



OPEN

## Color image encryption based on chaotic compressed sensing and two-dimensional fractional Fourier transform

Xingyuan Wang<sup>✉</sup> & Yining Su

Combining the advantages of structured random measurement matrix and chaotic structure, this paper introduces a color image encryption algorithm based on a structural chaotic measurement matrix and random phase mask. The Chebyshev chaotic sequence is used in the algorithm to generate the flip permutation matrix, the sampling subset and the chaotic cyclic matrix for constructing the structure perceptual matrix and the random phase mask. The original image is compressed and encrypted simultaneously by compressed sensing, and re-encrypted by two-dimensional fractional Fourier transform. Simulation experiments show the effectiveness and reliability of the algorithm.

Images can truly show us the real world, especially color images are becoming more and more important in life. For a color image, it has three elements of R, G, and B, so it contains more information than grayscale images, its original features, large amount of data, high redundancy, high correlation between pixels. In order to protect image information, major contributions have been made in the fields of steganography<sup>1,2</sup>, and encryption<sup>3-5</sup>. In recent years, chaos has some ideal cryptographic characteristics such as initial value sensitivity and pseudo-randomness, which makes the chaotic encryption scheme widely used<sup>6-17</sup>. For example, Ahmad et al. proposed a chaos-based high-key image encryption scheme that makes even one round of encryption, the key space is very large<sup>9</sup>. For the small number of keys and simple key transmission method, Wu et al. used 4D cat mapping and elliptic curve ElGamal for asymmetric encryption<sup>10</sup>. Considering the increase in image size, Chai et al. proposed to divide the image into blocks for scrambling and diffusion<sup>11</sup>.

Donoho proposed a new sampling reconstruction technology, this technology is called compressed sensing<sup>18</sup>. Various CS encryption schemes with high efficiency and low data volume have been proposed, but CS-based encryption schemes are not resistant to selective plaintext attacks<sup>19</sup>. Since the measurement matrix satisfies low cross-correlation, a random matrix such as a Gaussian random matrix has a large-capacity memory and high complexity<sup>20,21</sup>. Therefore, in order to eliminate the shortcomings of random matrix, an encryption scheme based on CS and chaotic system<sup>22-25</sup> is designed, and deterministic matrix is introduced instead of random matrix<sup>26,27</sup>. For example, Naidu proposes to use Euler lattice to construct a binary perceptual matrix, but this is limited to medical image and greatly limits the scope of application<sup>28</sup>. Combining the advantages of chaos, this paper proposes a structural chaotic matrix, using Chebyshev map to construct a flip scrambling matrix, a chaotic-based cyclic matrix, and a sampling subset. These three parts are completely determined structures. Although CS is used to reduce the amount of data re-encryption, the characteristics of color images still exist.

Due to the high speed and parallel processing of optical images, a large number of optical-based image encryption schemes have been proposed<sup>29-31</sup>. Although classical optical encryption based on double random phase mask is easy to attack<sup>32</sup>, it lays a foundation for subsequent optical encryption schemes. Use FrFT, Fresnel transform to enhance security<sup>33,34</sup> to overcome various attacks. For example, Farah et al. proposed a new method for encrypting optical images using fractional Fourier transform, DNA sequence manipulation, and chaos theory. The encryption method has high security but high complexity and cost<sup>35</sup>. In order to avoid high complexity and too large data to transmit, a combination of CS and optical encryption is proposed<sup>36-40</sup>. Zhang et al. proposed a fast and effective color image encryption scheme based on two-dimensional compressed sensing and fractional Fourier transform<sup>36</sup>. To solve the risk of linear transformation in image encryption technology, Zhou et al. proposed an image encryption scheme that combines compressed sensing and nonlinear fractional Merlin transformation<sup>37</sup>. In order to reduce the amount of data, the algorithm uses Chebyshev map to generate chaotic sequences to construct a deterministic structured sensing matrix and a random mask. The image is first

School of Information Science and Technology, Dalian Maritime University, Dalian 116026, China. ✉email: wangxy@dlut.edu.cn

compressed and subjected to two-dimensional fractional Fourier transform. Through simulation experiments, the algorithm has good security performance and can resist common attacks.

To overcome the above difficulties, we propose a structure-sensing matrix and two-phase random mask color image encryption algorithm. The main contributions of this paper are: (1) Combining the advantages of a structured random perception matrix and a chaotic structure, a random perception matrix with a secure structure is proposed. The novelty of the matrix is that the original signal is flipped, the flipped coefficient is measured quickly and pseudo-randomly, and the final sample is obtained by sampling. (2) In real life, the utilization rate of color images is higher, but the color image data has a large amount of data, high redundancy, and high correlation between pixels. Our proposed encryption scheme can overcome these difficulties. Using CS can simultaneously compress and Encryption, reducing data volume and degrading transmission costs. (3) The proposed encryption scheme overcomes the difficulty that the previous Fourier transform-based encryption schemes are easily attacked. The two-dimensional fractional Fourier transform is used to increase the key space. The experimental results and security analysis show the security of the algorithm.

### Algorithm foundation

**Compressed sensing.** Compressed sensing theory is a brand new signal sampling compression. Suppose that the signal  $f$  of size  $N \times 1$  can be expressed as sparse basis  $\Psi$ :

$$f = \Psi s \tag{1}$$

$\Psi$  is a sparse orthogonal base of size  $N \times N$ , and  $s$  is a sparse coefficient. If the coefficient  $s$  has  $k \ll N$  non-zero coefficients, then  $\Psi$  is said to be the sparse basis of the signal  $f$ .

The sampling process is a linear projection of the signal  $f$ :

$$y = \Phi f = \Phi \Psi s = As \tag{2}$$

where  $\Phi$  is a projection matrix of size  $m \times N$ .  $y$  is a linear measure of size  $m \times 1$  ( $m < N$ ). In addition, the sensing matrix  $A$  should satisfy the RIP criteria<sup>41</sup>:

$$(1 - \delta_k) \|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_k) \|x\|_2^2 \tag{3}$$

wherein the equidistance constant  $\delta_k \in (0, 1)$ ,  $k$  is the number of coefficients  $s$  that are not zero.

The signal  $f$  measured using  $\Phi$  can be recovered from  $y$ .

$$\min \|s\|_0 \text{ s.t. } y = As \tag{4}$$

In order to solve the above non-convex problems, many reconstruction algorithms have been proposed, such as orthogonal tracking algorithm (OMP)<sup>42</sup>, smooth norm ( $SL_0$ )<sup>43</sup> and so on.

**Fractional Fourier transform.** In order to improve the security of the system, the Fourier transform is improved to a fractional Fourier transform, and the required angle is used as a key to increase the key space and key sensitivity. First, the mathematical definition of one-dimensional fractional Fourier transform is introduced<sup>44</sup>:

$$F^p\{f(x)\}(u) = \int_{-\infty}^{+\infty} k_p(x, u) f(x) dx \tag{5}$$

where the kernel function  $k_p(x, u)$ ,

$$k_p(x, u) = \begin{cases} A_p \exp[i\pi(u^2 \cot \theta - 2ux \csc \theta + x^2 \cot \theta)], & p \neq 2n \\ \delta(x + u), & p = 4n + 2 \\ \delta(x - u), & p = 4n \end{cases}$$

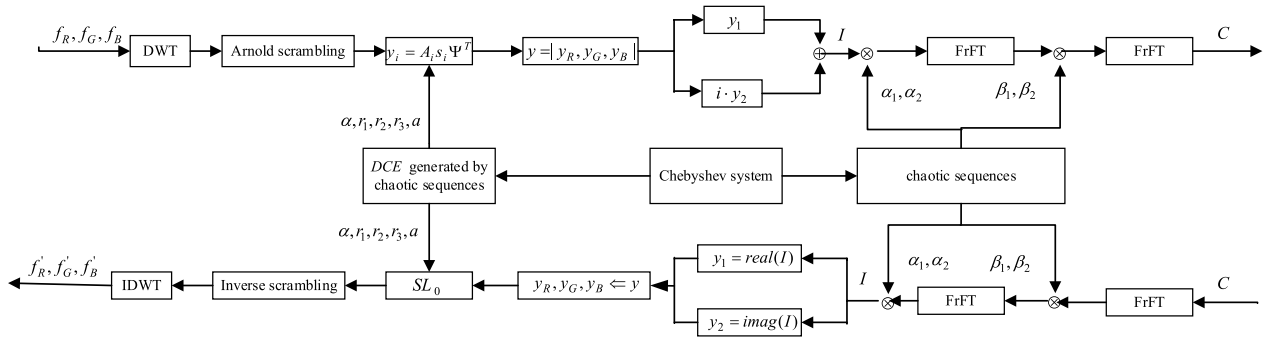
$$A_p = \sqrt{1 - i \cot \theta}, \theta = p\pi/2, p \neq 2n$$

where  $F^p\{f(x)\}(\cdot)$  is the  $p$ -order Fractional Fourier transform of signal  $f(x)$ , and  $p$  is the fractional order of Fractional Fourier transform.  $x$  and  $u$  respectively represent the input domain coordinates and the  $p$ -order fractional domain coordinates.  $\theta$  represents the rotation angle of the time–frequency plane,  $\delta(\cdot)$  represents the impulse function, and  $n$  is a positive integer.

The two-dimensional fractional Fourier transform is a generalization of the one-dimensional fractional Fourier transform. In the field of optics, a two-dimensional fractional Fourier transform is realized by optical instruments, which is defined as follows<sup>45</sup>:

$$F^{p1,p2}(u, v) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) k_{p1,p2}(x, y, u, v) dx dy \tag{6}$$

$f(x, y)$  is the original signal,  $k_{p1,p2}(x, y, u, v)$  is a kernel function,  $k_{p1,p2}(x, y, u, v) = k_{p1}(x, u) \times k_{p2}(y, v) = \frac{\sqrt{(1-i \cot \alpha)\sqrt{(1-i \cot \beta)}}}{2\pi} \exp[\frac{i(x^2+u^2)}{2 \tan \alpha} - \frac{ixu}{\sin \alpha}] \exp[\frac{i(y^2+v^2)}{2 \tan \beta} - \frac{iyv}{\sin \beta}]$   $p1, p2$  is expressed as a transformation order in the  $x, y$  directions, and  $\alpha, \beta$  represents a rotation angle.



