Encrypted image of Lena with the secret key K1, (b) decrypted image with the secret key K2, (c) decrypted image with the secret key K1, (d) encrypted image of Lena with the secret key K2, (e) decrypted image with the secret key K1, (f) decrypted image with the secret K2.

doi:10.1371/journal.pone.0119660.g012

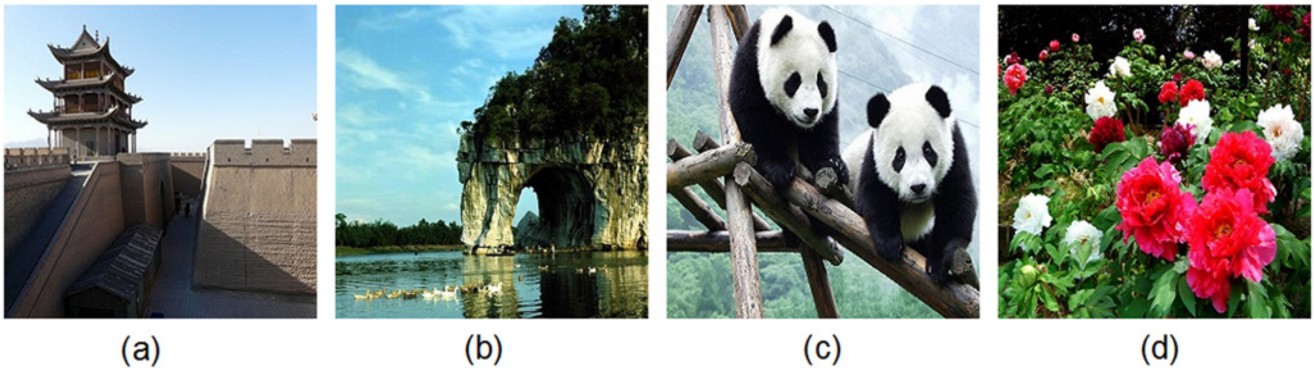

Fig 13. Some original plain-images used in Table 6. (a) City gate tower, (b) Elephantine mountain, (c) Pandas, (d) Peony.

doi:10.1371/journal.pone.0119660.g013

Table 6. Comparison of pixel difference between images encrypted by keys with one-bit difference.

Image	NPCR (%)			UACI (%)		
	Red	Green	Blue	Red	Green	Blue
Lena	99.591	99.593	99.629	33.349	33.51	33.561
Vegetables	99.637	99.611	99.605	33.396	33.549	33.583
City gate tower	99.608	99.644	99.583	33.429	33.537	33.340
Elephantine mountain	99.632	99.562	99.628	33.260	33.450	33.349
Pandas	99.634	99.634	99.615	33.498	33.529	33.530
Peony	99.614	99.623	99.628	33.462	33.505	33.522

doi:10.1371/journal.pone.0119660.t006

1.6 Speed performance

Besides the security consideration, the running speed of the algorithm is also an important issue for a well applicable cryptosystem, especially for real-time Internet applications. We implement the proposed algorithm by using Matlab 7.1. The speed performance is tested in a computer with an Intel Core 2 Duo CPU 2GHZ, 1.99GB Memory and 600GB hard-disk capacity, and the operating system is Microsoft Windows XP. From Refs. [47, 48, 49], we know that the encryption time of Refs. [5, 50, 51] are 3.704s, >10s and 2.901s, respectively. Table 7 shows the comparison of the experimental results between the proposed encryption method and other image cryptosystems in [5, 50, 51, 52]. Compared to the encryption schemes in [5, 50, 51, 52], we can see that the operation speed of our method is clearly faster for the image of Lena.

