Neural Computing and Applications
https://doi.org/10.1007/s00521-022-07849-3

S.I.: DATA PROCESSING TECHNIQUES AND APPLICATIONS FOR CYBER-PHYSICAL SYSTEMS (DPTA-2021)

Compound encryption of multiple images by utilizing a novel chaos and nonlinear transform

Limin Tao[1] · Xikun Liang[1] · Bin Hu[1] · Lidong Han[1]

Received: 23 May 2022 / Accepted: 14 September 2022

Abstract
The purpose of this paper is to propose a multiple image compound encryption based on nonlinear transforms and a novel chaos algorithm. XOR operation, sequence rearrangement, dot operation, and plaintext-related methods are all supported by our scheme. Multiple images are first unified into one image using the multilayered embedded technique. A diffusion-scrambling-nonlinear transform is then used to encrypt the compound image. Multiple images are combined and encrypted into one image; a new diffusion pattern is employed; and multiple nonlinear transforms are compounded to increase the complexity of the algorithm. Multiple nonlinear transforms are employed to increase the complexity of the cryptosystem, which is formed using a novel chaos theory. In order to verify the effectiveness and feasibility of the algorithm, simulations are developed and some performance indicators are calculated in order to evaluate its security and robustness. In comparison with some of the existing image encryption schemes, the proposed algorithm demonstrates some advantages. The first feature is that image compound encryption has greater time efficiency than single image encryption. As a second feature of the algorithm, the key space is extremely large. Furthermore, the new algorithm is more secure than some known algorithms in terms of overall security. Therefore, the proposed algorithm has the potential to be a useful tool in the field of image processing.

Keywords Image encryption · Nonlinear transform · Plaintext-related · Chaos

1 Introduction

Currently, people plan on storing their health records and sensitive photographs on their phones or in the cloud. As a result, the transmission and exchange of information via networks has become a more and more important part of daily life, making image security an inescapable issue in communication. Image encryption is widely recognized as the most important method for protecting image information in the fields of information security and applied cryptology [1, 2]. It is generally accepted that there are two major branches of image encryption: digital image encryption and optical image encryption. The Fourier transform is usually used in optical systems in conjunction with random phase masks (RPMs) to implement optical image encryption. The encryption of digital images is accomplished by digital computers. The method is commonly applied to pixels in the form of pixel scrambling or pixel diffusion [3–10]. Pseudo-random sequences are commonly used in digital image encryption in conjunction with both chaos [11] and plaintext-related technologies [12–15]. As well as this, there is another branch of study that focuses on the cryptography and protection of digital images. There are a number of representative fields,

✉ Xikun Liang
schenken@hznu.edu.cn

Limin Tao
tlm5460@hznu.edu.cn

Bin Hu
tinrant@163.com

Lidong Han
ldhan@hznu.edu.cn

[1] School of Information Science and Technology, Hangzhou Normal University, Hangzhou 311121, China

Published online: 10 November 2022

including image watermarking [16–19], image hiding [20, 21], and image fusion [22, 23].

Although great developments have been achieved in digital image encryption, we should pay much more attention to some problems. Firstly, most image encryption schemes adopted a single image information protection mode, but did not employ the comprehensive application of image cryptology methods. Second, the majority of existing image cryptosystems use pseudo-random numbers as their basic ciphers. However, the statistical security of pseudo-random sequences is not contested. Although the performance analysis was carried out in these schemes, the security of the algorithm is not guaranteed. In fact, insecure cipher-based image encryption algorithms can also pass the evaluation of performance analysis [4]. Thirdly, with the rapid performance improvement of modern computers, the pure chaotic cryptosystem with short cycles and small key space is vulnerable, and it is hard to guarantee the structural security and robustness of the encryption algorithms. To address these issues, we propose an image information hiding scheme by combining multiple images into an encrypted one. The main contributions of our work include: (i) We integrate multilayer image composition with chaos image encryption to form the compound image information protection scheme. (ii) We use a novel chaos with multiple control parameters to generate pseudo-random sequences, and these sequences are strictly tested by the universal secure criterion of pseudo-random numbers, i.e., SP800 R1a [24]; (iii) to optimize the algorithm structure, the classical image encryption pattern of 'confusion-diffusion' is expanded to one of 'diffusion-scrambling-nonlinear transform.'

The paper is mainly composed of the following parts: Part 2 introduces the mathematical foundations of the algorithm, which includes several nonlinear transforms of matrices and an irreversible chaos with multiple control parameters. Part 3 discusses the main contents of the algorithm, such as the multiple image composition, chaotic ciphers and plaintext-related ciphers, pixels diffusion, pixels scrambling, and pixels nonlinear transforms. Part 4 develops the encryption simulations. Part 5 performs the security test of the chaotic ciphers by SP800 R1a. Part 6 is the evaluation of the algorithm performance security. Part 7 summarizes the full paper.

2 Mathematical preliminaries

2.1 Matrix nonlinear transforms

This section introduces three nonlinear transforms of matrices that support the composition and encryption of images.

2.1.1 Nonlinear transform of matrices based on dot power

The 'dot operation' is a particular calculation in matrix theory. Rather than affecting the matrices themselves, this operation affects the elements of the matrices. $M = (m_{ij})_{hk}$ and $P = (p_{ij})_{hk}$ represent the matrices with the same size of $h \times k$. A dot multiply, a dot divide, and a dot power [25] between M and P can be expressed as follows:

$$M. * P = (m_{ij} \times p_{ij})_{hk},\ M./P = (m_{ij} \div p_{ij})_{hk}\ M. \wedge n = (m_{ij}^{n})_{hk} \tag{1}$$

where the symbol '.∗', './', and '.^' represent the operator of dot multiplication, dot division, and dot power, respectively.

According to the dot power defined in Eq. (1), The Nonlinear Transform of Matrices based on Dot Power (NTMDP) is introduced as a transform of matrices M and P, which is expressed as follows:

$$R_{M,P} = k_1 \cdot M + k_2 \cdot P.\hat{}n, \tag{2}$$

where $0 \le k_1, k_2 \le 1, n > 0$ represent varied parameters. It is evident that the matrix P from Eq. (2) can be easily calculated as follows:

$$\left(\left(R_{M,P} - k_1 \cdot M\right)/k_2\right).\hat{}(1/n). \tag{3}$$

Equation (3) is commonly referred to as the inverse of **NTMDP** (**NTMDP** $^{-1}$).

2.1.2 Rational linear transform of matrices

By means of the dot power defined in Eq. (1), a rational linear transform (**RLT**) of M can be defined as follows:

$$M_r = (a_1 \cdot M + a_2 \cdot N)./(a_3 \cdot M + a_4 \cdot N) + a_5 \cdot R, \tag{4}$$

where $a, b, c, d, e \in R$ represent optional parameters, while c and d cannot be taken as zero at the same time.N represents a known constant matrix with the same size of M, and R represents a random matrix.

As shown below, it is not difficult to obtain the inverse of **RLT** (**RLT**$^{-1}$):

$$M = -(a_4 \cdot (M_r - a_5 \cdot R) - a_2 \cdot N)./(a_3 \cdot (M_r - a_5 \cdot R) - a_1 \cdot N). \tag{5}$$

2.1.3 Matrix truncation transform

A rounding function ceil$(\cdot)$ is available in MATLAB. In the case of a nonzero real number, we can obtain the minimum integer not less than x by ceil(x). This function can be used to identify the truncation transform (**TT**) of the matrix X_{hk} as follows:

$$T(X) = \text{ceil}(X) - X. \tag{6}$$

Assuming that $C = \text{ceil}(M)$ is a constant matrix, $\mathbf{TT^{-1}}$ can be obtained by the inverse of Eq. (6), as follows:

$$X = C - T(X) \tag{7}$$

In terms of pixel diffusion encryption, these nonlinear transformations (Formulas (1–7)) are constructed in this subsection. It is essential that the diffusion transformation of pixels is reversible in order to facilitate decryption. Consequently, the above transformations are in complete accordance with the requirements. Further, the parameters involved in these nonlinear transformations can be used as encryption keys, thereby increasing the key space of the encryption algorithm. Particularly, some matrix-specific operations are used in these transformations, such as dot operations, which are consistent with the essence of image operations, namely matrix operations.