**Figure 1.** The proposed color image encryption and decryption scheme.

**Chebyshev chaotic map.** Since chaotic systems can generate pseudo-random sequences and are sensitive to initial values. In order to construct the sensing matrix, the Chebyshev chaotic system is used in this paper. The mathematical definition is as follows<sup>46</sup>:

$$r_{i+1} = \tau(r_j) = \cos(\alpha \cdot \arccos(r_j)) \tag{7}$$

$\alpha$  is a positive integer,  $r_j \in [-1, 1]$ , when  $r_0 \in [-1, 1]$  is the initial value,  $R_1 = \{r_j = \tau^j(r_0)\}, j = 0, 1, 2, \dots, r_0$  is a chaotic sequence.  $\alpha, r_0$  as the key of the cryptosystem. Chebyshev chaotic sequences are used to construct sensing matrices and random masks. That is to say, in the secure channel, we transmit the key instead of the perceptual matrix and the random mask. That is, the chaotic system controls the entire process.

### Image encryption and decryption process

Algorithm 1

- (1) The Chebyshev chaotic system generates a chaotic sequence  $R_1$ , and records the position sequence  $Y$  corresponding to the sequence  $R_1$ .
- (2) The sequence  $R_1$  is sorted in ascending order, and the corresponding position sequence  $Y$  becomes chaotic sequence  $Y_1$  as the order changes.
- (3) Select the first  $m$  numbers of the chaotic sequence  $Y_1$  to get a subset.
- (4) Select  $m$  rows of matrix according to subset.

Figure 1 is the process of color image encryption and decryption based on compressed sensing and two-dimensional fractional Fourier transform.

**Encryption process.** First, the Discrete Walsh transform (DWT) standard orthogonal basis sparse representation is performed on the three components of the color image from two directions:

$$f_i = \Psi s_i \Psi^T, i = R, G, B \tag{8}$$

The three components of the color image are processed separately, and no other formats need to be converted.

The sparse signal is then measured using a chaotic sequence constructed by Chebyshev to construct a measurement matrix  $\Phi$ . The measurement matrix is defined as follows:

$$\Phi = \sqrt{\frac{N}{M}} DCE \tag{9}$$

Among them,  $N$  is the width of the image,  $M = CR \times N$ . The coefficient  $\sqrt{\frac{N}{M}}$  is normalized to  $DCE$  such that the energy of the measured value is close to the energy of the original signal.  $D$  is a random sampling operator, which is a random sample of the  $m$  rows of  $CE$  according to a subset  $\{1, 2, \dots, n\}$ . In this paper, a chaotic system-based permutation algorithm is proposed for sampling. The essence of the permutation algorithm is to randomly select  $m$  rows using pseudo-random sequences generated by chaotic systems (algorithm 1).

$C$  is a cyclic matrix based on chaotic sequences. The size of  $C$  is  $N \times N$ , defined as follows.

$$C = \frac{1}{\sqrt{nv(r)}} \begin{pmatrix} r_{(n-1)} & r_{(n-2)} & \cdots & r_0 \\ r_0 & r_{(n-1)} & \cdots & r_1 \\ \vdots & \vdots & \ddots & \vdots \\ r_{(n-2)} & r_{(n-3)} & \cdots & r_{(n-1)} \end{pmatrix} \tag{10}$$

where  $r_{(i-1)}$  is the  $i$ th element of the chaotic sequence  $R_1$ ,  $v(r)$  is the variance of the matrix  $C$ ,  $\frac{1}{\sqrt{nv(r)}}$  is for normalizing  $C$ ,  $C$  is for passing important information in  $f$  to the measured value, and chaos-based cyclic matrix  $C$  is only required for  $n$  Element storage, which reduces memory requirements.

$E$  is a diagonal matrix in which diagonal elements are determined by chaotic sequences.

$$E_{i,i} = \begin{cases} +1, & 0 \leq r_{(i)} \leq 1 \\ -1, & -1 \leq r_{(i)} \leq 0 \end{cases} \quad i = 1, 2, \dots, N \tag{11}$$

$r_{(i)}$  is the  $i$  element of the sequence  $R_1$ . According to the nature of the Chebyshev sequence, the probability that the diagonal element  $E_{i,i}$  in  $E$  is equal to 1 or  $-1$  is the same. So  $E$  is equivalent to a pseudo-randomizer that can change the sign of the signal.

Since the sampling subset  $D$ , the diagonal matrix  $E$ , and the chaotic sequence circulant matrix  $C$  are all generated by the Chebyshev chaotic map, the  $\Phi$  is a certain measurement matrix. To generate different measurement matrices, only the initial conditions of the Chebyshev system need to be changed.

Compressed sensing process is as follows:

$$y = \Phi \Psi s \Psi^T = A s \Psi^T \tag{12}$$

The reconstruction algorithm (OMP or  $SL_0$ ) can be used to recover  $\Phi$ , and finally the original signal is obtained by performing the inverse operation of the sparse coefficient and the sparse basis.

Finally, the measured image is subjected to two-dimensional fractional Fourier transform encryption using two random phase masks, which are generated based on the chaotic sequence. If the fractional Fourier transform is used directly, the data will explode, so CS has a major role in overcoming this defect.

The detailed encryption operations are as follows:

Step 1: The color image can be divided into three images according to the RGB component, respectively denoted as  $f_R, f_G, f_B \in R^{N \times N}$ . They are respectively sparsed by the sparse base  $\Psi$  in the wavelet transform domain to obtain  $f_1, f_2, f_3$ . Then perform Arnold scrambling on  $f_1, f_2, f_3$  to get  $f'_1, f'_2, f'_3$ . Set the threshold TS, modify the elements of  $f'_1, f'_2, f'_3$ , if the absolute value of the element is less than TS, change the element value to 0, get  $f''_1, f''_2, f''_3$ .

Step 2: Generating measurement matrix  $\Phi$ , the specific process is as follows:

Given  $\alpha_1, r_1, r_2, r_3$  as the initial condition, the Chebyshev chaotic map is taken. The chaotic sequence  $R_i = \{r_0, r_1, \dots, r_{N-1}\}, i = 1, 2, 3$  is generated, and the matrix  $C_i \in R^{N \times N}$  is obtained according to Eq. (10) by  $R_i$ . Obtain the matrix  $E_i \in R^{N \times N}$  according to Eq. (11), obtain the sampling subset  $D_i$  according to the algorithm 1, and finally obtain the  $\Phi_i$  according to Eq. (9).

Step 3: The measurement matrix is measured in the three (stained) images of Eq. (8), which is compressed sensing. The measurement matrix  $\Phi_i$  measures the three thinned images in Eq. (8), that is, the compressed sensing. The three measured values are obtained as  $y_R, y_G, y_B \in R^{m \times N}$ .

Step 4: Next, the two measured images are subjected to two-dimensional fractional Fourier transform encryption.

Take the three measured images as an image  $F$ , the size is  $m \times 3N$ , divide  $F$  into two parts from the middle, the left part is  $y_1$ , the right part is  $y_2$ , their size is  $m \times \frac{3}{2}N$ , and the two parts are combined into a complex number,  $y_1$  is the real part,  $y_2$  is the imaginary part.

$$I(x, y) = y_1(x, y) + y_2(x, y)i \tag{13}$$

$I(x, y)$  is a complex image.

According to Eqs. (14)–(18),  $as, r_{11}, r_{22}, r_{33}$  is calculated as the initial value to iterate the chaotic system  $L + m \times N$  times, and the previous  $L$  times are discarded to obtain the chaotic sequence  $L_i, i = 1, 2, 3$ .

$$a = \text{sum}(f(:)) / (N \times N \times 255) \tag{14}$$

$$as = a - \text{floor}(a) \tag{15}$$

$$r_{11} = \text{mod}((as + r_1), 1) \tag{16}$$

$$r_{22} = \text{mod}((as + r_2), 1) \tag{17}$$

$$r_{33} = \text{mod}((as + r_3), 1) \tag{18}$$

$$L_i = \{l_0, l_1, \dots, l_{m \times \frac{3}{2}N-1}\}, i = 1, 2, 3 \tag{19}$$

$L_1, L_2$  is used as a random phase mask for fractional Fourier transform, and the image is encrypted as:

$$C(x, y) = FrFT^{\alpha_1, \alpha_2} \{FrFT^{\beta_1, \beta_2} \{I(x, y) \exp[iL_1(x, y)]\} \exp[iL_2(x, y)]\} \tag{20}$$

$L_1(x, y), L_2(x, y)$  is a two-phase random mask, and  $\alpha_1, \alpha_2, \beta_1, \beta_2 \in [-2, 2]$  is a fractional order in the  $x, y$  direction, respectively.

Step 5: Perform global scrambling, ascending  $L_3$ , record the changed position  $w$ ,  $w$  as the address code to reorder the image  $C$  to achieve scrambling effect. Convert the image  $C$  into a one-dimensional matrix in the order of the columns, and scramble the one-dimensional matrix according to the following rules.

$$C_1(i) = C(w(i)), i = 1, 2, \dots, m \times \frac{3}{2}N \quad (21)$$

Then, the scrambled matrix is converted into a two-dimensional matrix, and after being scrambled, the ciphertext is finally output as  $C_2$ .

**Decryption process.** The decryption step is the inverse process of encryption.

Step 1: The anti-scrambling process, imitating step 5 of the encryption process, generates a chaotic sequence  $L_i, i = 1, 2, 3$  according to the key  $as, r_{11}, r_{22}, r_{33}$ , sorts  $L_3$  to obtain an address code  $w$ , converts  $C_2$  into a one-dimensional matrix  $C_1$ , and the assignment direction becomes:

$$C(w(i)) = C_1(i), i = 1, 2, \dots, m \times \frac{3}{2}N \quad (22)$$

Convert to two-dimensional matrix  $C$ .

Step 2: Decrypt out  $I(x, y)$ :

$$I(x, y) = FrFT^{-\alpha_1, -\alpha_2} \{FrFT^{-\beta_1, -\beta_2} \{C(x, y) \exp[-iL_1(x, y)]\} \exp[-iL_2(x, y)]\} \quad (23)$$

Calculate the resulting complex-valued image and get two parts,

$$\begin{cases} y_1(x, y) = \text{real}\{I(x, y)\} \\ y_2(x, y) = \text{imag}\{I(x, y)\} \end{cases} \quad (24)$$

Step 3: Think of  $y_1, y_2$  as an image, then divide it into three images, use  $SL_0$  algorithm to reconstruct the image, Arnold inverse scrambling and then perform wavelet inverse transform to obtain  $f_R, f_G, f_B \in R^{N \times N}$ . Finally, the decrypted color image  $f$  is obtained.