Tolerance of Image Processing

Taking into account the variation tolerance of image processing operations such as noise addition, cropping, JPEG compression etc., the ability of surviving from these attacks for an image encryption scheme is also crucial, apart from the security consideration. *PSNR* (Peak Signal-to-Noise Ratio) is used in this paper to analyze the visual quality of the decrypted image *I'* in comparison with the plain-image *I*. *PSNR* is defined as:

$$PSNR = 20\log_{10}(255/\sqrt{MSE})\text{dB}, \tag{37}$$

where *MSE* is the mean squared error between the plain-image *I* and the cipher-image *I'*, which is given by

$$MSE = \frac{1}{MN}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[I(i,j) - I'(i,j)]^2. \tag{38}$$

The higher the *PSNR* value is, the less distortion is there to the plain image. Generally speaking,

Table 7. Comparison of encryption time between our proposed method and some other cryptosystems.

Algorithm	Encryption Time (seconds)
The proposed algorithm	1.25
Ref. [5]	3.704
Ref. [50]	>10
Ref. [51]	2.901
Ref. [52]	2.3

doi:10.1371/journal.pone.0119660.t007

Table 8. The *PSNR* of decrypted image under different image processing operations.

Image processing operations	*PSNR* (dB)
Pepper & Salt Noise 0.005	47.57
Pepper & Salt Noise 0.05	31.87
Pepper & Salt Noise 0.5	19.11
Gaussian Noise [0, 0.05]	19.25
Cropping 25%	29.15
Cropping 50%	23.88
Compression (Quality Factor = 80)	18.63
Compression (Quality Factor = 50)	18.65
Rotation 45°	18.65
Brighten	18.68
Darken	18.66

doi:10.1371/journal.pone.0119660.t008

when the value of $PSNR \geqslant 30$, the human eyes cannot percept differences between the plain-image and the decrypted image. When no attack occurs, the *PSNR* value of the decrypted image (Fig. 5(c)) is 76.28. If we observe the original plain-image (Fig. 5(a)) and the decrypted image (Fig. 5(c)), we can not find any visual degradation.

In the following, several common image processing operations such as noise addition, cropping, JPEG compression, rotation, brightening and darkening are performed on our proposed encryption algorithm. Results for the "Lena" image (Fig. 5(a)) are shown in this section. Table 8 displays the *PSNR* values of the decrypted image as the cipher-image is attacked by different image processing operations. The results demonstrate that the decrypted image is still recognizable despite the cipher-image being seriously distorted. The attacks are described as follows.

2.1 Noise addition

Generally, addition of noise is responsible for the degradation and distortion of the image. The cipher-image is also degraded by noise addition, resulting in difficulties in image decryption. We tested the proposed scheme's robustness against two types of noise: Pepper & Salt noise and Gaussian noise, which are added to the cipher-image. Fig. 14(a) and (b) show the plain-image of Lena and the corresponding cipher-image without noise addition, respectively. We add Pepper & Salt noise with different noise densities, i.e., 0.005, 0.05 and 0.5, to the cipher-image, as displayed in Fig. 14(c), (e) and (g), respectively. The proposed scheme is utilized to decrypt the noise-contaminated ciphered images. The decrypted images are shown in Fig. 14 (d), (f) and (h), respectively. The corresponding *PSNR* values of the decrypted images are 47.57dB, 31.87dB and 19.11dB, respectively. In addition, Gaussian white noise with mean value 0 and variance value 0.05 is added to the cipher-image, as shown in Fig. 14(i). Fig. 14(j) plots the decrypted image of the cipher-image in Fig. 14(i). Here the *PSNR* value of the decrypted image is 19.25dB. The results demonstrate that the noise-added encrypted image can still be decrypted appropriately, i.e., most information can be recovered.

2.2 Cropping

Image cropping is very common in real applications. Cropping removes the outer parts of an image to enhance framing, accentuate subject matter or modify aspect ratio, which is a lossy manipulation. Fig. 15(a) shows that 25% of the cipher-image is removed where 255 is inserted to the cropped pixels, and then the decrypted image is well obtained using the proposed