2.2 Multi-parameter irreversible nonlinear chaotic map

It is well known that chaos has some good properties and that it is widely employed in modern encryption algorithms. There are several common chaos systems, including the Logistic map, the Standard map, the Chebyshev map, the Baker map, the Henon map, etc. To generate chaotic ciphers, Henon's map is commonly described as follows [26]:

$$\begin{cases} x(n+1) = 1 - a \cdot x^2(n) + y(n) \\ y(n+1) = \beta \cdot x(n) \end{cases},$$

where α, β represent system parameters. When $0.54 < \alpha < 2, 0 < |\beta| < 1$, Henon system is shown to be in a hyper-chaotic state.

The following novel nonlinear chaotic map is constructed to improve the structural complexity of the Henon system:

$$\begin{cases} x(n+1) = 1 - \alpha \cdot x^2(n) + \beta \cdot \sin(x(n)) \cdot y(n) \\ \qquad\qquad + \gamma \cdot \sin(x(n)) + \lambda \cdot \cos(y(n)) \\ y(n+1) = \mu \cdot x(n) \end{cases}. \tag{8}$$

When the parameters satisfy $0.56 < \alpha < 2, 0 < |\beta|, |\gamma|, |\lambda|, |\mu| < 1$, Eq. forms a chaotic system because it has a positive Lyapunov exponent. In contrast to most chaotic maps, this map is irreversible. However, this feature does not affect its application in the proposed algorithm, since the encryption scheme has nothing to do with the chaos inverse. In the following text, this map is referred to as the Irreversible Generalized Henon Map (**IGHM**).

In comparison with the classical Henon map, trigonometric function terms and new parameters have been added to Eq. (8). As a result of such a change, the key space of the encryption algorithm is increased, and the reversible mathematical transformation is converted into an irreversible one. This enhances the security of the encryption algorithm. Despite the fact that the calculation in Eq. (8) is more complicated than that in a traditional Henon map, from the perspective of the performance of modern computing tools, the computational complexity of the algorithm has remained relatively unchanged.

3 Multiple images composition and image encryption

In the proposed algorithm, there are four basic stages: multiple image composition, compound image encryption, image decryption, and image separation from the decrypted image. Detailed information will be provided below.

3.1 Multiple images composition

Assume that $I^j_{m_j \times n_j}$, $j = 1, 2, \cdots, k$ represent the given images with different size of $m_j \times n_j$. Set $m_r = \max_j\{m_j\}$, $n_c = \max_j\{n_j\}$. By using the inner function 'rand' of MATLAB, a random matrix B_0 is introduced with the dimensions $m_r \times n_c$. The matrix B_0 is referred to as the Base-Plate Image (**BPI**). In terms of the known matrix B_0 and the given images I_j, **NTMDP** is used in Eq. (2) in order to implement the composition of images (layer by layer) in accordance with the following arithmetic:

$$\begin{cases} R_1 = p_{11} \cdot B_{r1} + p_{12} \cdot I_1. \wedge n_1 \\ R_2 = p_{21} \cdot B_{r2} + p_{22} \cdot I_2. \wedge n_2 \\ \qquad\qquad \cdots\cdots\cdots\cdots \\ R_k = p_{k1} \cdot B_{rk} + p_{k2} \cdot I_k. \wedge n_k \end{cases}, \tag{9}$$

where $p_{j1}, p_{j2}, n_j, j = 1, 2, \cdots, k$ represent the parameters similar to those mentioned in Eq. (2).B_{r1} represents a random matrix with the same size as I_1, and the elements of B_{r1} are randomly selected from the base-plate image B_0. Given the parameters p_{11}, p_{12} and an integer n_1, a transformed matrix R_1 can be easily obtained using the first formula in Eq. (9). Then, the elements are replaced at the same position in B_0 with the element of R_1. We further obtain the matrix B_1. Quite similarly, B_{r2} represents a random matrix with the same size as I_2, and the elements of B_{r2} are randomly selected from B_1. The matrix B_2. can be obtained after the elements are replaced at the same position in B_1 by the element of R_2. By analogy, the final matrix B_k is obtained. The matrix B_k is referred to as the final compound image. In addition, it represents the image that needs to be encrypted.

3.2 Image encryption

3.2.1 Chaos sequence, diffusion cipher, and pixels diffusion

3.2.1.1 Chaos cipher matrix related to plaintext Assume that a group of the parameters $\alpha_1, \beta_1, \gamma_1, \lambda_1, \mu_1$ is given. Using **IGHM** in Eq. (8), we can obtain the chaotic sequence C_0 in the length of $h \cdot k$. Then,C_0 is converted into the matrix C_1 of h rows and k columns. For the selected parameters a_1, b_1, c_1, d_1, e_1, RLT is performed in Eq. (4) among the matrix C_1, the original image P_0, and the random matrix R_1, and the matrix C_2 can be obtained as follows:

$$C_2 = (a_1 \cdot C_1 + b_1 \cdot P_0)./(c_1 \cdot C_1 + d_1 \cdot P_0) + e_1 \cdot R_1,, \quad (10)$$

where C_2 is referred to as the Chaotic Cipher Related to Plaintext **(CCRP)**.

3.2.1.2 Pixels diffusion The basic diffusion algorithm used in practical image encryption is based on the operation of XOR. Generally, forward and backward diffusion calculations using the XOR operator [27] can be expressed as follows:

$$\begin{aligned} Ct_i &= Ct_{i-1} \oplus C_i \oplus \overline{P}_i, i = 1, 2, \cdots hk, \ Ct_i \\ &= Ct_{i+1} \oplus C_i \oplus \overline{P}_i, \ i = hk, hk-1, \cdots 1 \end{aligned} \quad (11)$$

where Ct represents the diffused result, $\widetilde{C}$ represents the cipher vector,$\overline{P}$ represents the row vector rearranged from P_0.The initial value of Ct_0, Ct_{hk} is obtained by the keys, and the symbol $\oplus$ represents the XOR operator.

In Eq. (11), it is evident that the diffusion process is implemented on each element sequentially. The following pattern of forward and backward diffusion is introduced to increase the speed of pixels diffusion:

$$\begin{aligned} Ct_i^{(r)} &= Ct_{i-1}^{(r)} \oplus C_i^{(r)} \oplus P_i^{(r)}, i = 1, 2, \cdots h, \ Ct_j^{(c)} \\ &= Ct_{j-1} \oplus C_j^{(c)} \oplus P_j^{(c)}, j = 1, 2, \cdots k \end{aligned} \quad (12)$$

$$\begin{aligned} Ct_j^{(c)} &= Ct_{j+1}^{(c)} \oplus C_j^{(c)} \oplus P_j^{(c)}, \ j = k, k-1, \cdots 1, \ Ct_i^{(r)} \\ &= Ct_{i+1}^{(r)} \oplus C_i^{(r)} \oplus P_i^{(r)}, \ i = h, h-1, \cdots 1 \end{aligned} \quad (13)$$

where $Ct^{(r)}$ and $Ct^{(c)}$ represent the diffused result;$C^{(r)}$ and $C^{(c)}$ represent the cipher matrix's row vectors and column vectors, respectively; $P^{(r)}$ and $P^{(c)}$ represent the row and column vectors of P_0,respectively. The initial value of $Ct_0^{(r)}, Ct_0^{(c)}, Ct_h^{(r)}, Ct_k^{(r)}$ is obtained by the initial keys. The operations expressed by Eqs. (12) and (13) are referred to as Pixels Diffusion Row by Row and Column by Column (PDRC).