## Simulation results and performance analysis

**Simulation result.** In order to verify the feasibility and effectiveness of the encryption scheme, the security performance tests in this paper include key space, key sensitivity, correlation analysis, histogram analysis and various common attack tests. As shown in Fig. 2, matlab simulation experiments were performed using “House”, “Baboon”, “Pepper” and “Airplane” color images of size  $256 \times 256 \times 3$ , the corresponding TS are 10, 20, 10, 10. Figure 2a1–d1 are original images, Fig. 2a2–d2 are results of 2D CS, and Fig. 2a3–d3 are amplitudes of the encrypted image, the size of which is  $170 \times 384$ . The compression ratio is 0.664. Figure 2a4–d4 are the phases of the encrypted image, and Fig. 2a5–d5 are the decrypted images.

**PSNR analysis.** Restoring an image includes decoding and reconstruction, using FrFT to decode under the correct key, and solving the l1 norm minimized reconstructed image is only similar to the plaintext image, so the quality of the decrypted image is evaluated using PSNR, and the formula is as follows:

$$PSNR = 10 \log_{10} \left( \frac{255 \times 255}{MSE} \right) \quad (25)$$

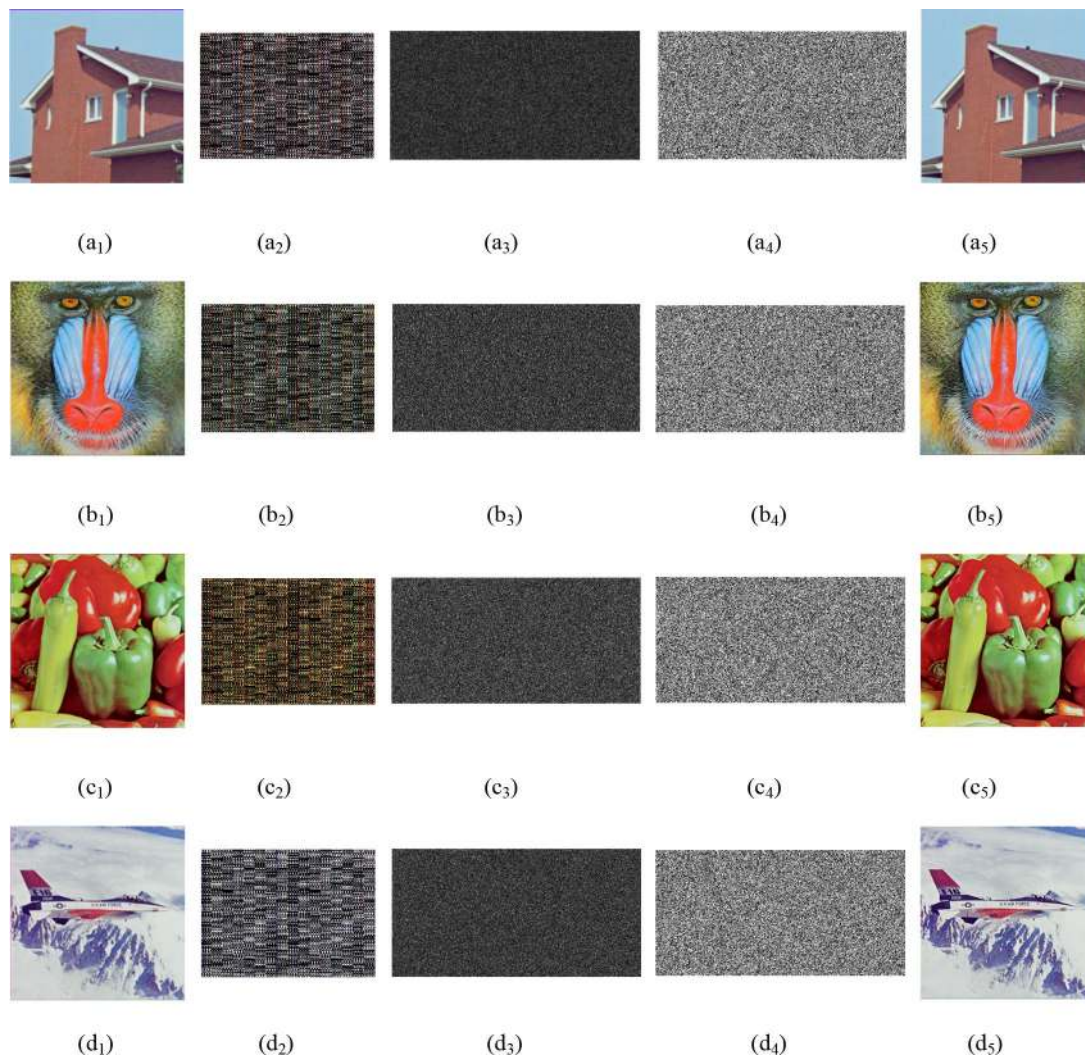
Of which,

$$MSE = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N (f(i, j) - \tilde{f}(i, j))^2 \quad (26)$$

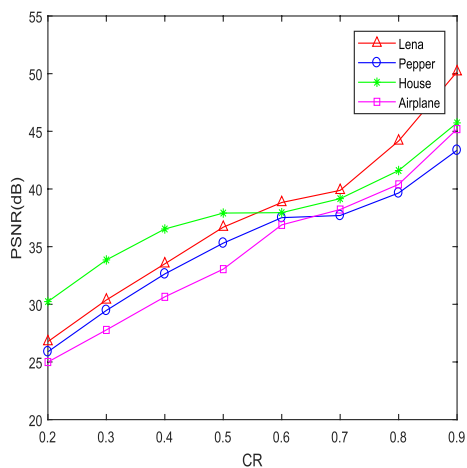
$f(i, j), \tilde{f}(i, j)$  denotes the original image and the decrypted image respectively. Under the correct key, the decrypted image is as shown in Fig. 2. The PSNR of the five images is 38.8780, 37.9466, 28.5954, 37.6960, 38.0903, respectively. Therefore, the image decrypted by this algorithm is good. Figure 3 shows the PSNR values of different CRs of Lena, Pepper, House and Airplane images. The larger the CR, the larger the PSNR value, and the better the reconstruction effect. Table 1 shows the reconstruction effect of Pepper image of different CR. It can be seen from Table 1 that the compression performance of this algorithm is good. Taking the Lena as an example, the Table 2 lists the reconstruction performance comparison between this algorithm and other algorithms. With the same CR from the Table 2, the reconstruction quality of this algorithm is better.

**Histogram analysis.** Histogram analysis of important indicators of image security after image encryption<sup>47</sup>. As shown in Fig. 4a1–a3, b1–b3, c1–c3 represents the R, G, and B components of the three color images of “Lena”, “House” and “Baboon”, respectively,  $a_4$ – $c_4, a_5$ – $c_5$  respectively represent the amplitude and phase after the three images are encrypted. Obviously, the histograms of the R, G, and B components of the three original images are different from each other, but different images are encrypted with very similar histograms, that is, the attacker cannot obtain useful messages from the ciphertext histogram.


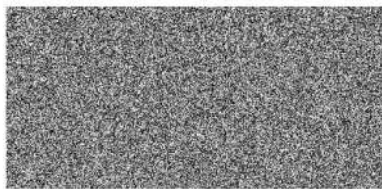

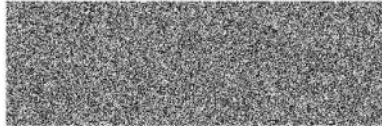

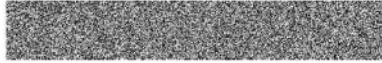

**Adjacent pixel correlation.** Randomly select the plaintext images R, G, B three channels and the amplitude and phase of the ciphertext on 2000 pairs of pixels for correlation testing<sup>48</sup>. The simulation results are shown in Fig. 5, from  $a_1$ – $a_3, d_1$ – $d_3$ , it is found that the correlation of the plaintext images in the horizontal, vertical, and diagonal directions is concentrated, showing a clear linear relationship, from  $b_1$ – $b_3, c_1$ – $c_3, e_1$ – $a_3, f_1$ – $f_3$  found that



**Figure 2.** Encrypting and decrypting images: (a<sub>1</sub>)–(d<sub>1</sub>) are the original images “House”, “Baboon”, “Pepper” and “Airplane”, (a<sub>2</sub>)–(d<sub>2</sub>) are the results of 2D CS, (a<sub>3</sub>)–(d<sub>3</sub>) is the amplitude of the encrypted image, (a<sub>4</sub>)–(d<sub>4</sub>) is the phase of the encrypted image, (a<sub>5</sub>)–(d<sub>5</sub>) is the decrypted image.



**Figure 3.** PSNR vs CR for different plain images.

Images	CR	Encrypted image	Decrypted image	PSNR
	0.75			38.35
	0.5			35.30
	0.25			27.89

**Table 1.** PSNR values for different compression ratios.

Plain image	CR	Our	Ref. <sup>22</sup>	Ref. <sup>25</sup>	Ref. <sup>38</sup>
Lena	0.25	28.74	26.52	26.06	–
	0.5	36.68	29.23	29.82	> 25
	0.75	41.94	29.21	29.56	> 29

**Table 2.** The compression performance of different algorithms.

the encrypted image pixel values are evenly distributed and scattered, indicating that the algorithm proposed in this paper makes the statistical features of the plaintext image spread evenly into the ciphertext.

