International Journal of Bifurcation and Chaos, Vol. 30, No. 4 (2020) 2050060 (16 pages)

DOI: 10.1142/S0218127420500601

An Image Encryption Algorithm Based on Random Walk and Hyperchaotic Systems

Cong Xu*, Jingru Sun† and Chunhua Wang‡
College of Computer Science and Electronic Engineering,
Hunan University, Changsha 410082, P. R. China
*xucong0703@163.com
†jt_sunjr@hnu.edu.cn
‡wch1227164@hnu.edu.cn

Received April 26, 2019; Revised July 5, 2019

In this paper, we propose an image encryption algorithm based on random walk and two hyperchaotic systems. The random walk method is adopted to scramble the position of pixels within a block. Furthermore, the permutation operation between blocks is presented to enhance the scramble effect. Thus, high correlation among pixels of original image is broken by permutation. Moreover, the chosen plaintext attack is used to test the anti-attack ability of the proposed algorithm. By analyzing experimental results and comparing with other image encryption algorithms, we show that the proposed algorithm has better performance and higher security.

Keywords: Chaos; image encryption; random walk; hyperchaos; block image.

1. Introduction

Digital image is a popular multimedia format, and image encryption is a hot research topic in image processing and information security. Image has some inherent features, such as high correlation among pixels and huge size of data, thus traditional encryption techniques cannot meet the demand of image encryption [Li *et al.*, 2007]. Due to the characteristics of randomness, uncertainity, and initial conditions' sensitivity, chaotic system has good prospects in the field of image encryption.

Since the chaos-based encryption scheme was first introduced [Matthews, 1989], a great deal of chaos-based encryption methods have been put forward. There are two main recent design ideas in view of the encryption schemes based on chaotic system, one is to enhance the security of key stream by utilizing new types of chaotic systems or hyperchaotic systems, the other idea is to design more complex algorithms by new technologies, such as bit-level permutation, DNA computing, compression sensing, cellular automata.

For the first design idea, some researchers have proposed image encryption schemes based on new chaotic systems instead of existing chaotic systems/maps. By integrating two existing one-dimensional chaotic maps, a novel simple chaotic system was introduced for image encryption [Zhou *et al.*, 2014]. After comparing the performance of real chaotic system with that of complex chaotic system, it is found that the latter is more suitable for image encryption, thus a high security color encryption scheme based on two complex chaotic systems was proposed [Wang *et al.*, 2016]. Wang *et al.* [2017a] presented a new compounded piecewise–linear map for encryption by considering both safety and computing expense, the scheme has outstanding cryptography features and good efficiency. In addition, with the characteristics of better unpredictability, more complex structure and larger key

‡Author for correspondence

space [Zhang & Wang, 2019a; Zhou *et al.*, 2018a; Zhou *et al.*, 2018b; Wang *et al.*, 2017b; Zhao *et al.*, 2019; Lin & Wang, 2020; Zhou *et al.*, 2016a, 2017; Zhang & Wang, 2019b], hyperchaotic system has many applications in image encryption [Zhu, 2012; Norouzi *et al.*, 2014; Li *et al.*, 2018; Yin & Wang, 2018].

For the second design idea, compared with pixel-level image encryption schemes, bit-level encryption schemes can affect both the pixel value and the pixel position in the permutation process, so more and more scholars have introduced bit-level encryption schemes. Wang *et al.* [2010] proposed a bit-level encryption scheme by using a perceptron model. In [Xu *et al.*, 2016], the cryptography algorithm was constructed by using chaotic maps, making the bits swap to arbitrary bit planes. In [Zhou & Wang, 2020], to strengthen the security of the encryption scheme, bit-level permutation was applied to image encryption scheme. Recently, due to the high-parallel nature, and enormous information capacity, DNA computing has become a research hotspot, and DNA is widely used in designing a cryptographic algorithm. For example, Hu *et al.* [2017a] designed an algorithm by utilizing two chaotic systems and cycle operations of DNA sequence. Chai *et al.* [2017a] presented a chaos-based encryption algorithm by using DNA sequence operations. A novel scheme which combined the spatiotemporal chaos and DNA method was proposed by Zhang *et al.* [2016]. At present, in order to realize image compression and encryption simultaneously, many researchers have aimed to design new encryption systems by utilizing compressive sensing (CS). For example, Zhou *et al.* [2016b] introduced a simultaneous encryption-compression scheme, which realized both image compression and image encryption by combining CS and hyperchaotic system. A secure encryption scheme was proposed by Chai *et al.* [2017b], the scheme achieved the security of image data and image appearance by using CS. Besides, cellular automata (CA) has the characteristic of complex behavior from simple operation, and some image encryption algorithms based on CA have been developed in recent years. For example, novel image encryption schemes [Enayatifar *et al.*, 2015; Chai *et al.*, 2017c; Chai *et al.*, 2018] were proposed by employing DNA coding, chaotic system and cellular automata. In those schemes, DNA coding rules, DNA sequence operations and CA rules were utilized simultaneously to encrypt the input image.

Generally, for the second design idea, namely, designing more complex encryption algorithm, some algorithms are based on mathematical transformations, and some others are based on the rules designed by authors. Those transformations and rules are deterministic, which lack randomness and unpredictability. To solve the problem above, we propose a new encryption scheme based on random walk and hyperchaos. The plaintext is divided into blocks of fixed size first, and the blocks do not overlap. Then all pixels within each block image are scrambled by random walk matrix which records the path of the random walk. In the generation process of random walk matrix, the starting position and direction of movement are controlled by the random numbers which are obtained by the chaotic system, moreover, the number of movement times is unpredictable, which is affected by the starting position and direction of movement. Furthermore, permutation method between blocks is applied to scramble the positions of pixels effectively. The correlation of neighborhood pixels within blocks and that of adjacent pixels between blocks can be both broken by the permutation methods in our paper. We also present the diffusion method to change the value of pixels. To strengthen the security of the scheme, the same chaotic sequence is not reused in the permutation and diffusion process. In addition, if we use the fixed secret keys when encrypting different images, the attackers will have opportunities to obtain the secret keys by using plaintext attacks, and then use the secret keys to decrypt other cipher images [Wang *et al.*, 2018a; Chen *et al.*, 2018; Chen *et al.*, 2017; Hu *et al.*, 2017b]. So, the initial keys of chaotic systems are related to the plain image in the proposed scheme, the effect of "one plain image, one key" is attained to improve the security.