3.2.2 Scrambling ciphers and pixels scrambling

3.2.2.1 Scrambling ciphers related to the diffused image In this subsection, the ciphers related to the diffused image are used to scramble the diffused image C_d. In terms of another group of parameters $\alpha_2, \beta_2, \gamma_2, \lambda_2, \mu_2$, Eq. (8) is used to derive another chaotic sequence V in the length of hk and convert V into the matrix V_m with the same size of C_d. The second group of parameters a_2, b_2, c_2, d_2, e_2 is used to carry out RLT in Eq. (4) among C_d,V_m and another nonzero random matrix R_2, and the following C_s can be obtained as follows:

$$C_s = (a_2 \cdot C_d + b_2 \cdot V_m)./(c_2 \cdot C_d + d_2 \cdot V_m) + e_2 \cdot R_2, \quad (14)$$

where the matrix C_s is referred to as the Scrambling Cipher Matrix (SCM).

3.2.2.2 Pixels scrambling Assume that $\overline{C_s}$ represents the row vector converted from C_s. By using MATLAB's function 'randperm,' a series of random positive integers are generated in the interval $[1, h \cdot k]$. Therefore, $\overline{C_s}$ is arranged into $\overline{\overline{C_s}}$ and the elements numbers are recorded before and after the rearranging as I_{old} and I_{new}, respectively.$\overline{C_d}$ is first rearranged by the sequence I_{new}. Then, the rearranged result is converted into a matrix $\overline{\overline{C_d}}$ with h rows and k columns. Thus, the scrambled image can be obtained.

3.2.3 Encryption using the truncation transformation

Performing TT in Eq. (6) to the scrambled result $\overline{\overline{C_d}}$, we can obtain

$$CT = \text{ceil}\left(\overline{\overline{C_d}}\right) - \overline{\overline{C_d}} \quad (15)$$

In this section,$C = \text{ceil}\left(\overline{\overline{C_d}}\right)$ is referred to as the Procedural Cipher Matrix (**PCM**). The final encrypted image CT has been obtained so far.

3.3 Image decryption

Cryptography can be decrypted by performing the inverse operation and inverse transformation on it. In terms of the encrypted image CT, we use Eq. (7) to deduce CT_1. In the following step, we change CT_1 into a one-dimensional array, rearrange the array by I_{old}, and convert the result into the matrix CT_2. The matrix CT_2 represents the decrypted result of the scrambled image. Furthermore, C_3 is used to perform the inverse of pixels diffusion (column by column and row by row) as follows:

$$\begin{aligned} I_j &= CT_{2,j-1}^{(c)} \oplus CT_{2,j}^{(c)} \oplus C_{3,j}^{(c)}, \quad j = 1, 2, \cdots k \\ I_i &= CT_{2,i-1}^{(r)} \oplus CT_{2,i}^{(r)} \oplus C_{3,i}^{(r)}, \quad i = 1, 2, \cdots h \end{aligned} \quad (16)$$

$$\begin{aligned} I_i &= CT^{(r)}_{2,i+1} \oplus CT^{(r)}_{2,i} \oplus C^{(r)}_{3,i}, \quad i = h, h-1, \cdots 1 \\ I_j &= CT^{(c)}_{2,j+1} \oplus CT^{(c)}_{2,j} \oplus C^{(c)}_{3,j}, \quad j = k, k-1, \cdots 1 \end{aligned} \tag{17}$$

where the symbols $[\cdot]^{(c)}$ and $[\cdot]^{(r)}$ represent the column and row vector of the corresponding matrix, respectively. Finally, the final decrypted image I can be obtained.

3.4 Image separation from the decrypted result

The final decrypted image I contains the information about the multiple compound images and the base-plate image. As a general principle, image separation involves extracting every image from I in reverse order of composition.

In terms of the image I and the known parameters p_{k1}, p_{k2}, n_k, an operation similar to Eq. (3) is performed on the last formula of Eq. (9) and the image I_k can be obtained as follows:

$$I_k = ((R_k - p_{k1} \cdot B_{rk})/p_{k2}). \wedge (1/n_k) \tag{18}$$

Similarly, the other images can be derived as follows:

$$I_{k-1} = ((R_{k-1} - p_{k-11} \cdot B_{rk-1})/p_{k-12}). \wedge (1/n_{k-1}) \tag{19}$$

$$I_1 = ((R_1 - p_{11} \cdot B_{r1})/p_{12}). \wedge (1/n_1) \tag{20}$$

Therefore, all the initial images are sequentially separated from the decrypted image I.

3.5 Algorithm description

The proposed algorithm for hiding image information consists of five main stages: image preprocessing, image composition, the compound image encryption, the image decryption, and image separation. Following is a summary of the details:

(I) *Image preprocessing*

■$_1$Convert the initial images with different sizes into the gray ones $I_1, I_2, \cdots, I_k$, and generate the base-plate image B_0 using the function 'rand' of MATLAB.

(II) *Image composition*

■$_2$ Set the values of parameters $p_{j1}, p_{j2}, n_j, j = 1, 2, \cdots, k$, execute the operation in Eq. (9) to deduce the compound image (it is also the image to be decrypted)B_k.

(III) *The compound image encryption*

■$_3$ Compute the diffusion cipher for the given parameters according to Eq. (4), and perform the diffusion operation described in Eqs. (12) and (13) to obtain the diffused image C_d.

■$_4$Set the proper parameters and implement the pixels scrambling operation as described in Sect. 3.2.2 to obtain the scrambled image $\overline{\overline{C_d}}$.

■$_5$Carry out the transform for the image $\overline{\overline{C_d}}$ in Eq. (15) to obtain the final encrypted image CT.

(IV) Image Decryption.

■$_6$Use Eq. (7) to deduce CT_1 for the encrypted image CT.

■$_7$Turn the result of CT_1 into CT_2;

■$_8$Perform the transforms in Eqs. (16) and (17) on CT_2.Then, rearrange the result into the matrix I.

(V) Image separation.

■$_9$Run the calculations in Eqs. (18), (19), and (20), respectively. The initial images are abstracted sequentially from the image I.

3.6 The flow charts of the algorithm

The flow charts for the algorithm are illustrated in Figs. 1 and 2.

4 Simulations

In this section, three images are selected, namely Girl, Coast, and Tulips, respectively, from the universal image datasets BSD-S500 [28] and Caltech 101 [6], to be used for image composition and the decryption and encryption of compound images. Simulations are performed using the platform MATLAB 2018a.

4.1 Images composition experiment

It should be noted that since each of the three experimental images is 481 × 321, 341 × 512, and 570 × 380 pixels, respectively, so should the base-plate be 570 × 512 pixels. The three initial images and the random base-plate can be seen in Fig. 3.

For each of the parameters mentioned in Eq. (9), set the following values:

p_{11}=0.75, p_{12}=0.25, n_1= 0.10, p_{21}=0.65, p_{22}=0.20, n_2= 0.12, p_{31}=0.60, p_{32}=0.22, n_3= 0.25.

Afterward, the three images are integrated into the base-plate in order, and the compound images are obtained as shown in Fig. 4. As shown in the figure, (b), (c), and (d) represent the results of integrating one, two, and three experimental images into the base-plate, respectively. Subfigure (d) represents the final compound result. The plain image to be encrypted is also shown in subfigure (d).