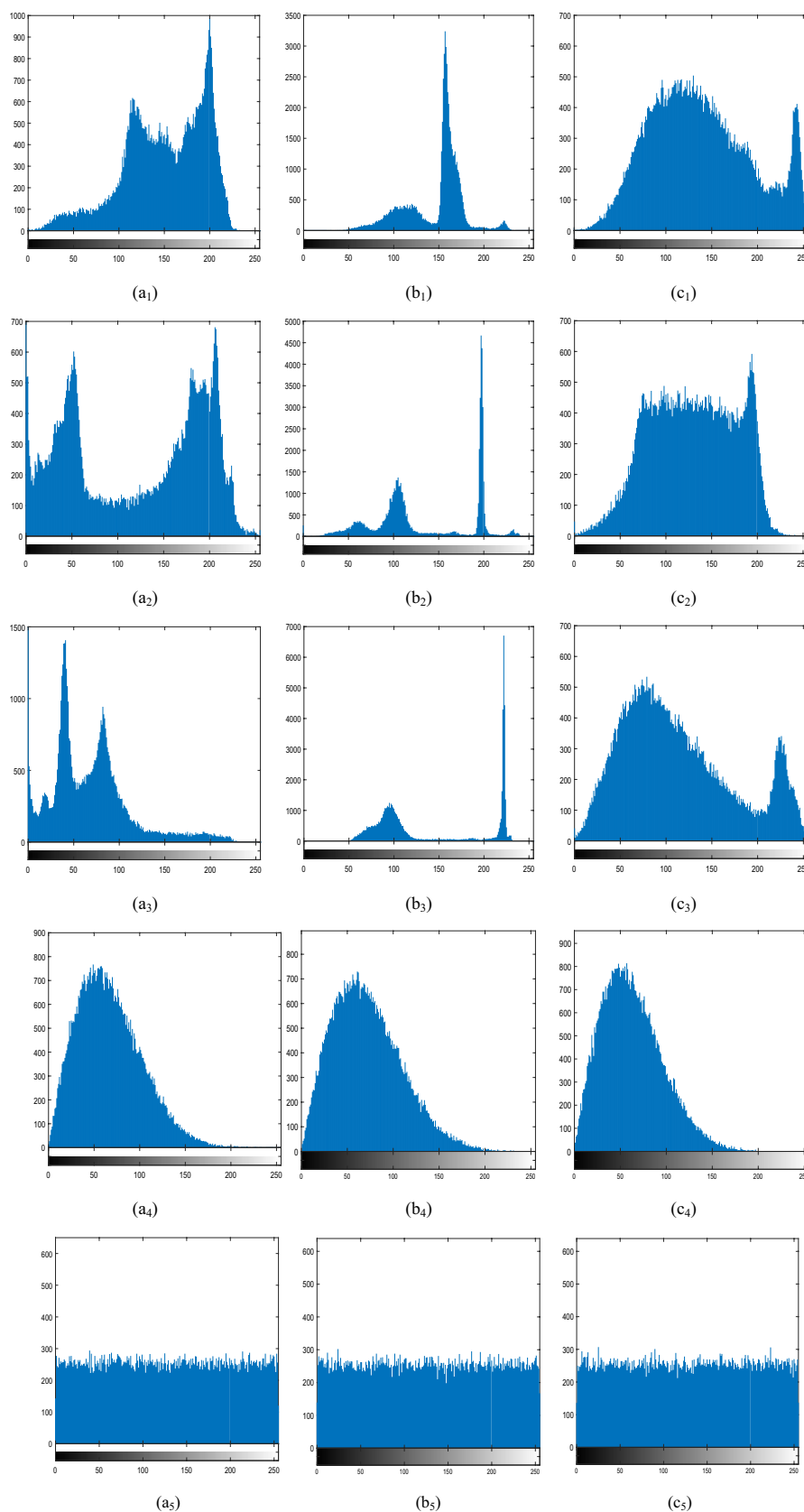
Randomly select the plaintext images R, G, B three channels and the amplitude and phase of the ciphertext on 2000 pairs of pixels for correlation testing. According to Eq. (27), the correlation coefficient of horizontal, vertical and diagonal angles is measured, and the operation is repeated 100 times to calculate the average of the correlation coefficients of horizontal, vertical and diagonal. The final statistical results are shown in Table 3.

$$r_{xy} = \frac{\text{cov}(x, y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{27}$$

Of which,

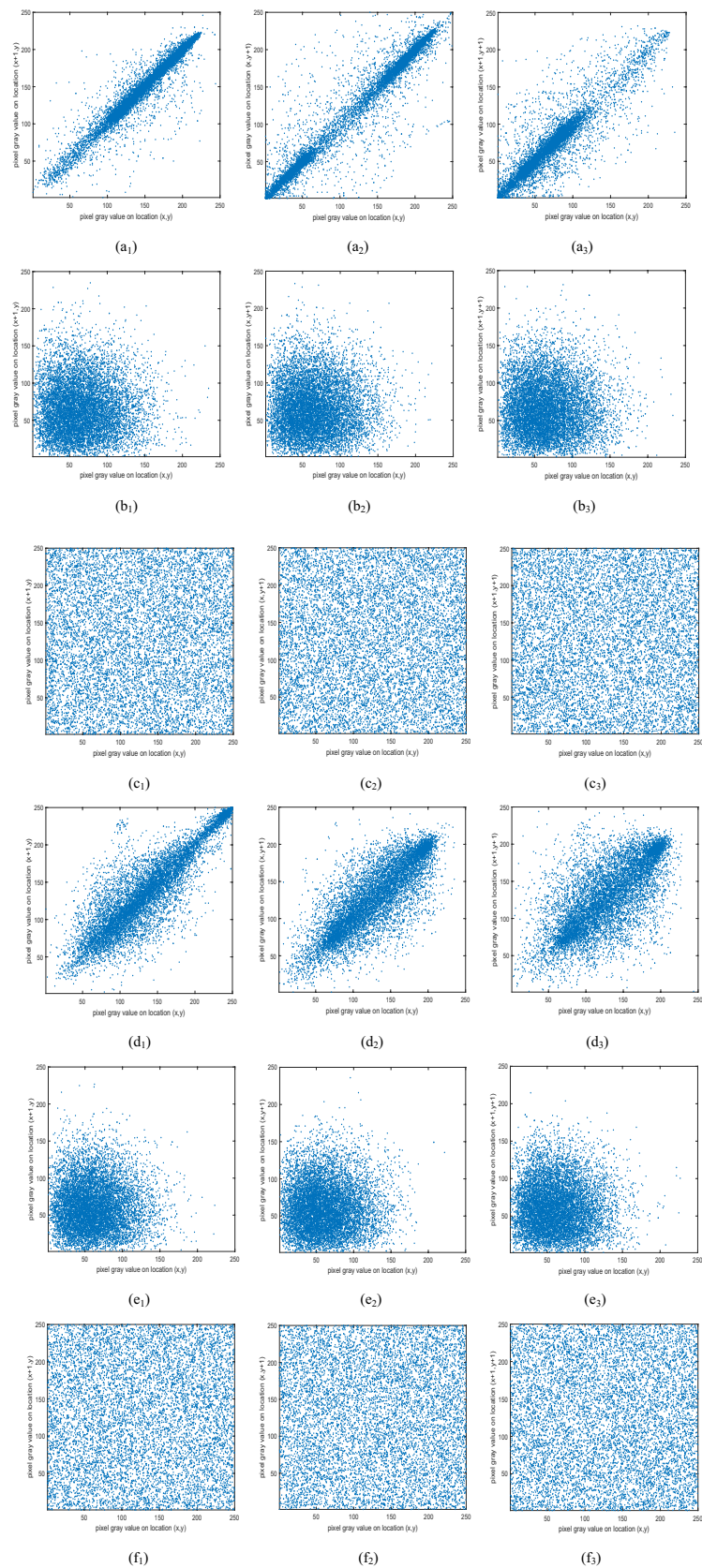
$$\text{cov}(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))(y_i - E(y)), \quad D(x) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x))^2, \quad E(x) = \frac{1}{N} \sum_{i=1}^N x_i.$$

It can be seen from Table 3 that the correlation coefficients of the plain image in the horizontal, vertical, and diagonal adjacent pixels are large. After encryption, the correlation coefficients of the ciphertext in the horizontal, vertical, and diagonal adjacent pixels are small and both at 1%. The algorithm proposed in this paper can effectively reduce the correlation of adjacent pixels. It can be seen from Table 4 that the encrypted correlation



**Figure 4.** The histograms of plaintexts and ciphertexts: The original images “Pepper”, “House” and “Baboon”. (a<sub>1</sub>)–(c<sub>1</sub>) The histogram of original images R component, (a<sub>2</sub>)–(c<sub>2</sub>) The histogram of original images G component, (a<sub>3</sub>)–(c<sub>3</sub>) The histogram of original images B component, (a<sub>4</sub>)–(c<sub>4</sub>) The histogram of amplitude of encrypted images, (a<sub>5</sub>)–(c<sub>5</sub>) The histogram of phase of encrypted images.





**Figure 5.** Horizontal, vertical, diagonal correlation test results for “Pepper” and “Baboon” plaintext images and their ciphertext images: (a<sub>1</sub>)–(a<sub>3</sub>), (d<sub>1</sub>)–(d<sub>3</sub>) represent the horizontal, vertical and diagonal correlation distribution of the plaintext image, (b<sub>1</sub>)–(b<sub>3</sub>), (c<sub>1</sub>)–(c<sub>3</sub>), (e<sub>1</sub>)–(e<sub>3</sub>), (f<sub>1</sub>)–(f<sub>3</sub>) represent the horizontal, vertical and diagonal correlation distribution of the amplitude and phase of the ciphertext, respectively.