The remainder of this paper is organized as follows. In the next section, the basic theories applied in the scheme are reviewed. Section 3 shows the detailed process of encryption and decryption. Sections 4 and 5 present the experimental results and security analysis, respectively. Section 6 shows the comparison results. In the last section, the conclusions are given.

2. Basic Theories

2.1. *Random walk matrix*

The term "random walk" was introduced by Pearson [1905]. Random walks are popular in the

sciences, and they are interesting from both theoretical and practical perspectives. There is a mathematical problem which is called a random walk problem, it can be described as: there is a man in a room, and the room size is $m \times n$, he stands in the designated starting position, then he random moves to one of the eight directions around him continuously. The process of random walk is not completed until all the positions of the room have been walked.

We simulate the process of random walk in a blank matrix, then the random walk path can be recorded in the matrix which we call random walk matrix (RWM), according to the rules as follows: each position records the number of times he walked in. It is clear that the process of random walk involves three parameters, they are the size of the matrix, the starting position and the direction of each movement. In this paper, we set the size of the matrix to be the same as the sub-block image, the starting position and direction of each movement are given by random numbers which are generated by the chaotic system.

2.2. *Chaotic systems*

Chaotic systems are adopted to obtain key sequences in our scheme, the hyperchaotic Lorenz system [Wang & Wang, 2008] and Chen's hyperchaotic system [Gao *et al.*, 2006] are represented by Eqs. (1) and (2), respectively,

$$\begin{cases} \dot{x} = \alpha(y - z) + w, \\ \dot{y} = \gamma x - y - xz, \\ \dot{z} = xy - \beta z, \\ \dot{w} = -yz + \eta w, \end{cases} \tag{1}$$

the system displays chaotic behavior with fixed or changed parameters ($\alpha = 10$, $\beta = 8/3$, $\gamma = 28$ and $-1.52 < \eta \leq -0.06$), and we choose $\eta = -1$ in our experiment

$$\begin{cases} \dot{x} = a(y - x), \\ \dot{y} = -xy + nx + my - w, \\ \dot{z} = xy - bz, \\ \dot{w} = x + k, \end{cases} \tag{2}$$

with $a = 36$, $b = 3$, $m = 28$, $n = 16$ and $k \in [-0.7, 0.7]$, the system has rich dynamics of hyperchaos, and k is set as 0.2 in our experiments.

3. The Process of Image Encryption and Decryption

We assume the size of the original image P is $M \times N$, and that of the random walk matrix is $m \times n$. Figure 1 gives the flow diagram of our proposed encryption scheme.

3.1. *The generation of initial keys*

It is obvious that hash values of two images with only 1-bit difference have obvious distinctions, and

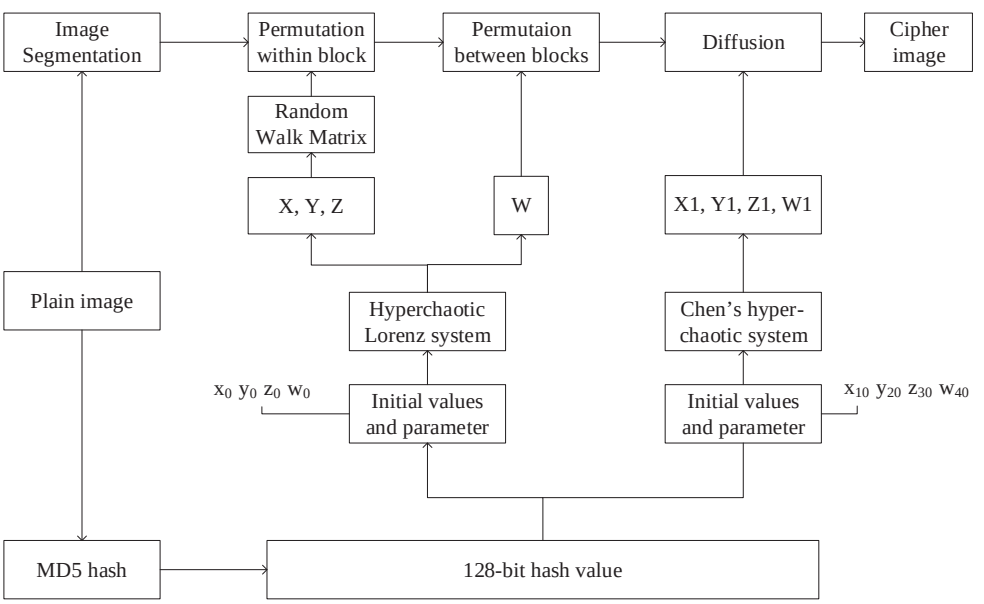


Fig. 1. The flow diagram of the scheme.

the initial keys related to MD5 hash values of plain image will enhance the sensitivity of the scheme. The 128-bit hash value K is divided into 16 blocks, and each block includes 8-bits, denoted as follows:

$$K = k_1, k_2, k_3, \ldots, k_{16}. \tag{3}$$

According to the following computing method, four initial keys of hyperchaotic Lorenz system can be obtained.