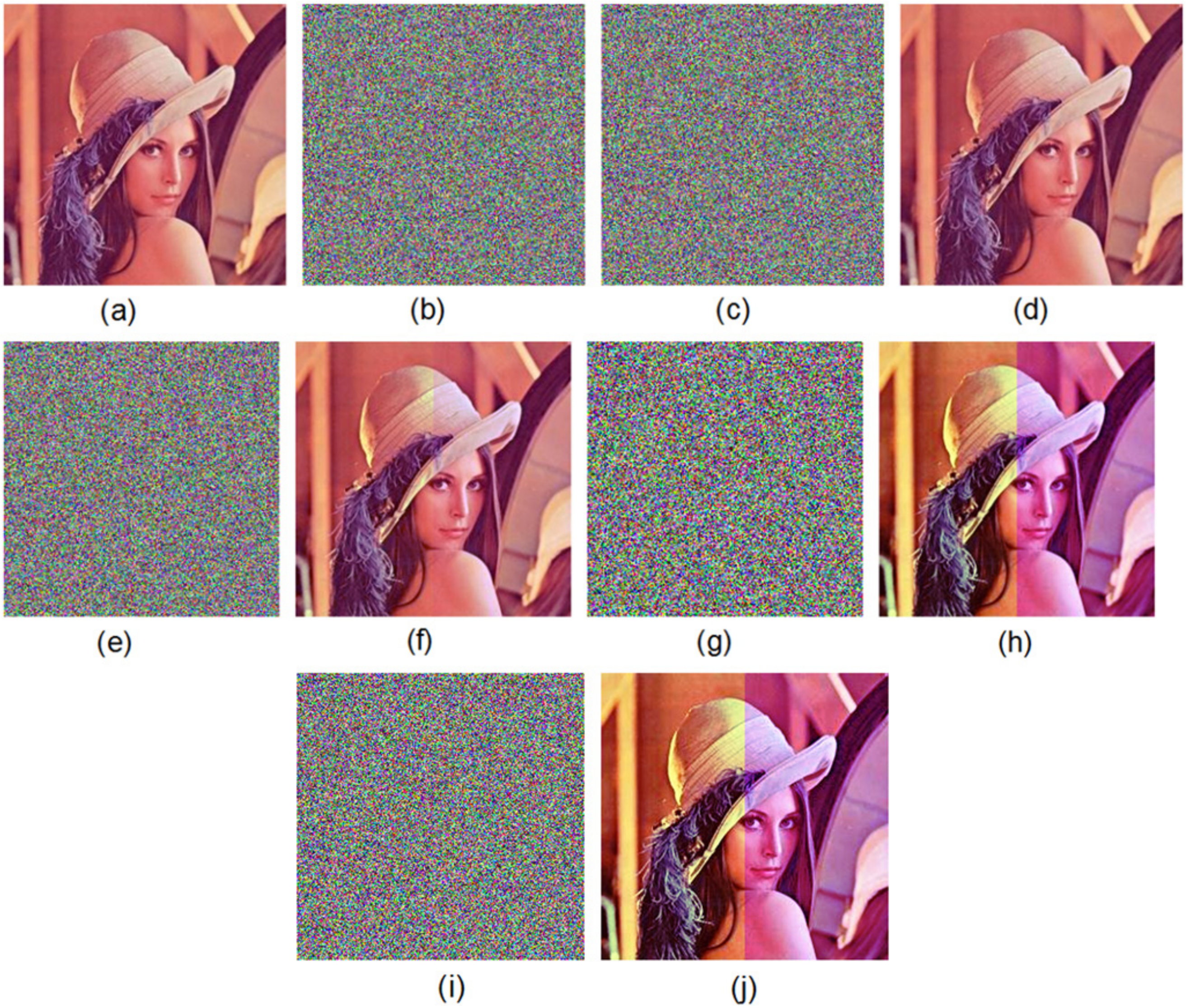

Fig 14. Test of noise addition. (a) Original image of Lena, (b) encrypted image of Lena without noise addition, (c) encrypted image of Lena under adding Pepper & Salt noise with noise density 0.005, (d) decrypted image under Pepper & Salt noise addition (noise density 0.005), (e) encrypted image of Lena under adding Pepper & Salt noise with noise density 0.05, (f) decrypted image under Pepper & Salt noise addition (noise density 0.05), (g) encrypted image of Lena under adding Pepper & Salt noise with noise density 0.5, (h) decrypted image under Pepper & Salt noise addition (noise density 0.5), (i) encrypted image of Lena under Gaussian white noise addition, (j) decrypted image under Gaussian white noise addition.