Correlation coefficient	The original images			Our encrypted images	
	R	G	B	Amplitude	Phase
<b>Lena</b>					
Horizontal	0.9389	0.9392	0.8932	0.0027	-0.0000
Vertical	0.9677	0.9688	0.9380	-0.0027	-0.0032
Diagonal	0.9090	0.9114	0.8474	0.0003	0.0026
<b>House</b>					
Horizontal	0.9670	0.9800	0.9818	-0.0013	-0.0055
Vertical	0.9353	0.9718	0.9747	0.0033	-0.0014
Diagonal	0.9127	0.9561	0.9621	0.0004	-0.0059
<b>Baboon</b>					
Horizontal	0.9135	0.8027	0.8774	0.0002	-0.0045
Vertical	0.8743	0.7570	0.8651	-0.0001	0.0065
Diagonal	0.8530	0.7010	0.8161	0.0027	-0.0007
<b>Pepper</b>					
Horizontal	0.9526	0.9620	0.9418	0.0035	0.0004
Vertical	0.9554	0.9688	0.9545	-0.0054	0.0023
Diagonal	0.9179	0.9365	0.9097	0.0012	0.0020
<b>Airplane</b>					
Horizontal	0.9107	0.9103	0.9257	0.0021	-0.0008
Vertical	0.8947	0.9048	0.8728	0.0015	-0.0042
Diagonal	0.8292	0.8464	0.8364	-0.0033	-0.0030

**Table 3.** The correlation coefficient of adjacent pixels.

Correlation coefficient	Encrypted images	
	Amplitude	Phase
<b>Lena</b>		
Horizontal	0.0027	-0.0000
Vertical	-0.0027	-0.0032
Diagonal	0.0003	0.0026
<b>Ref.<sup>9</sup></b>		
Horizontal	-	0.0026
Vertical	-	-0.0038
Diagonal	-	0.0062
<b>Ref.<sup>10</sup></b>		
Horizontal	-	0.0001
Vertical	-	0.0089
Diagonal	-	0.0091
<b>Ref.<sup>11</sup></b>		
Horizontal	-	0.0044
Vertical	-	0.0151
Diagonal	-	0.0012
<b>Ref.<sup>36</sup></b>		
Horizontal	0.0127	0.0127
Vertical	0.0101	-0.0271
Diagonal	0.0139	0.0183
<b>Ref.<sup>37</sup></b>		
Horizontal	0.0104	0.0158
Vertical	0.0299	0.0158
Diagonal	0.0062	-0.0339
<b>Ref.<sup>38</sup></b>		
Horizontal	0.2905	-0.0117
Vertical	0.4711	-0.2089
Diagonal	0.2894	0.0301

**Table 4.** Comparison of this algorithm with other algorithms.

Algorithm	R	G	B	Cipher
Lena	7.2775	7.5869	7.0133	7.9959
House	7.6756	7.4794	7.7526	7.9962
Baboon	6.4311	6.5389	6.2320	7.9959
Pepper	7.3449	7.5718	7.1005	7.9959
Airplane	6.8567	6.9602	6.3352	7.9954
Lena in Ref. <sup>9</sup>		7.3441		7.9832
Lena in Ref. <sup>10</sup>		7.7583		7.9912
Lena in Ref. <sup>12</sup>		–		7.9896

**Table 5.** Information entropy of different images.

Algorithm	Proposed algorithm	Ref. <sup>9</sup>	Ref. <sup>11</sup>	Ref. <sup>12</sup>	Ref. <sup>38</sup>
Key space	$10^{144}$	$10^{90}$	$10^{90}$	$10^{45}$	$10^{91}$

**Table 6.** Comparison of key space.

coefficient of this paper is lower than that most algorithms, so the encryption scheme of this paper can resist statistical analysis.

**Information entropy.** Test image randomness using entropy. If the entropy value is closer to 8, it means that the pixels of the image are more uniform. The formula for calculating entropy is as follows:

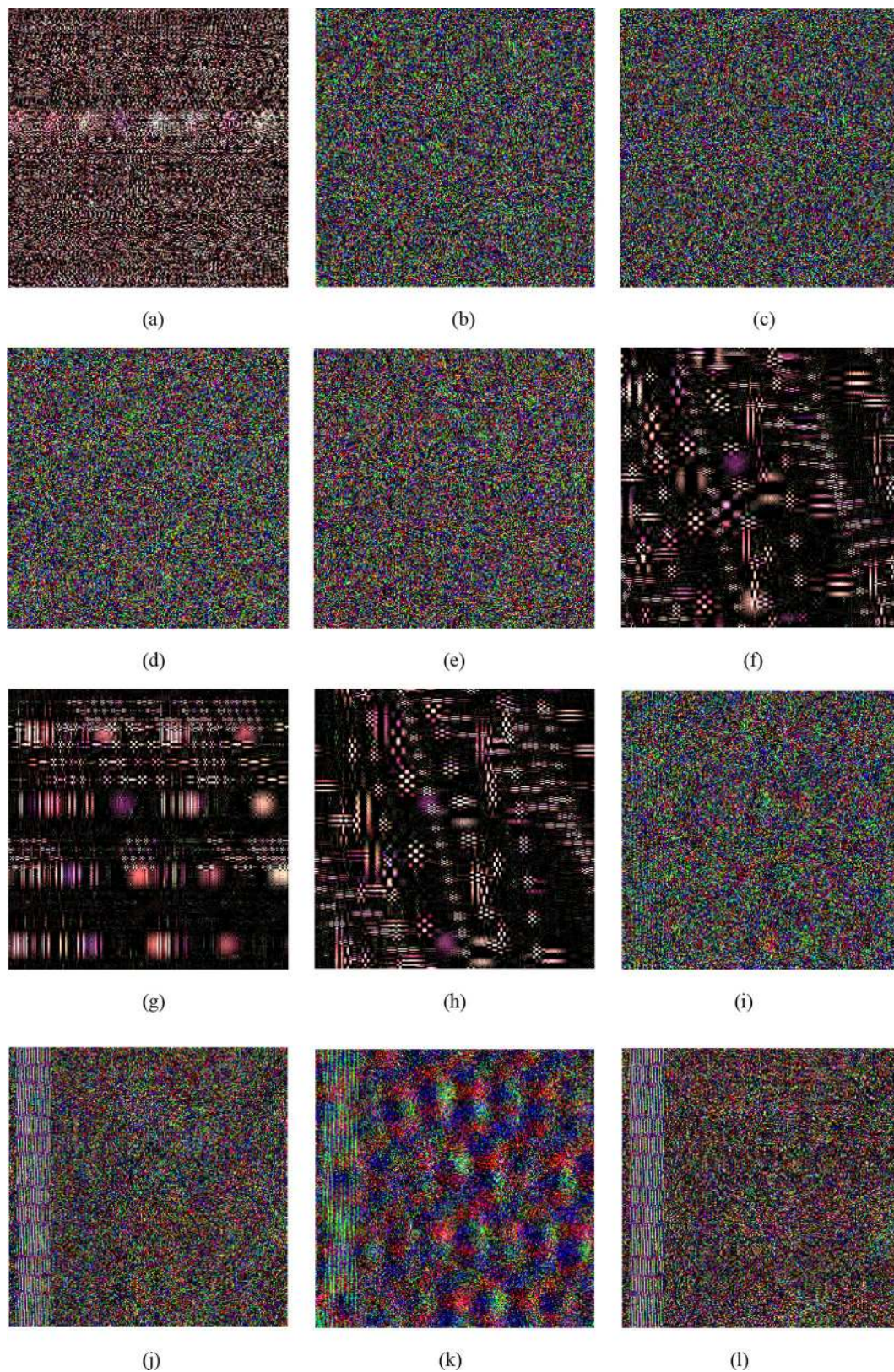
$$H(g) = \sum_{i=0}^{2^L-1} P(g_i) \log_2 \frac{1}{P(g_i)} \quad (28)$$

where  $g$  represents a set of pixels.  $P(g_i)$  is the probability of occurrence of  $g$ , and  $L$  is the total number of  $g_i$ . Table 5 shows the entropy corresponding to different images and comparison with other algorithms. The table shows that the encrypted image is close to 8, which means that it is safe against entropy attacks. Moreover, our algorithm is larger than the entropy value of the literature<sup>9,10,12</sup>, which shows that our algorithm is effective.

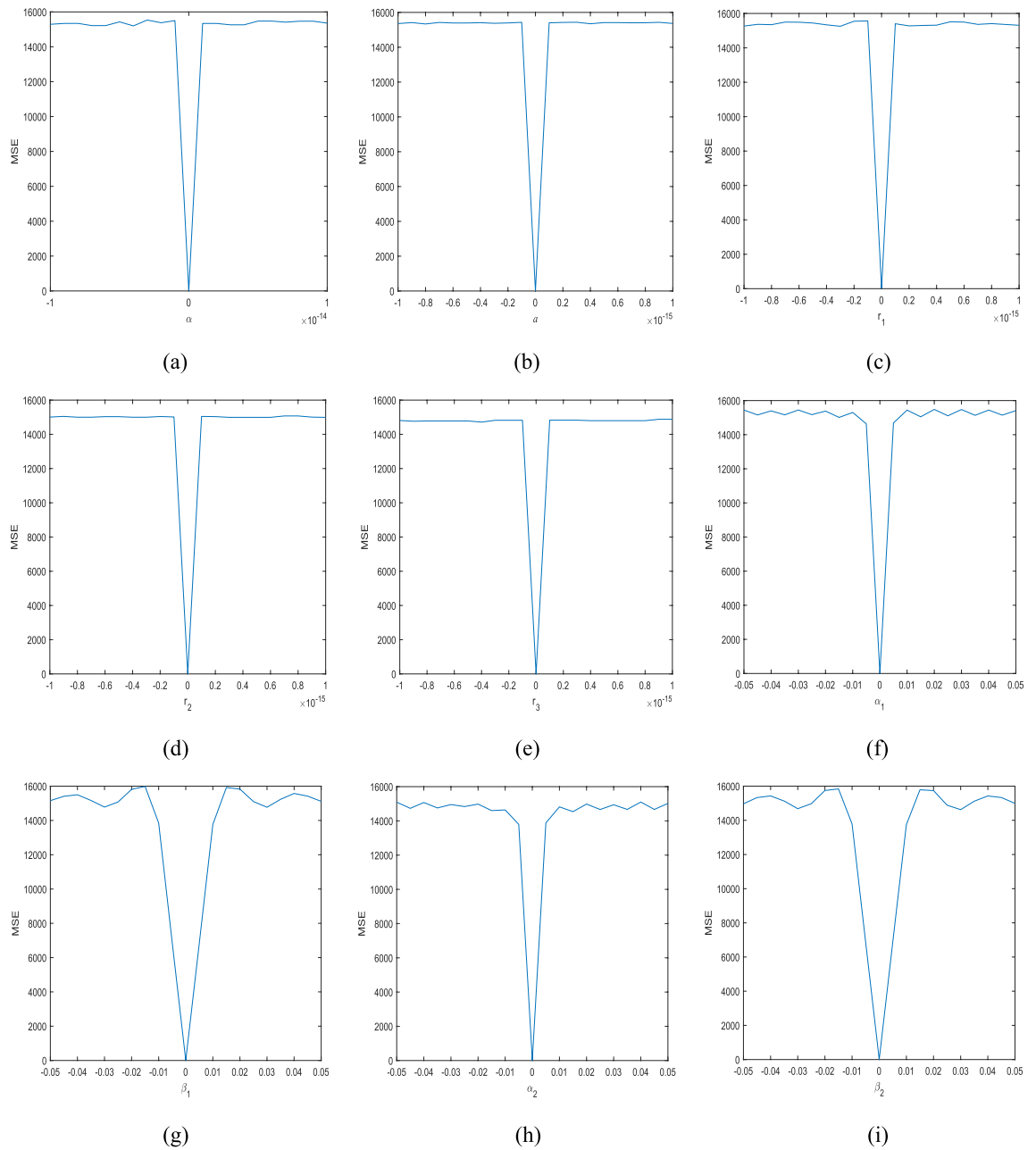
**Key space analysis.** When an attacker uses a violent attack, this requires enough key space to prevent the attacker from obtaining any information without the correct key<sup>36</sup>. In this algorithm, take lena picture as an example, the control parameters of the Chebyshev chaotic system are  $\alpha = 8$ , the initial value  $r_1 = 0.091, r_2 = 0.473, r_3 = 0.782$  is related to the plaintext control parameter  $a = 1.5041015625$ , the parameters of Arnold scrambling are  $b_1 = 8, b_2 = 7, t = 80$ , and the two-dimensional Discrete Fractional Fourier Transform (DFrFT) transform angle is  $\alpha_1 = 1.41, \alpha_2 = 0.6, \beta_1 = -0.41, \beta_2 = -1.6$ . Using Matlab for simulation experiments, the calculation accuracy is  $10^{-15}$ . The control parameters, initial conditions and Arnold scrambling parameter key space are all  $10^{15}$ , i.e.  $S_\alpha = S_{r_1} = S_{r_2} = S_{r_3} = S_a = S_{b_1} = S_{b_2} = S_t = 10^{15}$ ; the transformation angle key space of the two-dimensional DFrFT is  $S_{\alpha_1} = S_{\alpha_2} = S_{\beta_1} = S_{\beta_2} = 10^6$ ; By calculating, the system key space is  $S = S_\alpha \cdot S_{r_1} \cdot S_{r_2} \cdot S_{r_3} \cdot S_a \cdot S_{b_1} \cdot S_{b_2} \cdot S_t \cdot S_{\alpha_1} \cdot S_{\alpha_2} \cdot S_{\beta_1} \cdot S_{\beta_2} = 10^{144}$ . It is much larger than the key space of each algorithm in Table 6, so the algorithm can resist brute force attacks.