$$\begin{cases} x'_0 = x_0 + abs\left(\dfrac{k_1 + k_2 + k_3 + k_4}{256}\right) \\ \qquad - fix\left(\dfrac{k_1 + k_2 + k_3 + k_4}{256}\right) \\ y'_0 = y_0 + abs\left(\dfrac{k_5 + k_6 + k_7 + k_8}{256}\right) \\ \qquad - fix\left(\dfrac{k_5 + k_6 + k_7 + k_8}{256}\right) \\ z'_0 = z_0 + abs\left(\dfrac{k_9 + k_{10} + k_{11} + k_{12}}{256}\right) \\ \qquad - fix\left(\dfrac{k_9 + k_{10} + k_{11} + k_{12}}{256}\right) \\ w'_0 = w_0 + abs\left(\dfrac{k_{13} + k_{14} + k_{15} + k_{16}}{256}\right) \\ \qquad - fix\left(\dfrac{k_{13} + k_{14} + k_{15} + k_{16}}{256}\right) \end{cases} \tag{4}$$

where x_0, y_0, z_0, w_0 are the initial given values, $abs(a)$ to obtain the absolute value of a, $fix(a)$ takes the integer part of a. For another chaotic system as shown in Eq. (2), its initial keys can be computed by Eq. (5).

$$\begin{cases} x'_{10} = x_{10} + \dfrac{k_1 \oplus k_2 \oplus k_3 \oplus k_4}{256} \\ y'_{10} = y_{10} + \dfrac{k_5 \oplus k_6 \oplus k_7 \oplus k_8}{256} \\ z'_{10} = z_{10} + \dfrac{k_9 \oplus k_{10} \oplus k_{11} \oplus k_{12}}{256} \\ w'_{10} = w_{10} + \dfrac{k_{13} \oplus k_{14} \oplus k_{15} \oplus k_{16}}{256} \end{cases} \tag{5}$$

where $x_{10}, y_{10}, z_{10}, w_{10}$ are the initial given values, $a \oplus b$ indicates exclusive OR operation.

3.2. *The generation of random walk matrix*

In the paper, we simulate the random walk in a blank matrix controlled by the chaotic system, then generate random walk matrices (RWM) for permutation.

Step 1. Use the initial keys x'_0, y'_0, z'_0, w'_0 produced in Sec. 3.1 to produce four pseudo-random sequences x, y, z and w by iterating the hyperchaotic Lorenz system $N_0 + 8L$ times, where $L = M \times N$. To eliminate transient effect, the former N_0 values of each sequence are discarded.

$$N_0 = \mathrm{mod}(H, 1500), \tag{6}$$

where H is the decimal form of the MD5 hash value, $\mathrm{mod}(a, b)$ denotes modulo operation.

Step 2. The starting position (p_x, p_y) of random walk is computed by Eq. (7).

$$\begin{cases} p_x(i) = \mathrm{mod}(\mathrm{floor}(y(i) \times 10^{13}), m) + 1, \\ p_y(i) = \mathrm{mod}(\mathrm{floor}(z(i) \times 10^{13}), n) + 1, \end{cases} \tag{7}$$

where $\mathrm{floor}(a)$ finds the elements of a to the largest integer that is smaller than a, and $i = 1, 2, 3, \ldots, L/(m \times n)$.

Step 3. The direction of random walk is controlled by X, and X is computed by Eq. (8).

$$X(i) = \mathrm{mod}(\mathrm{floor}(x(i) \times 10^{13}), 8). \tag{8}$$

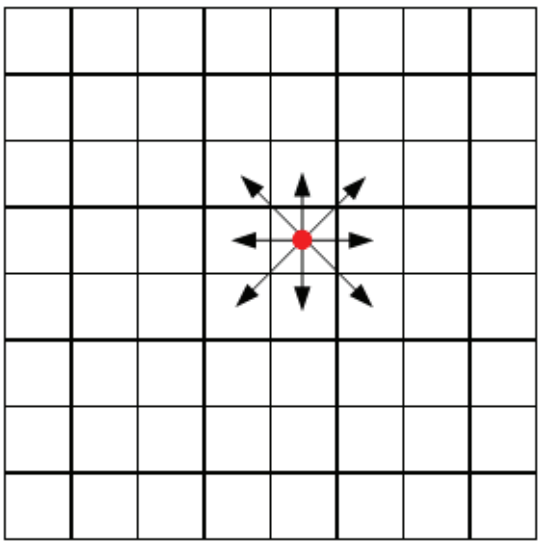
Step 4. Set $i = 1$.

Step 5. Perform random walk on a $m \times n$ blank matrix A_i as discussed in Sec. 2.1, the starting location is $(p_x(i), p_y(i))$ and the direction of each movement is controlled by $X(d)$, where $d = 1, 2, 3, \ldots, t$, and t is the number of movement times. Then the random walk matrix A_i is obtained. For example, the starting position in a blank matrix is $(4, 5)$ as shown in Fig. 2(a), and when the random walk is completed, we obtained the RWM A_i as shown in Fig. 2(b), each position of A_i records the number of times he walked in this position.

Step 6. Perform Step 5 repeatedly by setting $i = i + 1$. Then random walk matrices M_i $(i = 1, 2, 3, \ldots, L/(m \times n))$ are obtained.

3.3. *Permutation*

Considering the increasing size of the image, we manipulate scramble operation by block. Our



(a)

1	2	6	7	10	10	1	0
2	6	6	18	15	14	10	3
5	8	10	8	10	16	6	2
3	10	9	7	13	10	2	5
2	11	13	12	8	12	10	4
3	10	13	12	9	10	5	3
6	4	9	11	8	3	3	3
0	4	7	8	5	1	2	1

(b)

Fig. 2. An instance of random walk: (a) An 8×8 blank matrix, and the starting location is $(4, 5)$ and (b) random walk matrix is obtained.

permutation process contains the permutation within the block and the permutation between the blocks.

Firstly, the $(M \times N)/(m \times n)$ sub-blocks are obtained by segmenting images into the block, we set B_i indicating the ith block, here, numbering blocks according to top-to-bottom and left-to-right, and the size of each sub-block is $m \times n$. In general, M should be an integral multiple of m, N should be an integral multiple of n, if this is not satisfied, the image is padded with black color, so the algorithm has no limit for image size.

3.3.1. *Permutation within blocks based on random walk matrix*

In this phase, we scrambled positions of pixels within each sub-block image based on RWM. The permutation operation within blocks is described by a pseudo-code in Table 1. For instance, Fig. 3(a) is the 8×8 image block. In Fig. 3(b), the permuted image is obtained after permuting based on random walk matrix A which is shown in Fig. 2(b).