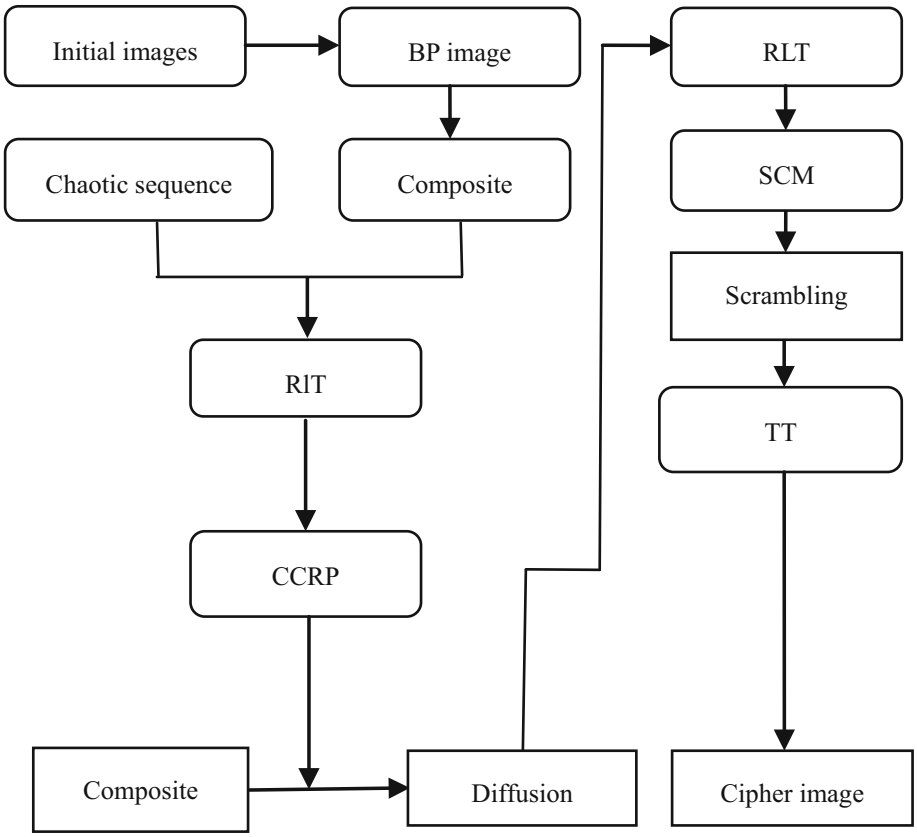


Fig. 1 The flow chart of images composition and encryption

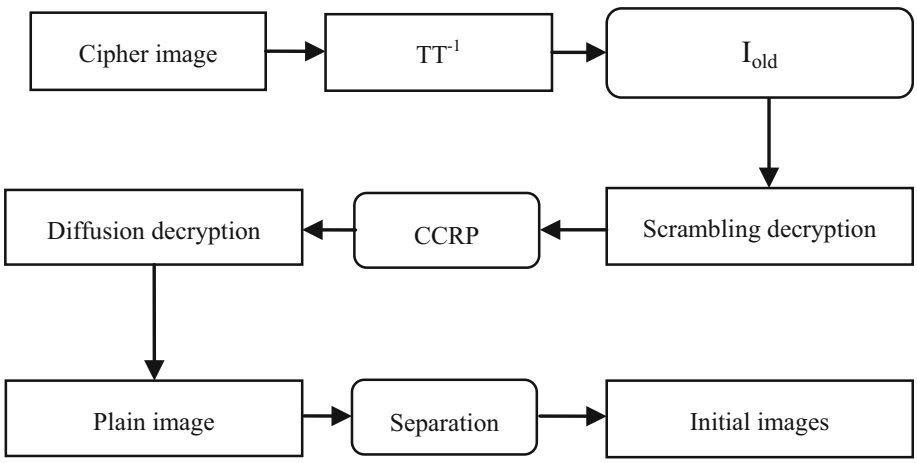


Fig. 2 The flow chart of image decryption and separation

4.2 The compound image encryption and decryption experiment

According to the encryption scheme described above, there are 22 parameters involved, and the key space is expressed as follows:

$$\Gamma_1 \times \Gamma_2 = (\alpha_1, \beta_1, \gamma_1, \lambda_1, \mu_1, a_1, b_1, c_1, d_1, e_1, R_1) \times (\alpha_2, \beta_2, \gamma_2, \lambda_2, \mu_2, a_2, b_2, c_2, d_2, e_2, R_2).$$

The matrices R_1 and R_2 in this set are randomly generated, and the other keys are assigned sequentially as follows:

$$\alpha_1 = 1.2, \beta_1 = 0.58, \gamma_1 = 0.72, \lambda_1 = 0.35, \mu_1 = 0.40,$$

$$a_1 = 2.00, b_1 = 5.00, c_1 = 3.00, d_1 = 4.00, e_1 = 4.00,$$

$$\alpha_2 = 1.1, \beta_2 = 0.55, \gamma_2 = 0.68, \lambda_2 = 0.30, \mu_2 = 0.45,$$

$$a_2 = 4.00, b_2 = 5.00, c_2 = 2.00, d_2 = 3.00, e_2 = 3.00.$$

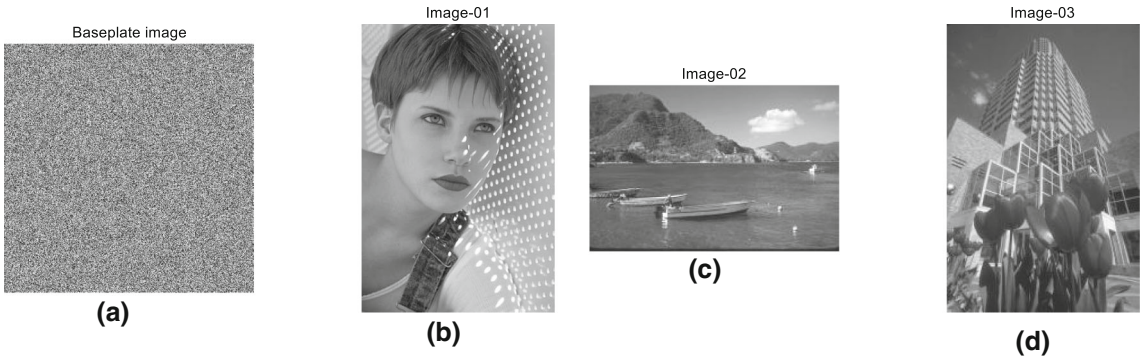


Fig. 3 The base-plate image and the three initial images

Fig. 4 The compound results of different images

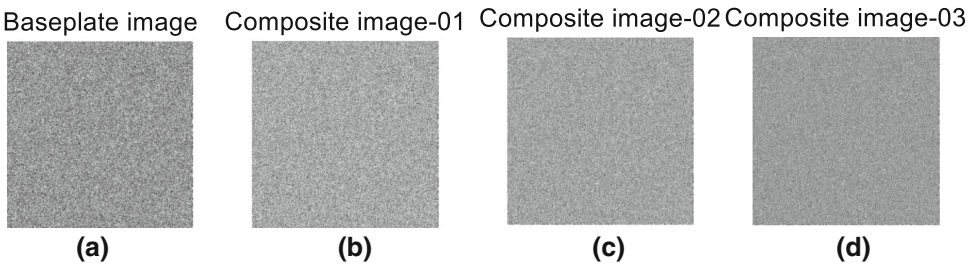
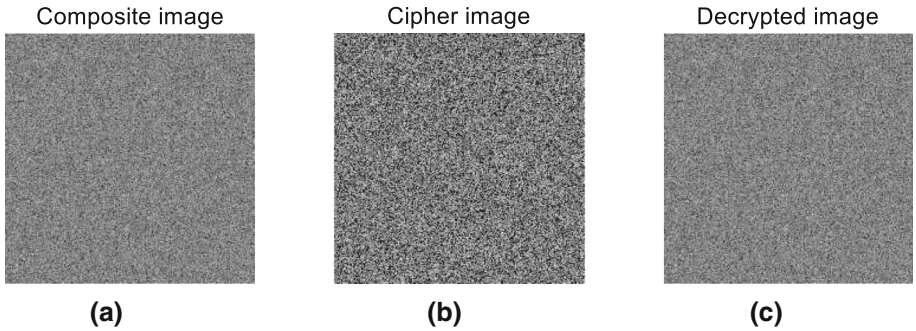


Fig. 5 The compound, the encrypted and decrypted images



As shown in Fig. 5, the results of the encryption and decryption of the compound image are displayed, respectively. The subfigure (a) represents the compound image, the subfigure (b) represents the encrypted image, and the subfigure (c) represents the decrypted image.

It is evident that the cipher pixels are random and disordered from an intuitive perspective. Therefore, we are able to preliminarily conclude that the encryption algorithm is valuable.