**Key sensitivity analysis.** The key sensitivity of the algorithm is very strong, any key small changes, other keys remain unchanged, under the correct encryption algorithm, cannot decrypt the correct plaintext image<sup>49</sup>. The simulation experiment results are shown in Fig. 6a–e respectively represent the decrypted image of  $\alpha = 8 + 10^{-15}, a = 1.5041015625 + 10^{-15}, r_1 = 0.091 + 10^{-15}, r_2 = 0.473 + 10^{-15}, r_3 = 0.782 + 10^{-15}$ . Figure 6f–h respectively represent the decrypted image of  $b_1 = 9, b_2 = 8, t = 79$ . Figure 6i–l respectively represent the decrypted image of  $\alpha_1 = 1.41 + 0.01, \alpha_2 = 0.6 + 0.01, \beta_1 = -0.41 + 0.01, \beta_2 = 1.6 + 0.01$ . Experiments have shown that small changes in the key have a great impact on decryption.

Figure 7a shows the mean square error (MSE) curve of the deviation of the control parameter  $\alpha$ . Try different values in the range  $[-10^{-14}, 10^{-14}]$ , step size is  $10^{-15}$ . Figure 7b–e shows the MSE curve of the deviation of initial condition  $a, r_1, r_2, r_3$ . Try different values in the range  $[-10^{-15}, 10^{-15}]$ , step size is  $10^{-16}$ . It can be seen that the key is slightly transformed, the MSE is large, and the original image cannot be seen in the decrypted image. Figure 7f–i show the MSE graph of the deviation of the FrFT order  $\alpha_1, \alpha_2, \beta_1, \beta_2$ , and it can be seen that the order is slightly changed, and the MSE is large. Therefore, this algorithm is very sensitive to keys.



**Figure 6.** Decrypted “Lena” with incorrect (a)  $\alpha$ , (b)  $a$ , (c)  $r_1$ , (d)  $r_2$ , (e)  $r_3$ , (f)  $b_1$ , (g)  $b_2$ , (h)  $t$ , (i)  $\alpha_1$ , (j)  $\alpha_2$ , (k)  $\beta_1$ , (l)  $\beta_2$ .



**Figure 7.** MSE curves for (a)  $\alpha$ , (b)  $a$ , (c)  $r_1$ , (d)  $r_2$ , (e)  $r_3$ , (f)  $\alpha_1$ , (g)  $\alpha_2$ , (h)  $\beta_1$ , (i)  $\beta_2$ .

**Noise attack.** Next, we test the resistance of this algorithm to noise. Take the Pepper as an example, add Gaussian noise (GN) with mean value of 0 and variance of  $10^{-5}$ ,  $10^{-4}$ ,  $10^{-3}$  to the encrypted image, and the decrypted images are shown in Fig. 8a–c. Add Salt and Pepper noise (SPN) with intensity  $10^{-4}$ ,  $10^{-3}$ ,  $10^{-2}$  to the encrypted image, and the decrypted images are shown in Fig. 8d–f. Add Speckle noise (SN) with intensity  $10^{-4}$ ,  $10^{-3}$ ,  $10^{-2}$  to the encrypted image, and the decrypted images are shown in Fig. 8g–i. Decrypting the noise-added image can see the rough information of the original image, so this algorithm has better robustness. Table 7 compares the PSNR value of the decrypted image and the Lena plaintext image when the encrypted image is attacked by GN, SPN, and SN when the compression rate is 50% with the algorithm<sup>25</sup>. It can be seen from the table that this algorithm has a stronger ability to resist noise attack. Add random noise of different intensities to the Lena ciphertext image, as shown in Eq. (29). The Table 8 is the PSNR value of the decrypted image and the plaintext image with random noise added with different intensities. It can be seen that the quality of the restored image by this algorithm is relatively high under the same intensity.

$$I = I + k \times \text{Noise} \tag{29}$$



**Figure 8.** The results of noise attack with different noise strengths: (a)  $10^{-5}$  GN, (b)  $10^{-4}$  GN, (c)  $10^{-3}$  GN, (d)  $10^{-4}$  SPN, (e)  $10^{-3}$  SPN, (f)  $10^{-2}$  SPN, and (g)  $10^{-4}$  SN, (h)  $10^{-3}$  SN, (i)  $10^{-2}$  SN.

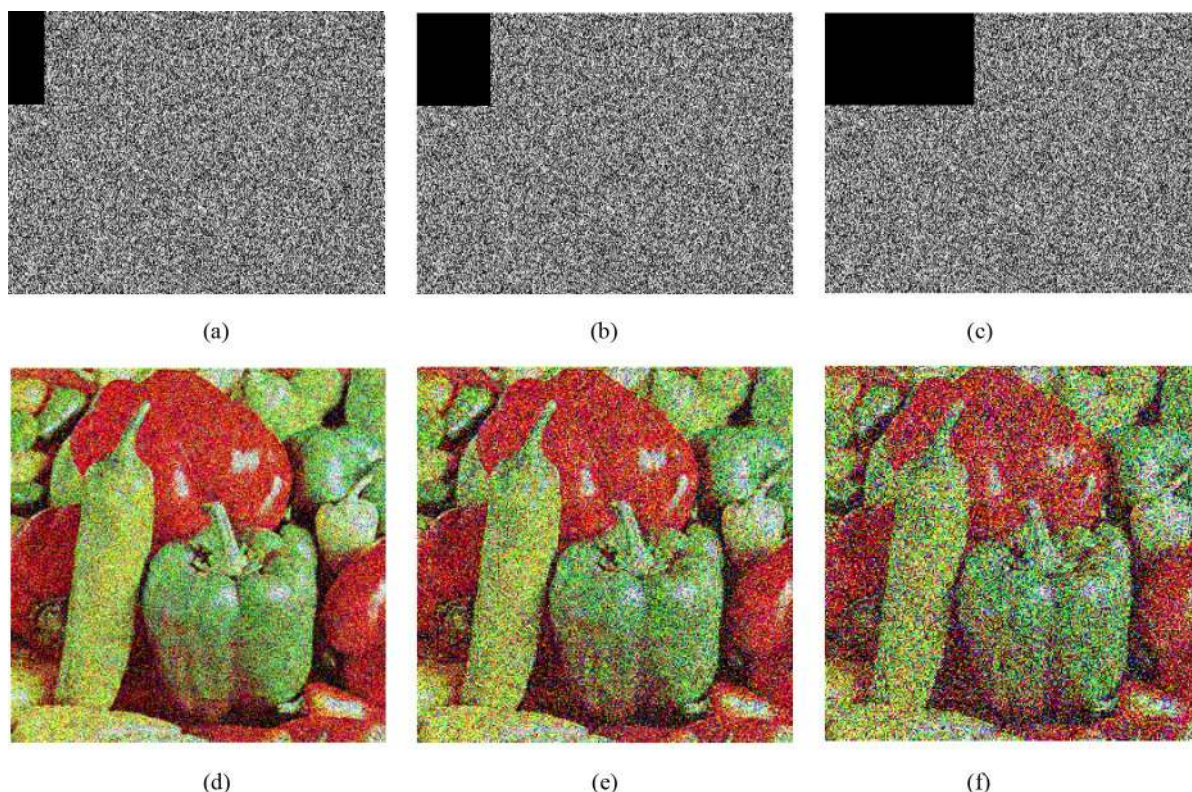
Image	GN			SPN			SN		
	$10^{-6}$	$3 \times 10^{-6}$	$5 \times 10^{-6}$	$10^{-6}$	$3 \times 10^{-6}$	$5 \times 10^{-6}$	$10^{-6}$	$3 \times 10^{-6}$	$5 \times 10^{-6}$
Proposed	34.8113	33.2051	32.4611	35.3472	35.3472	35.3472	35.3472	34.5733	33.9705
Ref. <sup>25</sup>	30	<28	<27	31	<31	<30	31	<32	<32

**Table 7.** The anti-noise performance comparison of two methods in the 50% sampling rate.

**Clipping attack.** When the ciphertext is subjected to a tailoring attack during transmission, there is no doubt that the quality of the decrypted image will decrease. Figure 9 shows three different clipping methods and their recovery results. Experiments show that although the decrypted image is a rough version of the original image, the main information of the original image can still be represented by the correct key. Experiments have shown that encryption algorithms can resist tailoring attacks. Table 9 is a comparison of the PSNR of Lena’s decrypted image and plaintext image with other algorithms. The image is restored after 5%, 10%, and 20%

Image	Random noise attack k=0.1	Random noise attack k=0.3	Random noise attack k=0.6
Proposed	41.3867	38.7516	35.1461
Ref. <sup>39</sup>	20.09	12.27	9.58

**Table 8.** The anti-noise performance comparison of two methods in the 75% sampling rate.



**Figure 9.** Robustness of the encryption scheme: (a) encrypted image with 5% data loss, (b) encrypted image with 10% data loss, (c) encrypted image with 20% data loss. (d)–(f) are corresponding decrypted images of (a)–(c), respectively.