doi:10.1371/journal.pone.0119660.g014

scheme (Fig. 15(b)). The corresponding *PSNR* value is 29.15. Even there were only a half of the encrypted image remained (Fig. 15(c)), the deciphered image is still recognizable, as shown in Fig. 15(d). Here the *PSNR* value is 23.88. In fact, we can always decrypt the cropped cipher-image with most recover information when the cropped part has a size of less than 256×256 pixels.

2.3 JPEG compression

Image compression is another very prevalent operation in digital images. In testing the tolerance of JPEG compression, the results show that the encrypted image, if being JPEG compressed, can be decrypted by the designed image encryption scheme, and the decrypted image

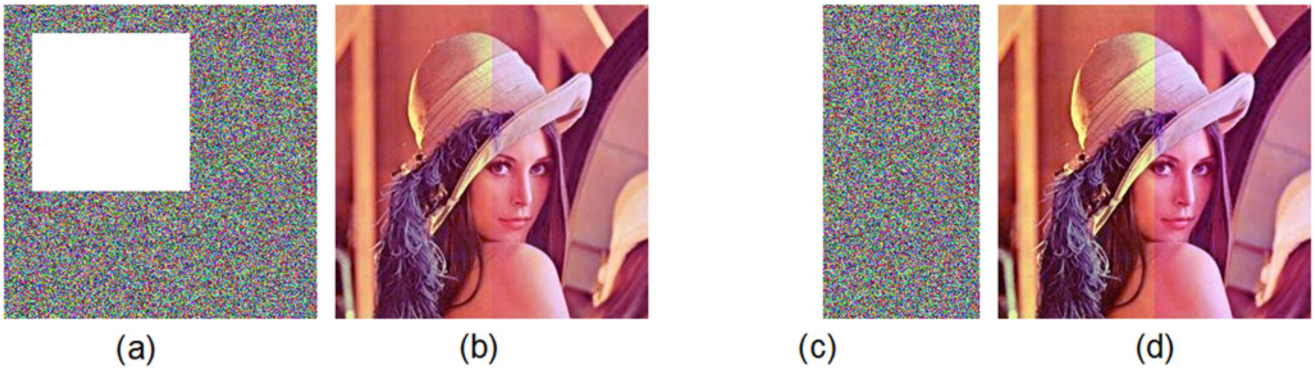

Fig 15. Test of image under cropping. (a) Cropped cipher-mage by removing 25% of the encrypted image of Lena (Fig. 14(b)), (b) decrypted image of the cropped cipher-image (a), (c) cropped cipher-mage by removing 50% of the encrypted image of Lena (Fig. 14(b)), (d) decrypted image of the cropped cipher-image (c).

doi:10.1371/journal.pone.0119660.g015

after JPEG compression is still recognizable. A test is shown in Fig. 16. Fig. 16(a) displays the cipher-image after JPEG compression, where the quality factor used by the JPEG compression is 50. Here, the quality factor is a kind of measure for JPEG compression, commonly within a range between 1 to 100: the bigger the factor, the better the quality of the image after JPEG compression and, correspondingly, the smaller the compression rate. The *PSNR* value of the decrypted image (Fig. 16(b)) is 18.65dB. Further simulation results show that the deciphered image can still be decrypted with most recover information while any quality factor in [1,100].