A more objective evaluation of the encryption result can be obtained by calculating the three indexes (PSNR, SSIM, and coefficient correlation (CORR). A comparison of PSNR, SSIM, and coefficient correlation (CORR), which are typically used to assess the similarity between two images, is presented in Table 1. As can be seen from Table 1, the values of the three indicators are very close to theoretical values. Thus, it can be inferred that the proposed algorithm has good encryption effect.

4.3 Experimentation with separating the encrypted image from the unencrypted image

The separated images are shown in Fig. 6. A table of similarity indexes is also included in Table 2.

It is evident from Fig. 6 that the initial images can be clearly separated from the decrypted images. According to Table 2, the PSNR, SSIM, and CORR values are quite close to those predicted by the theory. It is clear from this that the proposed algorithm provides excellent results in terms of decryption. Additionally, the data in lines 4 and 5 demonstrate that the algorithm is lossless, since decrypted images are identical to their original ones.

Table 1 PSNR, SSIM, and CORR

	PSNR	SSIM	CORR
The compound plaintext-The ciphertext	7.7601	0.0060	3.9112×10^{-4}
Theoretical value	≤ 30	0.0000	0.0000

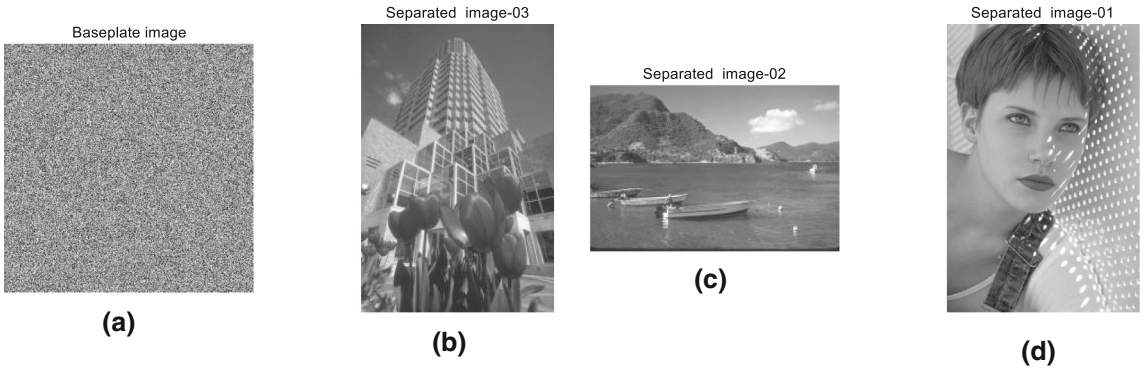


Fig. 6 The separated images from the decrypted image

Table 2 PSNR, SSIM, and CORR between the initial images and the corresponding decrypted images

	Girl	Coast	Tulips	Base-plate	Theoretical value
Size	481×321	341×512	570×380	570×512	–
PSNR	350.7228	355.8016	359.2723	Infinite	≥ 65
SSIM	1.0000	1.0000	1.0000	1.0000	1.0000
CORR	1.0000	1.0000	1.0000	1.0000	1.0000

5 Security testing of the cryptosystem

Pseudo-random sequences are universal in nature. A cryptographic system cannot be constructed using an arbitrary pseudo-random sequence. Cryptosystems can only be based on those that have been proven to be secure. In this paper, the cryptosystem is based on the chaotic sequence generated by Eq. (8). Therefore, it is necessary to test the security of the chaotic sequence. It is assumed that the parameters in Eq. (8) are assigned as 1.2, 0.58, 0.72, 0.35, and 0.40, respectively. A statistical analysis of random numbers is carried out using 15 indicators based on the international standard SP800.R1a [24]. As shown in Table 3, the results are recorded.

As per SP800, R1a, if the value of an indicator is greater than 0.01, the indicator is considered to pass the security test. The random number generation method is considered secure if all the indicators pass the test. According to Table 3, all the indicators of the proposed chaotic sequence pass the test. Based on these findings, it can be concluded that the cryptosystem based on the proposed chaotic sequence is secure.

6 Algorithm performance analysis and security evaluation

In this section, a quantitative analysis of the proposed algorithm security indexes will be presented. Because the algorithm involves a number of operations, the results obtained from one round may differ from those obtained from another. As a result, this is the inherent characteristic of the algorithm. As a matter of fact, this characteristic does not affect the conclusions drawn from the analysis.

6.1 The average running time of the algorithm

The purpose of this subsection is to provide a running time evaluation of the proposed algorithm. To compute the algorithm time cost, we randomly select five groups of images (each group includes three images with different sizes) from [28] and [6]. As shown in Table 4, the average time it takes for the algorithm to run 50 times is summarized. Figure 7 illustrates the time fitting curves of the cubic polynomial (the abscissa represents the number of pixels in the base-plate image, while the ordinate represents the time). It is evident that the algorithm consumes a

Table 3 The 15 statistical indicators of SP800.R1a for chaotic sequence

Number	1	2	3	4	5	6	7
Index	FT	FTB	RT	TLROB	BMRT	DFTT	NTMT
Test result	0.3332	0.9369	0.9724	0.9963	0.0130	0.2341	0.2761
Test parameter	$n = 10^5$	$n = 10^5$, $m = 100$	$n = 10^5$	$n = 10^5$, $m = 128$	$n = 10^5$, $M_1 = Q = 32$	$n = 10^5$	$n = 10^5$, $N = 20$, $m_1 = 10$
Conclusion	Reasonable	Reasonable	Reasonable	Reasonable	Reasonable	Reasonable	Reasonable

Number	8	9	10	11	12	13	14	15
Index	OTMT	MT	LCT	ST	AET	CST	RET	REVT
Test result	0.1666	0.3166	0.0872	0.1291,0.0722	0.7125	0.9925	D1	D2
Test parameter	$n = 10^5$, $N = 128$, $M = 512$, $K = 5$	$n = 10^4$	$n = 10^6$, $M = N = 10^3$ $K = 6$	$n = 10^5$, $m = 3$	$n = 10^3$	$n = 10^4$	$n = 10^5$	$n = 10^5$
Conclusion	Reasonable	Reasonable	Reasonable	Reasonable	Reasonable	Reasonable	Reasonable	Reasonable

n represents the length of the test sequence; m represents the length of the block; M_1 and Q represent the length and width of the binary matrix, respectively; N represents the number of the block; m_1 represents the length of the template; M represents the length of the block; K represents the degree of freedom for χ^2 statistics. D1 = {0.6277, 0.5074, 0.3554, 0.8539, 0.6566, 0.3686, 0.4629, 0.6531}, D2 = { 0.3130, 0.2619, 0.5132, 0.5964, 0.6111, 0.5881, 0.6493, 0.9362, 0.6111, 0.5175, 0.6500, 0.9670, 0.9165, 0.9754, 0.9002, 0.8175, 0.8113, 0.7198}.