3.3.2. *Permutation between blocks*

After the permutation based on RWM is employed to each sub-block, the permutation method between block images is employed, and pixels of block image B'_i $(i = 1, 2, 3, \ldots, (M \times N)/(m \times n))$ will be swapped with pixels of other block images located after it. The permutation operation between blocks is described by pseudo-code in Table 2.

Table 1. Pseudo-code of the permutation within blocks.

Input: Random walk matrix A, sub-block B
Output: Permuted sub-blocks B'
1. Convert A, B into one-dimension sequence
2. for $i = 1 : (M \times N)/(m \times n)$
3. index sequence $l \leftarrow A_i$ is sorted in ascending order
4. for $j = 1 : (m \times n)$
5. $B'_i \leftarrow B'_i(j) = B_i(l(j))$
6. end
7. end

3.4. *Diffusion*

We utilized a key stream strongly related to plain image in the diffusion process for a good diffusion effect. Iterating Chen's hyperchaotic system $N_0 + MN$ times, we obtained the sequence x_1, y_1, z_1 and w_1, where the initial keys of the system are as in Sec. 3.1. The former N_0 values of each sequence are discarded to eliminate the transient effect. Then we can obtain the key stream X_1, Z_1 from Eqs. (9) and (10), and the diffusion operation is described by pseudo-code in Table 3.

$$X_1(i) = \mathrm{mod}(\mathrm{floor}((x_1(i) + y_1(i)) \times 10^{14}), 256), \tag{9}$$

$$Z_1(i) = \mathrm{mod}(\mathrm{floor}((z_1(i) + w_1(i)) \times 10^{14}), 256). \tag{10}$$

(a)

1	9	17	25	33	41	49	57
2	10	18	26	34	42	50	58
3	11	19	27	35	43	51	59
4	12	20	28	36	44	52	60
5	13	21	29	37	45	53	61
6	14	22	30	38	46	54	62
7	15	23	31	39	47	55	63
8	16	24	32	40	48	56	64

(b)

8	9	58	54	25	23	44	45
57	52	62	60	28	38	46	21
1	56	63	7	11	12	50	22
48	59	15	10	27	14	53	36
49	4	16	17	32	19	13	42
64	6	61	18	37	33	31	34
2	47	3	51	39	35	29	43
5	55	40	24	20	41	30	26

Fig. 3. Sub-block image permutation based on RWM: (a) An 8×8 sub-block image and (b) permuted image of (a) based on random walk matrix.

Table 2. Pseudo-code of the permutation between blocks.

Input: Permuted sub-blocks B', chaotic sequence w
Output: Permuted image D
1. $W \leftarrow \text{floor}(w \times 10^{13})$
2. for $i = 1 : (M \times N)/(m \times n)$
3. for $j = 1 : (m \times n)$
4. $k \leftarrow (W \bmod((M \times N)/(m \times n) - i)) + 1$
5. $i' \leftarrow i + k$
6. Exchange $B'_{i'}(j)$ and $B'_i(j)$
7. end
8. end
9. $D \leftarrow$ reconstruct sub-blocks into image

Table 3. Pseudo-code of the diffusion phase.

Input: Key streams X_1, Z_1, permuted image D
Output: Ciphered image C
1. Convert D into one-dimension sequence
2. for $i = 1 : (M \times N)$
3. $\text{Aver} \leftarrow \text{sum}(D)/(M \times N)$
4. $C(1) \leftarrow ((X_1(1) + Z_1(1)) \bmod 256) \oplus ((D(1) + \text{Aver}) \bmod 256)$
5. $C(i) \leftarrow ((X_1(i) + Z_1(i)) \bmod 256) \oplus ((D(i) + C(i-1)) \bmod 256)$
6. end

3.5. *The process of decryption*

Decryption is the converse of encryption.

4. Results of Simulation

Plain images for testing are Lena (512×512), Butterfly (512×768), Terrace (1200×1280) and Bridge (color 256×256). In the encryption, the color image is divided into R, G and B components, then the three components are encrypted separately. The secret keys are ($x_0 = 3.3133$, $y_0 = 12.0546$, $z_0 = 40.8897$, $w_0 = -34.5677$, $x_{10} = 0.3838$, $y_{10} = 0.9876$, $z_{10} = 32.1234$, $w_{10} = 0.6565$).

Figure 4 exhibits the results of encryption and decryption by using the proposed scheme. Obviously, all plain images and decrypted images are identical. Additionally, images of various sizes can be effectively encrypted by the proposed algorithm, that is, the encryption scheme has no limit for image size.

5. Security Analysis

In all of these simulations, testing images are the 512×512 pixels with 8-bit gray level. The security analysis is shown as below.

5.1. *Analysis of key space*

The secret keys of our scheme contain the given initial keys and hash values of plain image. The initial values are double-precision, whose key space is $(10^{16})^8 = 10^{128} \approx 2^{384}$. The key space of 128-bit hash value is 2^{64}. The total key space is $2^{384} \times 2^{64} = 2^{448}$. For a secure scheme, the key space should be larger than 2^{100} [Alvarez & Li, 2006]. Clearly, it is not feasible for attackers to break the proposed scheme by using brute-force attacks.