Data loss intensity (%)	PSNR	Ref. <sup>39</sup>
5	13.25	12.51
10	10.21	10.67
20	7.19	8.72

**Table 9.** The comparison of performance against cropping of two algorithms in the 75% sampling rate.

loss of encrypted image data. It can be seen that the PSNR is lower as the data is lost more. Compared with the algorithm<sup>39</sup>, the PSNR value of this algorithm is dominant in the data loss of 5%, but with the increase of data loss, it is not dominant. Therefore, our algorithm can resist tailoring attacks to a certain extent.

**Differential attack.** To test whether an encryption scheme is good, NPCR (Number of Pixels Change Rate) and the UACI (Unified Average Changing Intensity) are important standards. If a slight change is made to the plaintext pixel value, which corresponds to a large change in the encrypted pixel value, it means that the encryption scheme is good. NPCR and UACI are the numerical response of this standard. The calculation method of NPCR and UACI is as follows<sup>11</sup>:

$$NPCR = \frac{\sum_{i,j} D(i,j)}{W \times H} \times 100\%, \tag{30}$$

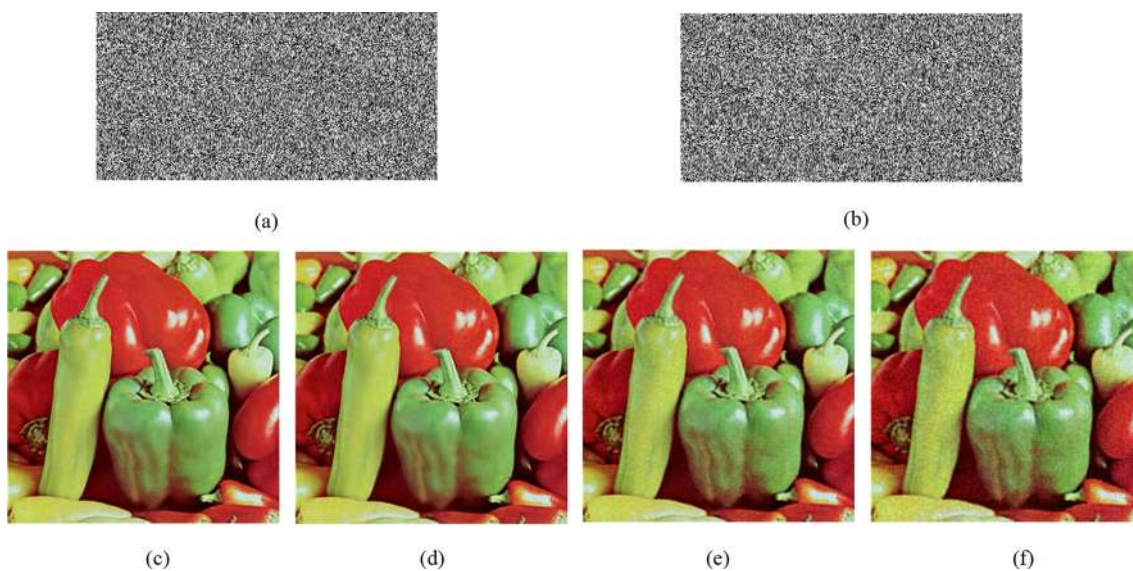
and

Images	NPCR (%)	UACI (%)
Lena	99.6078	33.4531
House	99.6323	33.4499
Baboon	99.6246	33.3373
Pepper	99.5818	33.3626
Airplane	99.6155	33.1816

**Table 10.** The mean NPCR and UACI of ciphered images.

Lena images	NPCR (%)	UACI (%)
Proposed algorithm	99.6078	33.4531
Ref. <sup>9</sup>	99.66	33.62
Ref. <sup>10</sup>	1	33.47
Ref. <sup>11</sup>	99.62	33.45
Ref. <sup>12</sup>	99.6090	33.4727
Ref. <sup>13</sup>	99.6155	33.2744
Ref. <sup>14</sup>	99.6017	28.1370

**Table 11.** Comparison of NPCR and UACI on 'Lena'



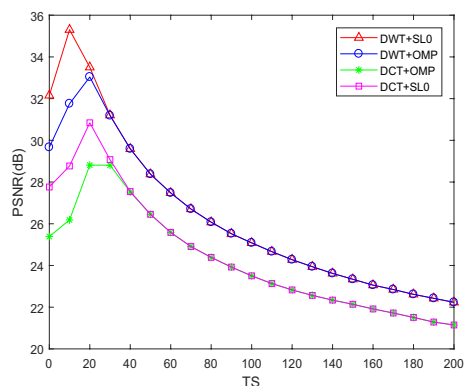
**Figure 10.** Simulation results of different sparse and reconstruction methods for Pepper.

$$UACI = \frac{1}{W \times H} \left[ \sum_{ij} \frac{|d_1(i,j) - d_2(i,j)|}{255} \right] \times 100\%. \quad (31)$$

Here  $M$  and  $N$  respectively represent the width and height of the image, and  $d_1$  and  $d_2$  are the two ciphertext images after the original plaintext image has been changed by one pixel value. If  $d_1(i,j) \neq d_2(i,j)$ , then  $D(i,j) = 1$ , otherwise,  $D(i,j) = 0$ . We add 1 to any pixel value, calculate 100 groups, and take the average to get Table 10. It can be seen from Table 10 that the NPCR obtained by the encryption scheme is about 99.60%, and the UACI is greater than 33%. Table 11 is the comparison result between this algorithm and other algorithms. We can find that although our results are not the best, they can resist differential attacks.

**The influence of different sparse and reconstruction methods on encryption and decryption results.** To analyze the impact of sparse methods and reconstruction methods, we use DWT and DCT sparse  $256 \times 256$  Pepper, and use OMP and SL to reconstruct image. As shown in Fig. 10, (a) is an encrypted image using DWT, (b) is an encrypted image using DCT, (c) is an image reconstructed using DWT sparse and SL0, and





**Figure 11.** PSNR vs TS for Pepper with different sparse and reconstruction methods.

Images size	Lena 256 × 256	Baboon 256 × 256	Pepper 512 × 512	Airplane 512 × 512
CR=0.25	1.67	1.65	5.13	5.15
CR=0.5	1.80	1.72	5.79	5.51
CR=0.75	1.79	1.81	6.35	6.04

**Table 12.** Encryption time (second).

Images size	Lena 256 × 256	Baboon 256 × 256	Pepper 512 × 512	Airplane 512 × 512
CR=0.25	4.59	3.94	10.37	10.97
CR=0.5	4.24	4.26	13.31	11.50
CR=0.75	4.70	4.47	14.58	13.78

**Table 13.** Decryption time (second).

Images size	Lena 256 × 256	Baboon 256 × 256	Pepper 256 × 256
Proposed	1.67	1.65	1.79
Ref. <sup>4</sup>	3.23	3.53	3.68
Ref. <sup>5</sup>	11.12	11.45	12.13
Ref. <sup>9</sup>	2.25	2.55	2.76

**Table 14.** The encryption time comparison results with other algorithms (second).

(d) is an image reconstructed using DWT sparse and OMP, (e) is an image reconstructed using DCT sparse and SL0, (f) is an image reconstructed using DCT sparse and OMP. It can be seen from the figure that using DWT sparse, the reconstructed visual quality is better. Figure 11 shows the relationship between the reconstruction effect and the threshold TS. It can be seen that when TS=10, using DWT sparse, the PSNR value of SL0 reconstruction is the largest.

**Time analysis.** In practical applications, both safety performance and time must be considered. As shown in Tables 12 and 13, this paper analyzes the encryption and decryption time of different sizes of images and different CRs. It can be seen from the table that for the same image, different CRs have a slight impact on the time. For 256 × 256 images, the encryption time range is 1.5–2, and for 512 × 512 images, the encryption time range is 5–6. For 256 × 256 images, the decryption time range is 3–5, for 512 × 512 images, the decryption time range is 10–15. The reason for the increase in the decryption time is that the reconstruction process takes a long time to find the optimal solution. When CR is equal, as the image size increases, the encryption and decryption process takes more time. Therefore, in practice, CR and time are comprehensively considered for selection. Table 14 compares the time with other algorithms. As shown in the table, our algorithm takes the shortest time.

## Conclusion

This paper combines the advantages of structured random perceptual matrix and chaos to obtain a structured sensing matrix measurement image. A compression-based and two-dimensional fractional Fourier image encryption is proposed. This paper first compresses and encrypts through CS, and then re-encrypts through 2D FrFT. The inverse scrambling matrix, the chaotic cyclic matrix, the sampling subset and the double random phase mask are generated by the Chebyshev chaotic sequence, that is, the chaotic system controls the encryption process. Simulation experiments show that the proposed algorithm has good resilience and robustness. It can not only resist statistical analysis, noise attack and tailoring attacks, but also has a large key space and is sensitive to keys. Therefore, the algorithm has good performance and security.