2.4 Rotation

Image rotation makes the coordinate axes changed. Without synchronization of the orthogonal axes, one cannot decrypt the cipher-image correctly. Here, we do not consider the question of how to recover the axes, which have been geometrically distorted. We assume that the distorted axes have been recovered before the cipher-image is decrypted. Simulations have shown that in

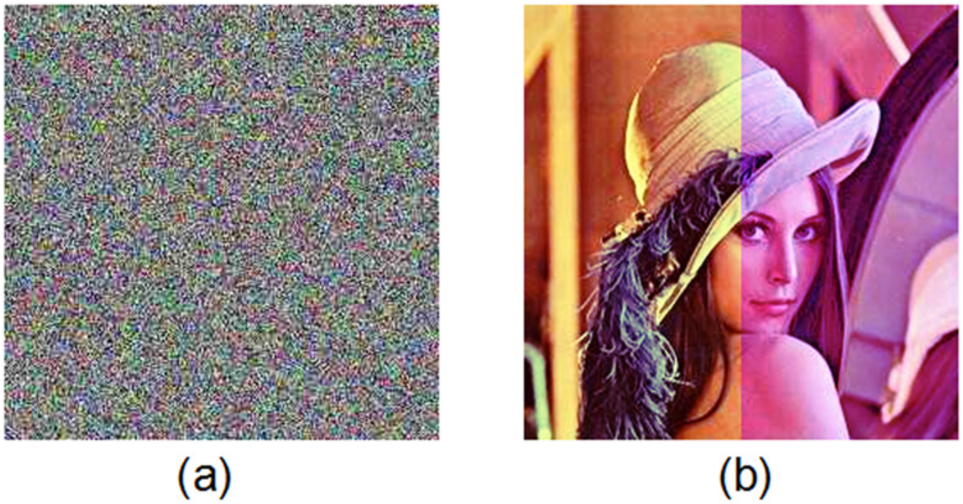

Fig 16. Test of image encryption with JPEG compression. (a) JPEG compressed cipher-mage of Lena, (b) decrypted image under JPEG compression.

doi:10.1371/journal.pone.0119660.g016

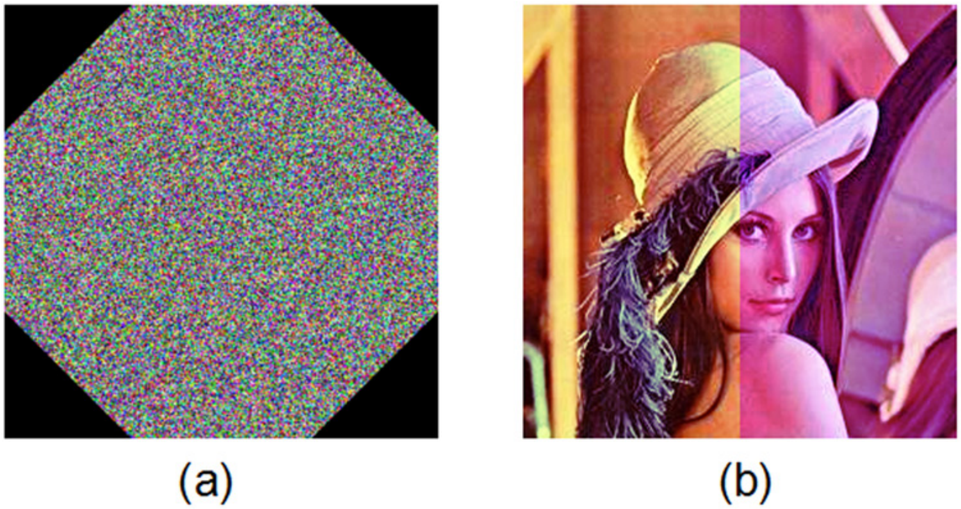

Fig 17. Test of image under rotation. (a) Rotated cipher-image of Lena, (b) decrypted image under rotation.

doi:10.1371/journal.pone.0119660.g017

this case we can still decipher the encrypted image (Fig. 17(b)) when the ciphered image is rotated by 45°, as shown in Fig. 17(a). Here the value of PSNR is 18.65dB.

2.5 Brightening and darkening

We also test the proposed scheme's robustness against image brightening and darkening attacks, which increases or decreases the color intensities in a colormap. Simulation results are given in Fig. 18. Fig. 18(a) and (c) display the cipher-image after brightened and darkened, respectively. Fig. 18(b) and (d) show the deciphered images decrypted from the brightened and darkened encrypted images, respectively, where the PSNR values of the decrypted images separately are 18.68dB and 18.66dB. These test results show that the ciphered image after brightened or darkened can still be decrypted with most recovered information.