5.2. *Analysis of key sensitivity*

The original keys are set as ($x_0 = 3.3133$, $y_0 = 12.0546$, $z_0 = 40.8897$, $w_0 = -34.5677$, $x_{10} = 0.3838$,

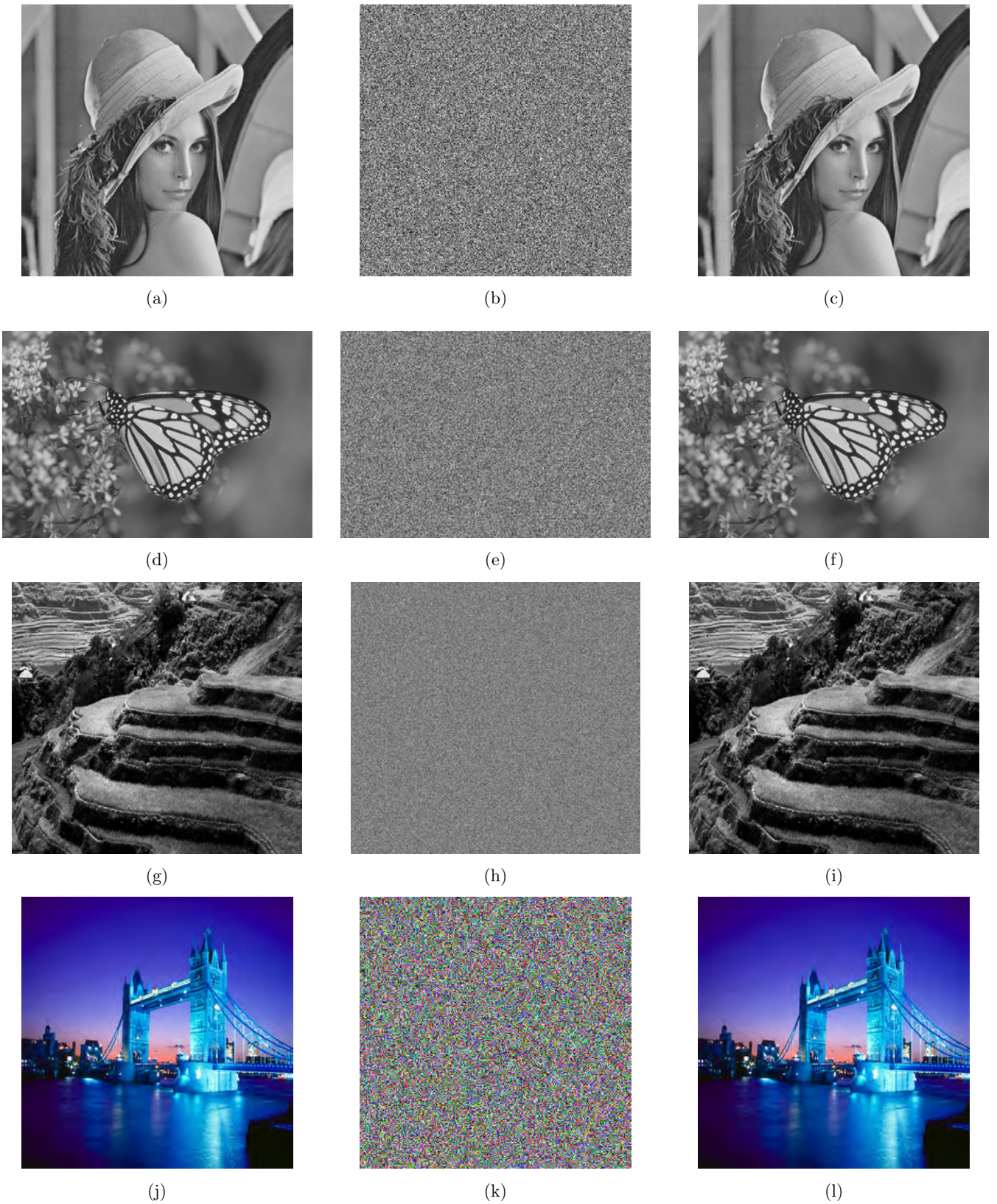
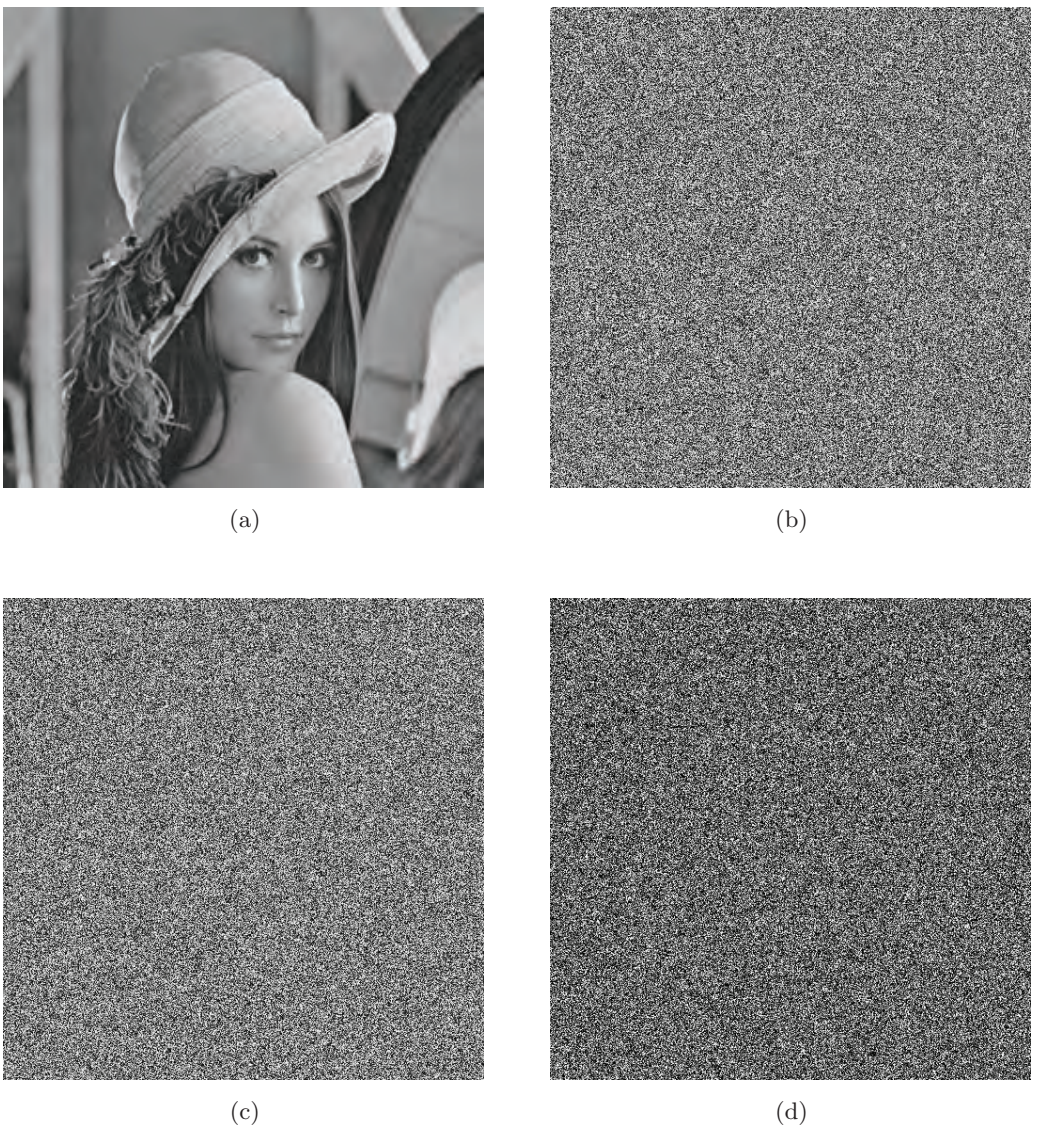