Received: 26 July 2020; Accepted: 16 October 2020

Published online: 29 October 2020

## References

- Liao, X., Yingbo, Yu., Li, B., Li, Z. & Qin, Z. A new payload partition strategy in color image steganography. *IEEE Trans. Circuits Syst. Video Technol.* **30**, 685–696 (2020).
- Liao, X., Yin, J., Chen, M. & Qin, Z. Adaptive payload distribution in multiple images steganography based on image texture features. *IEEE Trans. Dependable Secur. Comput.* <https://doi.org/10.1109/TDSC.2020.3004708> (2020).
- Xian, Y. J., Wang, X. Y., Yan, X. P., Li, Q. & Wang, X. Y. Image encryption based on chaotic sub-block scrambling and chaotic digit selection diffusion. *Opt. Lasers Eng.* **134**, 106202 (2020).
- Ahmed, F., Anees, A., Abbas, V. U. & Siyal, M. Y. A noisy channel tolerant image encryption scheme. *Wirel. Pers. Commun.* **77**, 2771–2791 (2014).
- Anees, A., Siddiqui, A. M. & Ahmed, F. Chaotic substitution for highly autocorrelated data in encryption algorithm. *Commun. Nonlinear Sci. Numer. Simul.* **19**, 3106–3118 (2014).
- Wang, X. Y. & Gao, S. Image encryption algorithm for synchronously updating Boolean networks based on matrix semi-tensor product theory. *Inf. Sci.* **507**, 16–36 (2020).
- Wang, X. Y., Feng, L. & Zhao, H. Y. Fast image encryption algorithm based on parallel computing system. *Inf. Sci.* **486**, 340–358 (2019).
- Chen, J., Chen, L., Zhang, L. Y. & Zhu, Z. Medical image cipher using hierarchical diffusion and non-sequential encryption. *Nonlinear Dyn.* **6**, 301–322 (2019).
- Ahmad, J. & Hwang, S. O. A secure image encryption scheme based on chaotic maps and affine transformation. *Multimed. Tools Appl.* **75**, 13951–13976 (2015).
- Wu, J., Liao, X. & Yang, B. Color image encryption based on chaotic systems and elliptic curve ElGamal scheme. *Signal Process.* **141**, 109–124 (2017).
- Chai, X., Gan, Z. & Zhang, M. A fast chaos-based image encryption scheme with a novel plain image-related swapping block permutation and block diffusion. *Multimed. Tools Appl.* **76**, 15561–15585 (2017).
- Wu, X., Wang, K., Wang, X., Kan, H. & Kurths, J. Color image DNA encryption using NCA map-based CML and one-time keys. *Signal Process.* **148**, 272–287 (2018).
- Guesmi, R., Farah, M. A. B., Kachouri, A. & Samet, M. A novel chaos-based image encryption using DNA sequence operation and Secure Hash Algorithm SHA-2. *Nonlinear Dyn.* **83**, 1123–1136 (2016).
- Liu, H. & Wang, X. Image encryption using DNA complementary rule and chaotic maps. *Appl. Soft. Comput.* **12**, 1457–1466 (2012).
- Wang, X. Y. & Gao, S. Image encryption algorithm based on the matrix semi-tensor product with a compound secret key produced by a Boolean network. *Inf. Sci.* **539**, 195–214 (2020).
- Hua, Z. & Zhou, Y. Image encryption using 2D Logistic-adjusted-Sine map. *Inf. Sci.* **339**, 237–253 (2016).
- Niyat, A. Y., Moattar, M. H. & Torshiz, M. N. Color image encryption based on hybrid hyper-chaotic system and cellular automata. *Opt. Lasers Eng.* **90**, 225–237 (2017).
- Donoho, D. L. Compressed sensing. *IEEE Trans. Inf. Theory* **52**, 1289–1306 (2006).
- Huang, R., Rhee, K. H. & Uchida, S. A parallel image encryption method based on compressive sensing. *Multimed. Tools Appl.* **72**, 71–93 (2014).
- Candes, E. & Tao, T. Decoding by linear programming. *IEEE Trans. Inf. Theory* **51**, 4203–4215 (2005).
- Tropp, J. A. & Gilbert, A. C. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Inf. Theory* **53**, 4655–4666 (2007).
- Xu, Q. Y., Sun, K. H., Cao, C. & Zhu, C. X. A fast image encryption algorithm based on compressive sensing and hyperchaotic map. *Opt. Laser Eng.* **121**, 203–214 (2019).
- Gong, L., Qiu, K., Deng, C. & Zhou, N. An image compression and encryption algorithm based on chaotic system and compressive sensing. *Opt. Laser Technol.* **115**, 257–267 (2019).
- Zhou, N., Jiang, H., Gong, L. & Xie, X. Double-image compression and encryption algorithm based on co-sparse representation and random pixel exchanging. *Opt. Laser Eng.* **110**, 72–79 (2018).
- Chai, X., Zheng, X., Gan, Z., Han, D. & Chen, Y. An image encryption algorithm based on chaotic system and compressive sensing. *Signal Process.* **148**, 124–144 (2018).
- Preishuber, M., Hutter, T., Katzenbeisser, S. & Uhl, A. Depreciating motivation and empirical security analysis of chaos-based image and video encryption. *IEEE Trans. Inf. Forensics Secur.* **13**, 2137–2150 (2018).
- Zhang, J., Han, G. & Fang, Y. Deterministic construction of compressed sensing matrices from protograph LDPC codes. *IEEE Signal Proc. Lett.* **22**, 1960–1964 (2015).
- Naidu, R. R., Jampana, P. & Sastry, C. S. Deterministic compressed sensing matrices: construction via euler squares and applications. *IEEE Trans. Signal Process.* **64**, 3566–3575 (2016).
- Xi, S. *et al.* Optical encryption method of multiple-image based on  $\theta$  modulation and computer generated hologram. *Opt. Commun.* **445**, 19–23 (2019).
- Sun, W., Wang, L., Wang, J., Li, H. & Wu, Q. Optical image encryption using gamma distribution phase masks in the gyrator domain. *J. Eur. Opt. Soc. Rapid* **14**, 28 (2018).
- Zhan, W., Zhang, L., Zeng, X., Chen, J. & Zhang, D. Study on an optical encryption algorithm based on compressive ghost imaging and super-resolution reconstruction. *Laser Phys.* **28**, 125202 (2018).
- Refregier, P. & Javidi, B. Optical image encryption based on input plane and Fourier plane random encoding. *Opt. Lett.* **20**, 767–769 (1995).
- Rawat, N., Kumar, R. & Lee, B. G. Implementing compressive fractional Fourier transformation with iterative kernel steering regression in double random phase encoding. *Optik* **125**, 5414–5417 (2014).
- Li, X. *et al.* Secret shared multiple-image encryption based on row scanning compressive ghost imaging and phase retrieval in the Fresnel domain. *Opt. Laser Eng.* **96**, 7–16 (2017).

35. Farah, M. A. B., Guesmi, R., Kachouri, A. & Samet, M. A novel chaos based optical image encryption using fractional Fourier transform and DNA sequence operation. *Opt. Laser Technol.* **121**, 105777 (2020).
36. Zhang, D., Liao, X., Yang, B., Yang, B. & Zhang, Y. A fast and efficient approach to color-image encryption based on compressive sensing and fractional Fourier transform. *Multimed. Tools Appl.* **77**, 2191–2208 (2018).
37. Zhou, N., Li, H., Wang, D., Pan, S. & Zhou, Z. Image compression and encryption scheme based on 2D compressive sensing and fractional Mellin transform. *Opt. Commun.* **343**, 10–21 (2015).
38. Liu, X. B., Mei, W. B. & Du, H. Q. Optical image encryption based on compressive sensing and chaos in the fractional Fourier domain. *J. Mod. Opt.* **61**, 3106–3118 (2014).
39. Liu, Q., Wang, Y., Wang, J. & Wang, Q. H. Optical image encryption using chaos-based compressed sensing and phase-shifting interference in fractional wavelet domain. *Opt. Rev.* **25**, 46–55 (2018).
40. Yi, J. & Tan, G. Optical compression and encryption system combining multiple measurement matrices with fractional Fourier transform. *Appl. Opt.* **54**, 10650–10658 (2015).
41. Candes, E. J. The restricted isometry property and its implications for compressed sensing. *CR Math.* **346**, 589–592 (2008).
42. Tu, G., Liao, X. & Xiang, T. Cryptanalysis of a color image encryption algorithm based on chaos. *Optik* **124**, 5411–5415 (2013).
43. Candes, E. J., Romberg, J. & Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **52**, 489–509 (2006).
44. Roopkumar, R. Quaternionic one-dimensional fractional Fourier transform. *Optik* **127**, 11657–11661 (2016).
45. Sahin, A., Ozaktas, H. M. & Mendlovic, D. Optical implementations of two-dimensional fractional Fourier transforms and linear canonical transforms with arbitrary parameters. *Appl. Opt.* **37**, 2130–2141 (1998).
46. Liu, H. & Wang, X. Color image encryption based on one-time keys and robust chaotic maps. *Comput. Math. Appl.* **59**, 3320–3327 (2010).
47. Kang, X. J. & Tao, R. Color image encryption using pixel scrambling operator and reality-preserving MPFRHT. *IEEE Trans. Circ. Syst. Video Technol.* **6**, 1919–1932 (2019).
48. Kang, X. J., Ming, A. & Tao, R. Reality-preserving multiple parameter discrete fractional angular transform and its application to color image encryption. *IEEE Trans. Circ. Syst. Video Technol.* **6**, 1595–1607 (2018).
49. Wang, X. Y., Guan, N. N., Zhao, H. Y., Wang, S. W. & Zhang, Y. Q. A new image encryption scheme based on coupling map lattices with mixed multi-chaos. *Sci. Rep.* **10**, 9784 (2020).

## Acknowledgements

This research is supported by the National Natural Science Foundation of China (No: 61672124), the Password Theory Project of the 13th Five Year Plan National Cryptography Development Fund (No: MMJJ20170203), Liaoning Province Science and Technology Innovation Leading Talents Program Project (No: XLYC1802013), Key R&D Projects of Liaoning Province (No: 2019020105-JH2/103), Jinan City ‘20 universities’ Funding Projects Introducing Innovation Team Program (No: 2019GXRC031).

## Author contributions

X.W. provides ideas, design solutions, and the division of labor throughout the project. Y.S. wrote the main manuscript and code.

## Competing interests


The authors declare no competing interests.

## Additional information

**Correspondence** and requests for materials should be addressed to X.W.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

**Publisher’s note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

 **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2020