Remark 5. From the above results, one can easily conclude that the proposed image encryption technique is highly robust against some common image processing operations and geometric attacks, e.g. noise addition, cropping, JPEG compression, rotation, brightening and darkening.

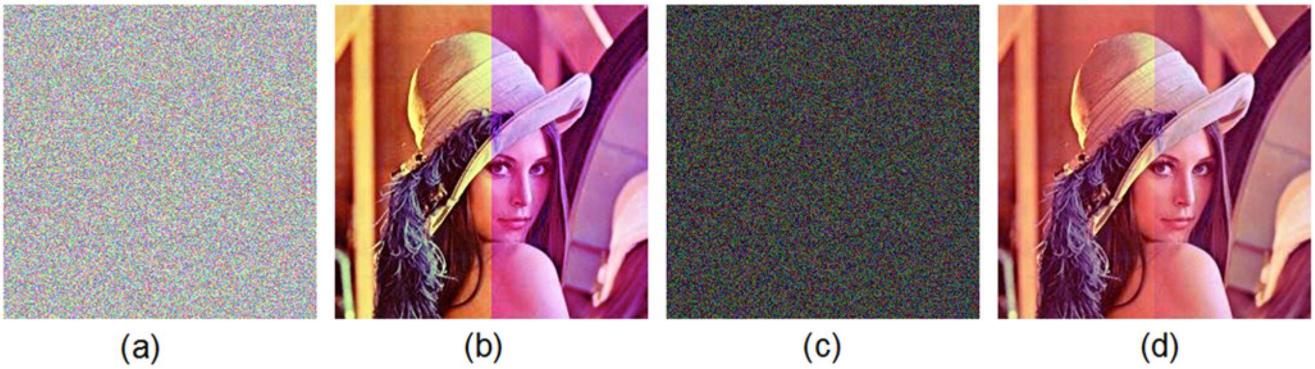

Fig 18. Test of image under brightening and darkening. (a) Brightened cipher-image of Lena, (b) decrypted image under brightening, (c) darkened cipher-image of Lena, (d) decrypted image under darkening.

doi:10.1371/journal.pone.0119660.g018

Conclusions

In this paper, we have proposed a new color image encryption algorithm based on a CML and a fractional-order chaotic system. The presented cryptosystem is composed of four processes, i.e., an image division-shuffling process, a key streams generation process, an image permutation process and an image diffusion process, to enhance the security and sensitivity of the cryptosystem. Moreover, the generation of the key streams and the image division-shuffling process are carried out simultaneously in a parallel mode, which accelerate the operation speed of our method. Experimental results have demonstrated that, comparing with current image encryption algorithms, the proposed encryption algorithm has a better performance in terms of security, sensitivity, speed and robustness. Furthermore, corresponding results also show that the presented encryption method efficiently overcomes the drawbacks in the present one-dimensional chaotic image encryption algorithms.

Acknowledgments

The authors would like to thank the anonymous reviewers and the editor for their helpful comments.

Author Contributions

Conceived and designed the experiments: XJW. Performed the experiments: XJW YL. Analyzed the data: XJW YL. Contributed reagents/materials/analysis tools: YL. Wrote the paper: XJW. Review and revise the original manuscript: JK.