Table 4 Resources consumption and occupancy

Images group		1	2	3	4	5
Image sizes (pixels)	1	128 × 256	128 × 321	281 × 321	481 × 321	481 × 364
	2	300 × 321	341 × 256	384 × 256	341 × 512	388 × 512
	3	288 × 364	300 × 380	320 × 400	570 × 380	640 × 384
	Base-plate	300 × 364	341 × 380	384 × 400	570 × 512	640 × 512
Encryption time		0.64 s	1.42 s	2.43 s	3.32 s	3.56 s
Decryption time		0.31 s	0.76 s	1.04 s	1.65 s	1.78 s

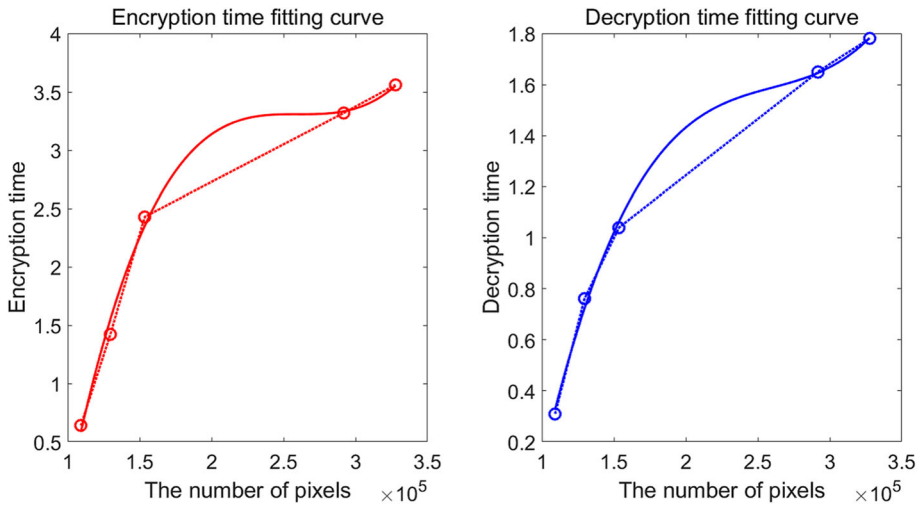


Fig. 7 Time fitting curves of the cubic polynomial

relatively small amount of time. Despite the fact that some random factors and calculations are involved in the algorithm and the experimental result is closely related to the computer configuration, data presented in Table 4 should be regarded as relative and should only be used as reference.

6.2 Security of the brute force attack

6.2.1 Keys space

A key space is the dataset composed of all the possible values of the keys, which is a key indicator of how secure the algorithm will be against brute force attacks. The application of cryptography has demonstrated the need for a secure key space that exceeds 128 bits for an 8-bit integer image.

It is known that the algorithm's key space is $\Gamma_1 \times \Gamma_2$. Assume that all the parameters are double-precision floating-point decimals. Based on theoretical considerations, the magnitude of each parameter is 10^{14}. Therefore, the key space is not less than $\log_2(10^{308}) \approx 1024$ bit. Even if these variables are conservatively taken from the interval $[10^{-4}, 10^4]$, the key space is greater than $\log_2(10^{176}) \approx 585$ bit. In such a case, the key space is large enough to withstand brute force attacks.

6.2.2 Keys sensitivity

In this subsection, three parameters α_1, b_1, c_2 are selected to estimate the algorithm key sensitivity. The compound image is encrypted using the following four groups of keys.

$$K_0 = (\overline{\alpha}_1, \overline{\beta}_1, \overline{\gamma}_1, \overline{\lambda}_1, \overline{\mu}_1, \overline{a}_1, \overline{b}_1, \overline{c}_1, \overline{d}_1, \overline{e}_1, \overline{\alpha}_2, \overline{\beta}_2, \overline{\gamma}_2, \overline{\lambda}_2, \overline{\mu}_2, \overline{a}_2, \overline{b}_2, \overline{c}_2, \overline{d}_2, \overline{e}_2, \overline{R}_1, \overline{R}_2);$$

$$K_1 = (\overline{\alpha}_1 + 10^{-10}, \overline{\beta}_1, \overline{\gamma}_1, \overline{\lambda}_1, \overline{\mu}_1, \overline{a}_1, \overline{b}_1, \overline{c}_1, \overline{d}_1, \overline{e}_1, \overline{\alpha}_2, \overline{\beta}_2, \overline{\gamma}_2, \overline{\lambda}_2, \overline{\mu}_2, \overline{a}_2, \overline{b}_2, \overline{c}_2, \overline{d}_2, \overline{e}_2, \overline{R}_1, \overline{R}_2)$$

$$K_2 = (\overline{\alpha}_1, \overline{\beta}_1, \overline{\gamma}_1, \overline{\lambda}_1, \overline{\mu}_1, \overline{a}_1, \overline{b}_1 + 10^{-10}, \overline{c}_1, \overline{d}_1, \overline{e}_1, \overline{\alpha}_2, \overline{\beta}_2, \overline{\gamma}_2, \overline{\lambda}_2, \overline{\mu}_2, \overline{a}_2, \overline{b}_2, \overline{c}_2, \overline{d}_2, \overline{e}_2, \overline{R}_1, \overline{R}_2)$$

$$K_3 = (\overline{\alpha}_1, \overline{\beta}_1, \overline{\gamma}_1, \overline{\lambda}_1, \overline{\mu}_1, \overline{a}_1, \overline{b}_1, \overline{c}_1, \overline{d}_1, \overline{e}_1, \overline{\alpha}_2, \overline{\beta}_2, \overline{\gamma}_2, \overline{\lambda}_2, \overline{\mu}_2, \overline{a}_2, \overline{b}_2, \overline{c}_2 + 10^{-10}, \overline{d}_2, \overline{e}_2, \overline{R}_1, \overline{R}_2)$$

The summary of the four cipher images can be found in Fig. 8. We cannot determine the difference between the ciphertexts from the visual representations. As a result, we calculate the objective indicators between (a) and (b), (a) and (c), and (a) and (d) in order to evaluate the difference between the ciphers. As shown in Table 5, the results are summarized. As shown in Table 5, the encryption algorithm exhibits a high level of keys sensitivity.

6.3 Security of the statistic attack

6.3.1 Gray histogram

For an image encryption scheme to be secure and robust, the pixels of the ciphertext should display a uniform distribution that differs from the distribution of the plaintext. This feature is revealed by the gray histogram of the encrypted image. We plot the compound image and cipher image histograms in Fig. 9.

As shown in Fig. 9, the gray histogram of plaintext is convex, whereas that of ciphertext is uniform and smooth. Consequently, it can be concluded that the algorithm is secure and robust in terms of gray statistical characteristics.

6.3.2 Pixels correlation

In order to generate the compound image, we randomly select 3000 pixels from the plaintext and the corresponding pixels from the ciphertext in the x-direction, the y-direction, and the direction of the line y = x, respectively. After that, we plot the distribution diagrams of pixels in Figs. 10 and 11.

Figures 9 and 10 illustrate that plain pixels have an approximate ellipse distribution in the three directions, whereas cipher pixels have a decentralized and indefinite distribution. As a result, the plain pixels correlation has been effectively eliminated by the encryption algorithm.

6.3.3 Information entropy

The term entropy is used to describe the randomness and unpredictable nature of a system. Cryptography and chaos have demonstrated that the least upper bound for the

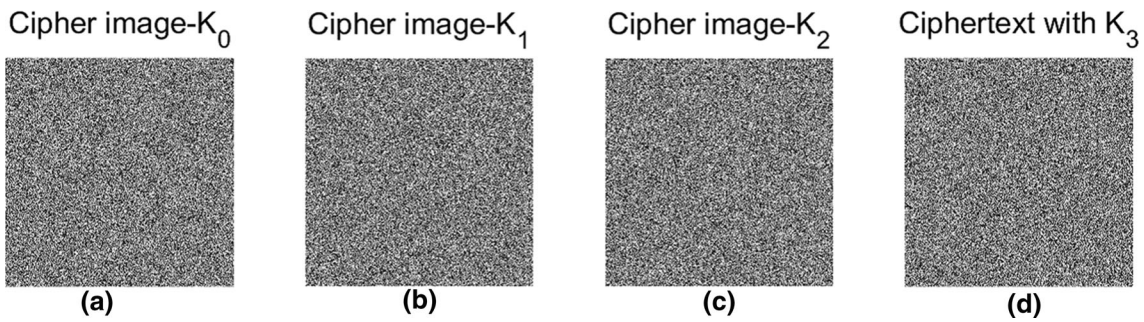


Fig. 8 Cipher images corresponding to the four groups of different keys

Table 5 CCOR, SSIM, NPCR, UACI, and BACI of the ciphertexts before and after the key's alternation

Indexes	K_1–K_0	K_2–K_0	K_3–K_0	Theoretical values
CCOR	− 0.0016	− 0.0020	0.0010	0.0000
SSIM	0.0044	0.0041	0.0070	0.0000
NPCR	99.6073	99.6135	99.6289	99.6094%
UACI	33.4970	33.5269	33.4094	33.4636%
BACI	26.8223	26.7933	26.7227	26.7712%

entropy of an 8-bit integer image is 8. In image processing, a secure and robust encryption algorithm should be able to guarantee that the cipher entropy nearly equals 8.