Fig. 4. Results of encryption and decryption: From the first to the third column are plain images, cipher images and decrypted images.



(a) (b)

(c) (d)

Fig. 5. Test of key sensitivity for encryption: (a) Plain image, (b) ciphertext with original keys, (c) ciphertext with modified keys and (d) difference between the two ciphertexts.

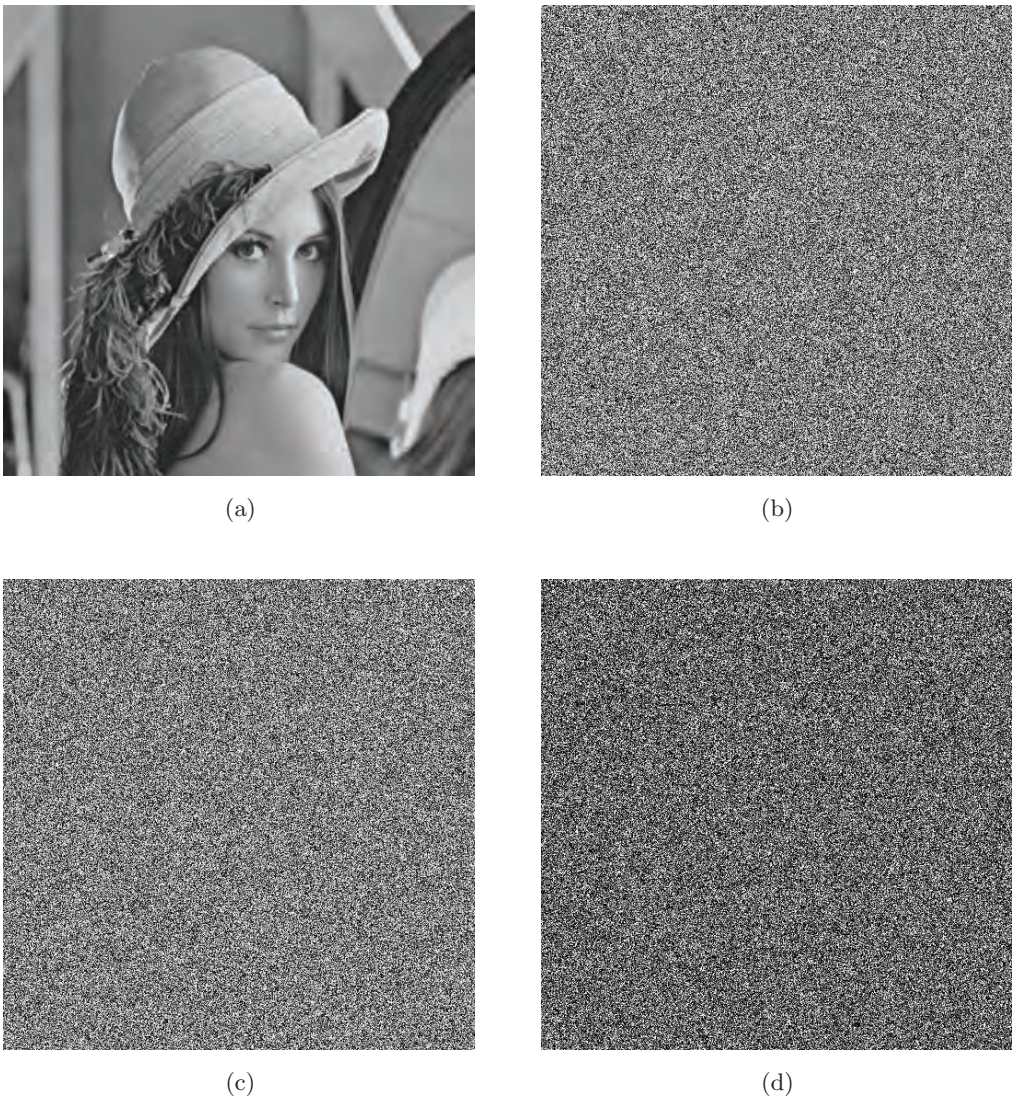
$y_{10} = 0.9876$, $z_{10} = 32.1234$, $w_{10} = 0.6565$), the modified keys are set as ($x_0 = 3.3133$, $y_0 = 12.0546$, $z_0 = 40.8897$, $w_0 = -34.5677$, $x_{10} = 0.3838 + 10^{-14}$, $y_{10} = 0.9876$, $z_{10} = 32.1234$, $w_{10} = 0.6565$).

In the process of encryption, the Lena image is encrypted by the original keys and modified keys, respectively. As Fig. 5 shows, ciphertext with the original keys is shown in Fig. 5(b), then we use the modified keys to encrypt the plain image again and obtain another ciphertext. By observing Fig. 5(d), it is clear that a slight modification in keys makes a tremendous difference in ciphertext. In the process of decryption, the ciphertext of Lena is decrypted by the original keys and modified keys, respectively. As Fig. 6 shows, the original keys restored the plain image successfully, but the plain image cannot be recovered correctly by using the slightly modified keys.