References

1. Matthews R. On the derivation of a chaotic encryption algorithm. Cryptologia 1989; 13: 29–42.
2. Baptista MS. Cryptography with chaos. Phys Lett A 1999; 240: 50–54.
3. Fridrich J. Symmetric ciphers based on two-dimensional chaotic maps. Int J Bifurcat Chaos 1998; 8: 1259–1284.
4. Chen GR, Mao YB, Chui CK. A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos Soliton Fract. 2004; 21: 749–761.
5. Lian S, Sun J, Wang Z. A block cipher based on a suitable use of chaotic standard map. Chaos Soliton Fract. 2005; 26: 117–129.
6. Hermassi H, Rhouma R, Belghith S. Security analysis of image cryptosystems only or partially based on a chaotic permutation. J Syst Software 2012; 85: 2133–2144.
7. Volos ChK Kyprianidis IM, Stouboulos IN. Image encryption process based on chaotic synchronization phenomena. Signal Process. 2013; 93: 1328–1340.
8. Zhang G, Liu Q. A novel image encryption method based on total shuffling scheme. Opt Commun. 2011; 284: 2775–2780.
9. Li C, Zhang LY, Ou R, Wong KW, Shu S. Breaking a novel colour image encryption algorithm based on chaos. Nonlinear Dyn. 2012; 70: 2383–2388.
10. Pareek NK, Patidar V, Sud KK. Image encryption using chaotic logistic map. Image Vision Comput. 2006; 24: 926–934.
11. Wang X, Teng L, Qin X. A novel colour image encryption algorithm based on chaos. Signal Process. 2012; 92: 1101–1108.
12. Guan Z, Huang F, Guan W. Chaos-based image encryption algorithm. Phys Lett A 2005; 346: 153–157.
13. Fu C, Chen J, Zou H, Meng W, Zhan Y, Yu Y. Chaos-based digital image encryption scheme with an improved diffusion strategy. Opt Express 2012; 20: 2363–2378. doi: 10.1364/OE.20.002363 PMID: 22330475
14. Mao YB, Chen GR, Lian SG. A novel fast image encryption scheme based on 3D chaotic baker maps. Int J Bifurcat Chaos 2004; 14: 3613–3624.

15. Behnia S, Akhshani A, Mahmodi H, Akhavan A. A novel algorithm for image encryption based on mixture of chaotic maps. Chaos Soliton Fract. 2008; 35: 408–419.

16. Gao TG, Chen ZQ. A new image encryption algorithm based on hyper-chaos. Phys Lett A 2008; 372: 394–400.

17. Rhouma R, Belghith S. Cryptanalysis of a new image encryption algorithm based on hyper-chaos. Phys Lett A 2008; 372: 5973–5978.

18. Gao T, Gu Q, Emmanuel S. A novel image authentication scheme based on hyper-chaotic cell neural network. Chaos Soliton Fract. 2009; 42: 548–553.

19. Zhu C. A novel image encryption scheme based on improved hyperchaotic sequences. Opt Commun. 2012; 285: 29–37.

20. Özkaynak F, Özer AB, Yavuz S. Cryptanalysis of a novel image encryption scheme based on improved hyperchaotic sequences. Opt Commun. 2012; 285: 4946–4948.

21. Sun F, Liu S, Li Z, Lü Z. A novel image encryption scheme based on spatial chaos map. Chaos Soliton Fract. 2008; 38: 631–640.

22. Teng L, Wang X. A bit-level image encryption algorithm based on spatiotemporal chaotic system and self-adaptive. Opt Commun. 2012; 285: 4048–4054.

23. Song CY, Qiao YL, Zhang XZ. An image encryption scheme based on new spatiotemporal chaos. Optik 2013; 124: 3329–3334.

24. Patidar V, Pareek NK, Sud KK. A new substitution diffusion based image cipher using chaotic standard and logistic maps. Commun Nonlinear Sci Numer Simulat. 2009; 14: 3056–3075.

25. Rhouma R, Solak E, Belghith S. Cryptanalysis of a new substitution-diffusion based image cipher. Commun Nonlinear Sci Numer Simulat. 2010; 15: 1887–1892.

26. Huang CK, Nien HH. Multi chaotic systems based pixel shuffle for image encryption. Opt Commun. 2009; 282: 2123–2127.

27. Solak E, Rhouma R, Belghith S. Cryptanalysis of a multi-chaotic systems based image cryptosystem. Opt Commun. 2010; 283: 232–236.

28. Rhouma R, Meherzi S, Belghith S. OCML-based colour image encryption. Chaos Soliton Fract. 2009; 40: 309–318.

29. Liu H, Wang X. Color image encryption using spatial bit-level permutation and high-dimension chaotic system. Opt Commun. 2011; 284: 3895–3903.