In Table 6, the results of the entropies are presented. Based on these data, it can be concluded that the algorithm has good statistical properties.

6.4 Plaintext sensitivity

$P_0, P_0 + \Delta P, C_{P_0}$, and $C_{P_0+\Delta P}$ are used to denote the plaintext images before and after the pixels alteration and the corresponding ciphertext images, respectively. We consider an encryption scheme to have plaintext sensitivity if a slight variation of plaintext pixels results in a large difference between C_{P_0} and $C_{P_0+\Delta P}$.

Assume that $\Delta P = 10^{-10}$. We calculate a number of similarity indicators between the two ciphertexts and record the results in Table 7. The results of the study indicate that the encryption algorithm is highly sensitive to plaintext attacks and secure against differential attacks.

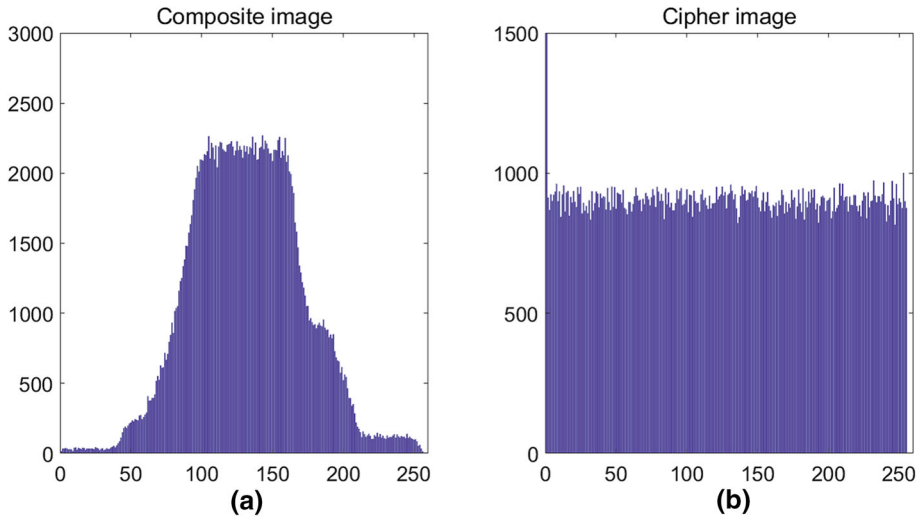


Fig. 9 Histograms of plaintext and ciphertext

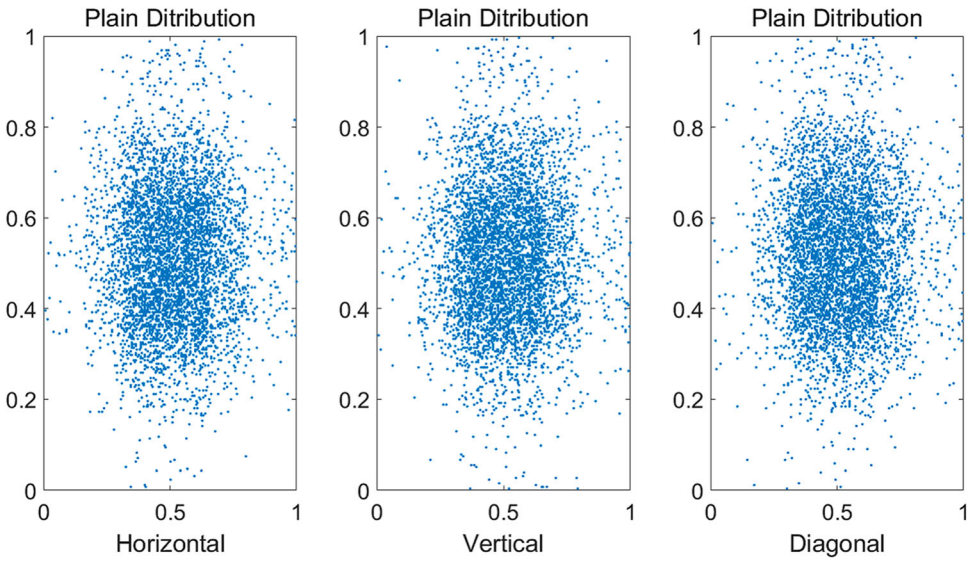


Fig. 10 Distributions of the plain pixels

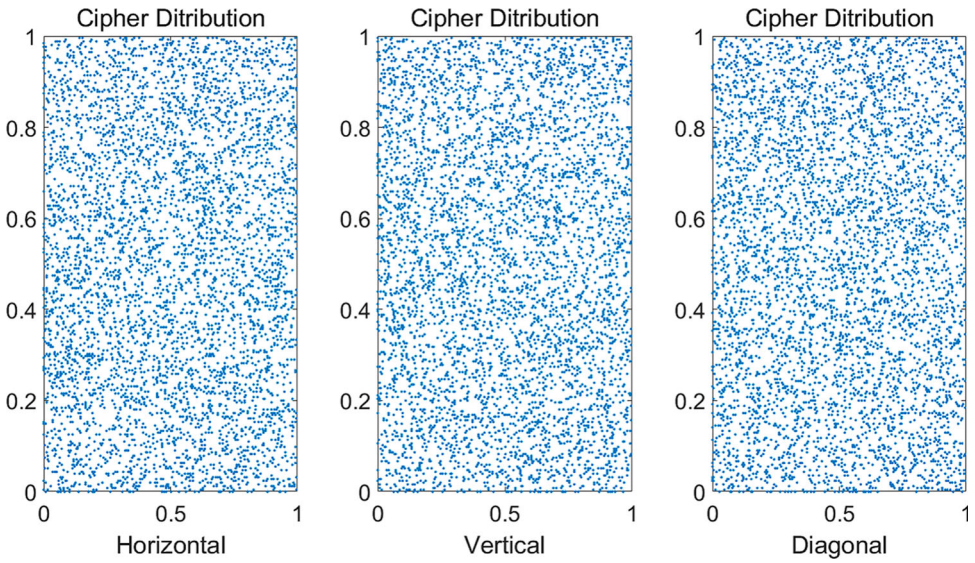


Fig. 11 Distributions of the cipher pixels

Table 6 Entropies of the plain image and cipher image

Item	The plain imager	The cipher image	Theoretical value
Entropy	7.2865	7.9917	8.0000

Table 7 The similarity indicators of the two ciphertexts

Index	The proposed algorithm	Theoretical values
CCOR	− 0.0011	0.0000
SSIM	0.0047	0.0000
NPCR	99.6166	99.6094%
UACI	33.4784	33.4636%
BACI	26.7849	26.7712%

6.5 Resistance to chosen-plaintext attack

In the early days of image encryption, simple pixels scrambling or pure sequence encryption was mainly used. Several image encryption schemes are vulnerable to chosen-plaintext attacks due to their limited structure complexity and robust stability. We use a compound pattern of diffusion, scrambling, and nonlinear transformation in this work in order to ensure the structural complexity of the encryption system. It is concluded in Sect. 6.4 that this composed mode is quite sensitive to plaintext. Particularly, both scrambling and diffusion ciphers are generated using secure chaos, multiple nonlinear transforms, and plaintext-related technology. Additionally, the ciphers include random components. As a result, the algorithm's structural security has been greatly enhanced. The measures outlined above provide the algorithm with powerful barriers against the attack of the chosen-plaintext.

6.6 Comparison with other schemes

A comparison is made between the proposed algorithm and the secured performance indicators in References [9, 28, 29] and [29] in this subsection. A summary of the results can be found in Table 8. Based on the data, two indicators of our algorithm are more effective than others. Regarding the other indicators, the proposed algorithm is neither the best nor the worst. All other indicators of all schemes approximate the theoretical values quite closely. The slight differences between them have no substantive impact on the security. We rank every performance indicator as a score in order to evaluate these encryption schemes comprehensively. In general, the smaller the difference between each index and the theoretical value, the higher the score. As a general rule, 5 points are awarded for the best, 4 points for the suboptimal, and 1 point for the worst performance. Table 9 provides a detailed listing of the rating scores. Table 9 indicates that the proposed algorithm has overall comparative advantages. Particularly, the algorithm has some innovative characteristics, such as the mode of hiding image information, the application of multiple nonlinear transforms, and the composed algorithm structure.