5.3. *Analysis of statistical attacks*

5.3.1. *Analysis of histogram*

Histogram is used to show the number of grayscale values with certain values. The cipher image with a uniform histogram can be obtained by a high security image encryption algorithm. The histograms of Lena image are shown in Fig. 7. In contrast with the histogram of plain image, the ciphertext has uniform distribution of pixel, which means that our scheme could resist the statistical attack.



(a) (b)

(c) (d)

Fig. 6. Test of key sensitivity for decryption: (a) Plain image, (b) ciphertext with original keys, (c) recovered image with modified keys and (d) recovered image with original keys.

5.3.2. *Analysis of correlation coefficient*

An image correlation coefficient means a statistical relationship between neighborhood pixels. High correlation between neighborhood pixels of plain image should be broken by an ideal encryption scheme. From the plain image and the corresponding cipher image, we randomly choose a number of pixels to calculate the correlation coefficients in three different directions by Eqs. (11)–(14).

$$\gamma_{xy} = \frac{\operatorname{cov}(x, y)}{\sqrt{D(x)D(y)}}, \tag{11}$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i, \tag{12}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x_i))^2, \tag{13}$$

$$\operatorname{cov}(x, y) = E(x_i - E(x_i))(y_i - E(y_i)). \tag{14}$$

The gray values of neighborhood pixels are x and y, respectively, N is the total number of pixels chosen. The correlation distribution in three directions are displayed in Fig. 8. By comparing the correlation coefficients between neighborhood pixels of plain image and that of cipher image (as listed in Table 4), we could get a conclusion that cipher images have a low correlation between neighborhood pixels, thus, our proposed scheme has the good ability to defend the statistical analysis attacks.

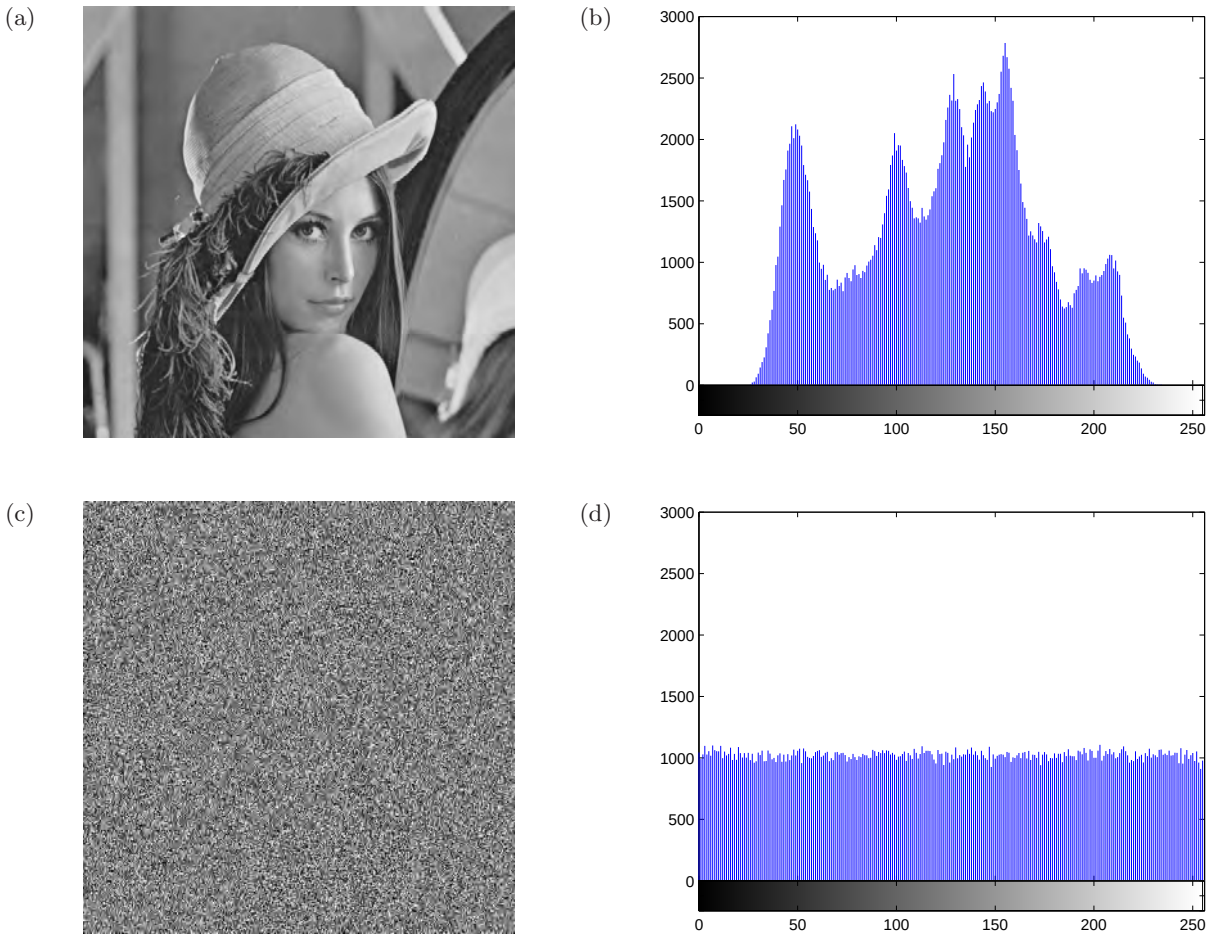


Fig. 7. Histogram analysis: (a) plaintext, (b) histogram of plaintext, (c) ciphertext and (d) histogram of ciphertext.

5.3.3. *Analysis of information entropy*

In information theory, the information entropy is used to measure information quantity, and the entropy can be calculated by Eq. (15)

$$H(m) = \sum_{i=0}^{2^N-1} p(x_i) \log_2 \frac{1}{p(x_i)}, \tag{15}$$

where the probability of symbol x_i's presence is denoted as $P(x_i)$, and the number of bits per pixel is N. Table 5 lists the entropy values of test images, and the calculation results are approaching the ideal theoretical value 8, this indicates the proposed algorithm can realize randomness for cipher images by encrypting plain images.