30. Wei X, Guo L, Zhang Q, Zhang J, Lian S. A novel color image encryption algorithm based on DNA sequence operation and hyper-chaotic system. J Syst Software 2012; 85: 290–299.

31. Liu L, Zhang Q, Wei X. A RGB image encryption algorithm based on DNA encoding and chaos map. Comput Electr Eng. 2012; 38: 1240–1248.

32. Hartley TT, Lorenzo CF, Qammer HK. Chaos in a fractional Chua's system. IEEE Trans Circuits Syst I 1995; 42: 485–490.

33. Li C, Chen G. Chaos and hyperchaos in fractional order Rössler equations. Physica A 2004; 341: 55–61.

34. Deng W, Lu J. Design of multidirectional multi-scroll chaotic attractors based on fractional differential systems via switching control. Chaos 2006; 16: 043120. PMID: 17199398

35. Petras I. A note on the fractional-order Chua's system. Chaos Soliton Fract. 2008; 38: 140–147.

36. Wu X, Wang H, Lu H. Modified generalized projective synchronization of a new fractional-order hyperchaotic system and its application in secure communication. Nonlinear Anal RWA 2012; 13: 1441–1450.

37. Wu X, Bai C, Kan H. A new color image cryptosystem via hyperchaos synchronization. Commun Nonlinear Sci Numer Simulat. 2014; 19: 1884–1897.

38. Kiani-B A, Fallahi K, Pariz N, Leung H. A chaotic secure communication scheme using fractional chaotic systems based on an extended fractional Kalman filter. Commun Nonlinear Sci Numer Simulat. 2009; 14: 863–879.

39. Kocarev L. Chaos-based cryptography: a brief overview. IEEE Circuits Syst Mag. 2001; 1: 6–21.

40. Kaneko K, Tsuda I. Complex systems: chaos and beyond: a constructive approach with applications in life sciences. Berlin, New York: Springer; 2001.

41. Li C, Chen G. Chaos in the fractional order Chen system and its control. Chaos Soliton Fract. 2004; 22: 549–554.

42. Li C, Li S, Zhang D, Chen G. Cryptanalysis of a chaotic neural network based multimedia encryption scheme. In: Proceedings of the 5th Pacific-Rim Conference on Advances in Multimedia Information Processing. Tokyo, Japan: PCM 2004, pp. 418–425.
43. USC-SIPI. The USC-SIPI Image Database; 1977. Available: http://sipi.usc.edu/database/database.php?volume = misc.
44. Stinson DR. Cryptography: Theory and Practice. Boca Raton: CRC Press; 2007.
45. Liu H, Kadir A, Niu Y. Chaos-based color image block encryption scheme using S-box. Int J Electron Commun. (AEÜ) 2014; 68: 676–686.
46. Kadir A, Hamdulla A, Guo WQ. Color image encryption using skew tent map and hyper chaotic system of 6th-order CNN. Optik 2014; 125: 1671–1675.
47. Mohammad Seyedzadeh S, Mirzakuchaki S. A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map. Signal Process. 2012; 92: 1202–1215.
48. Ye G, Wong KW. An efficient chaotic image encryption algorithm based on a generalized Arnold map. Nonlinear Dyn. 2012; 69: 2079–2087.
49. Enayatifar R, Abdullah AH, Lee M. A weighted discrete imperialist competitive algorithm (WDICA) combined with chaotic map for image encryption. Opt Lasers Eng. 2013; 51: 1066–1077.
50. Ye RS. A novel chaos-based image encryption scheme with an efficient permutation-diffusion mechanism. Opt Commun. 2011; 284: 5290–5298.
51. Abdullah AH, Enayatifar R, Lee M. A hybrid genetic algorithm and chaotic function model for image encryption. Int J Electron Commun. (AEÜ) 2012; 66: 806–816.
52. Ye G. Image scrambling encryption algorithm of pixel bit based on chaos map. Pattern Recognition Lett. 2010; 31: 347–354.