The purpose of this paper is only to simulate the encryption of conventional images, due to space constraints. Due to the fact that the encryption and decryption algorithm is implemented at the pixel level, the proposed encryption and decryption algorithm is also suitable for images in other formats or forms, including images and compressed images.

Table 8 Performance data of different algorithms

	Ref. [9]	Ref. [28]	Ref. [29]	Ref. [29]	Our algorithm	Theoretical value
Key space	798 bit	259 bit	449bit	144 bit	1024 bit	> = 128 bit
Information entropy	7.9993	7.9979	7.9921	7.9910	7.9917	8.0000
NPCR	99.5906%	99.5892%	99.5961	99.6021%	99.6166	99.6094%
UACI	33.4489%	33.4554%	33.4480	33.4521%	33.4784	33.4636%
BACI	26.7530	26.7545	26.7788	26.7669	26.7849	26.7712%

Table 9 Indicator scores and the comprehensive evaluation of algorithm security

	Ref. [9]	Ref. [28]	Ref. [29]	Ref. [29]	Our algorithm
Key space	4	2	3	1	5
Information entropy	5	4	3	1	2
NPCR	2	1	3	4	5
UACI	2	5	1	4	3
BACI	1	2	4	5	3
Total score	14	14	14	11	18

7 Conclusions

A combined image information hiding scheme is presented in this paper. We have developed a new chaos, multiple nonlinear transforms, a new diffusion mode for pixels, and plaintext-related ciphers in order to support this scheme. As a result of the application of nonlinear transforms and the mechanism of plaintext-related functions, the proposed algorithm possesses a high structural complexity. According to the results of the experiment, the new cryptosystem is capable of passing the international standard test for pseudo-random sequences. Moreover, the performance evaluations indicate that the algorithm is sufficiently secure to withstand a variety of attacks. Compared to other image encryption methods, our proposed algorithm demonstrates some local advantages and can serve as a candidate for image information hiding.

In this paper, the basic flow of the algorithm is as follows: image composition—composite image encryption—encrypted image decryption—decrypted image decomposition. The following composed encryption and decryption modes may also be considered: single image encryption—encrypted image composition—composed image decryption—image decomposition. In another paper, we will discuss the algorithm and security of this encryption mode. Furthermore, we propose a new key generation scheme and encryption algorithm based on a combination of neural network, generalized chaotic map, and nonlinear transformation. With this new idea, we hope to be able to deduce the keys of better statistical properties and to develop novel algorithms that are more secure.

Acknowledgements The work was partially supported by the National Natural Science Foundation of China [No. 61702153]. We would like to express our sincere gratitude to those who have provided us with assistance.

Declarations

Conflict of interest As far as the publication of this paper is concerned, the authors declare that there are no conflicts of interest.

References

1. Xu M, Tian Z (2019) A novel image cipher based on 3D bit matrix and Latin cubes. Inf Sci 478:1–14
2. Nkandeu YPK, Tiedeu A (2019) An image encryption algorithm based on substitution technique and chaos mixing. Multimed Tools Appl 78(8):10013–10034
3. Ravichandran D, Praveenkumar P, Rayappan JBB, Amirtharajan R (2016) Chaos based crossover and mutation for securing DICOM image. Comput Biol Med 72:170–184
4. Preishuber M, Hütter T, Katzenbeisser S, Uhl A (2018) Depreciating motivation and empirical security analysis of chaos-based image and video encryption. IEEE Trans Inf Forensics Secur 13(9):2137–2150
5. Zhang Y (2018) The unified image encryption algorithm based on chaos and cubic S-Box. Inf Sci 450:361–377
6. Jin X, Yin S, Liu N, Li X, Zhao G, Ge S (2018) Color image encryption in non-RGB color spaces. Multimed Tools Appl 77(12):15851–15873
7. Liang X, Tan X, Tao L, Hu B (2019) Image hybrid encryption based on matrix nonlinear operation and generalized Arnold transformation. Int J Pattern Recognit Artif Intell 33(06):1954022
8. Hua Z, Zhou Y, Huang H (2019) Cosine-transform-based chaotic system for image encryption. Inf Sci 480:403–419
9. Mansouri A, Wang X (2020) A novel one-dimensional sine powered chaotic map and its application in a new image encryption scheme. Inf Sci 520:46–62
10. Zhang Y, Chen A, Tang Y, Dang J, Wang G (2020) Plaintext-related image encryption algorithm based on perceptron-like network. Inf Sci 526:180–202
11. Zhang Y (2016) Chaotic digital image cryptosystem [M]. Tsinghua University Press, Beijing

12. Ritwik MG, Krishna D, Sahoo A (2017) Cryptanalysis of image encryption using traditional encryption techniques. Image Vis Comput 23(5):89–97
13. Cheng P, Yang H, Wei P, Zhang W (2015) A fast image encryption algorithm based on chaotic and lookup table. Nonlinear Dyn 79(3):2121–2131
14. Liu Q, Li P, Zhang M, Sui Y, Yang H (2015) A novel image encryption algorithm based on chaos maps with Markov properties. Commun Nonlinear Sci Numer Simul 20(2):506–515
15. Zhang Y (2014) A chaotic system based image encryption algorithm using plaintext-related confusion. TELKOMNIKA 12(11):7952–7962
16. Moosazadeh M, Ekbatanifard G (2019) A new DCT-based robust image watermarking method using teaching-learning-Based optimization. J Inf Secur Appl 47:28–38
17. Najafi E, Loukhaoukha K (2019) Hybrid secure and robust image watermarking scheme based on SVD and sharp frequency localized contourlet transform. J Inf Secur Appl 44:144–156
18. Zhang L, Wei D (2020) Image watermarking based on matrix decomposition and gyrator transform in invariant integer wavelet domain. Signal Process 169:107421
19. Ko H-J, Huang C-T, Horng G (2020) Shiuh-Jeng WANG, Robust and blind image watermarking in DCT domain using inter-block coefficient correlation. Inf Sci 517:128–147
20. Lu G, Smidtaite R, Howard D, Ragulskis M (2019) An image hiding scheme in a 2-dimensional coupled map lattice of matrices. Chaos, Solitons Fractals 124:78–85
21. Hazer A, Yıldırım R (2020) Hiding data with simplified diffractive imaging based hybrid method. Opt Laser Technol 128:106237
22. Shulei Wu, Chen H (2020) Smart city oriented remote sensing image fusion methods based on convolution sampling and spatial transformation. Comput Commun 1571:444–450
23. Mustafa HT, Zareapoor M, Yang J (2020) MLDNet: multi-level dense network for multi-focus image fusion. Signal Process Image Commun 85:115864
24. Rukhin A et al. (2020) A statistical test suite for random and pseudorandom number generator for cryptographic applications. Special Publication 800–22 Revision 1a. National Institute of Standards and Technology (NIST). https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-22r1a.pdf
25. Zhao B, Chen M, Zou FS et al (2018) Proficiency in MATLAB-Science computation and the application of data statistics. Posts and Telecommunications Press, Beijing
26. Guo FM, Tu L (2015) Application of chaos theory in cryptography [M]. Beijing Institute of Technology Press, Beijing
27. Xue HW, Du J, Li SL, Ma WJ (2018) Region of interest encryption for color images based on a hyperchaotic system with three positive Lyapunov. Opt Laser Technol 106:506–516
28. Contour Detection and Image SegmentationResources [DB/OL]. (2020) https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html.
29. Chen YC, Ye RS (2017) A novel image Encryption algorithm based on improved standard mapping. Comput Sci Appl 7(8):753–773