5.4. *Analysis of differential attacks*

Differential attackers break cryptographic algorithms by comparing and analyzing the corresponding cipher images of two plain images with a tiny difference. NPCR and UACI [Mao *et al.*, 2004] are two criteria to measure the capability of resistance to differential attacks.

$$\text{NPCR} = \frac{1}{M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} D(x,y) \times 100\%, \tag{16}$$

$$\text{UACI} = \frac{1}{M \times N} \left[\sum_{x=1}^{M} \sum_{y=1}^{N} \frac{|C_1(i,j) - C_2(i,j)|}{255} \right] \times 100\%, \tag{17}$$

$$D(x,y) = \begin{cases} 0, & C_1(x,y) = C_2(x,y), \\ 1, & C_1(x,y) \neq C_2(x,y), \end{cases} \tag{18}$$

where pixel values of cipher images are denoted as $C_1(x,y)$ and $C_2(x,y)$, and cipher images are corresponding to the two original images with only 1-bit difference.

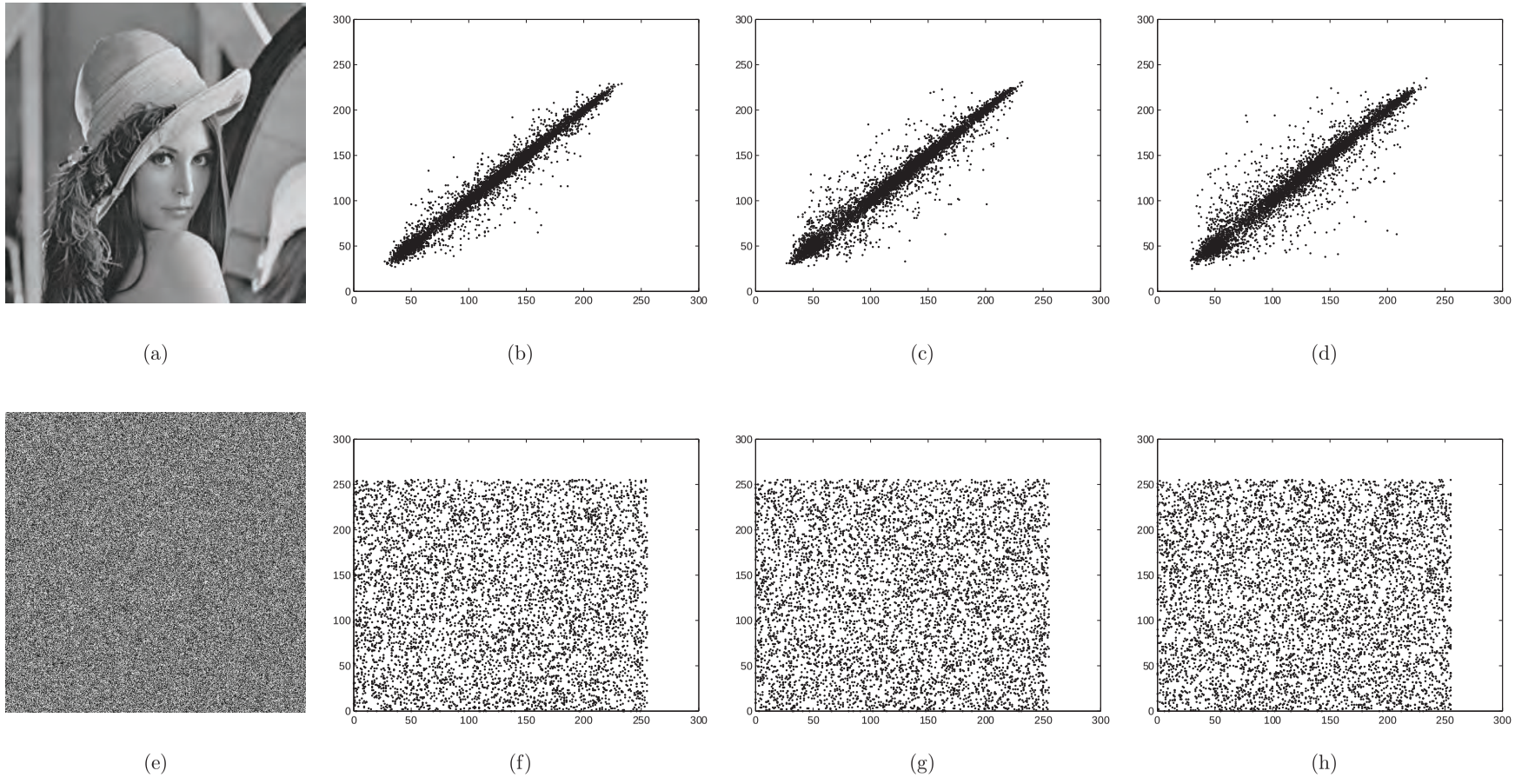


Fig. 8. The correlation distribution of (a) Lena and (e) ciphered image. (b)–(d) and (f)–(h) Correlation distribution in horizontal, vertical and diagonal directions, respectively.

Table 4. Correlation coefficients.

	Horizontal		Vertical		Diagonal	
Image	Plain	Cipher	Plain	Cipher	Plain	Cipher
Lena	0.9852	−0.0122	0.9720	0.0013	0.9583	−0.0043
Boat	0.9669	0.0111	0.9704	−0.0237	0.9413	0.0042
Baboon	0.7645	0.0237	0.8632	0.0046	0.7345	−0.0023
Girl	0.9873	−0.0062	0.9842	−0.0079	0.9749	−0.0019
Pepper	0.9815	−0.0110	0.9782	−0.0073	0.9637	−0.0116

Table 5. The entropies of cipher images.

Image	Lena	Boat	Baboon	Girl	Pepper
Entropy	7.9993	7.9992	7.9993	7.9994	7.9993

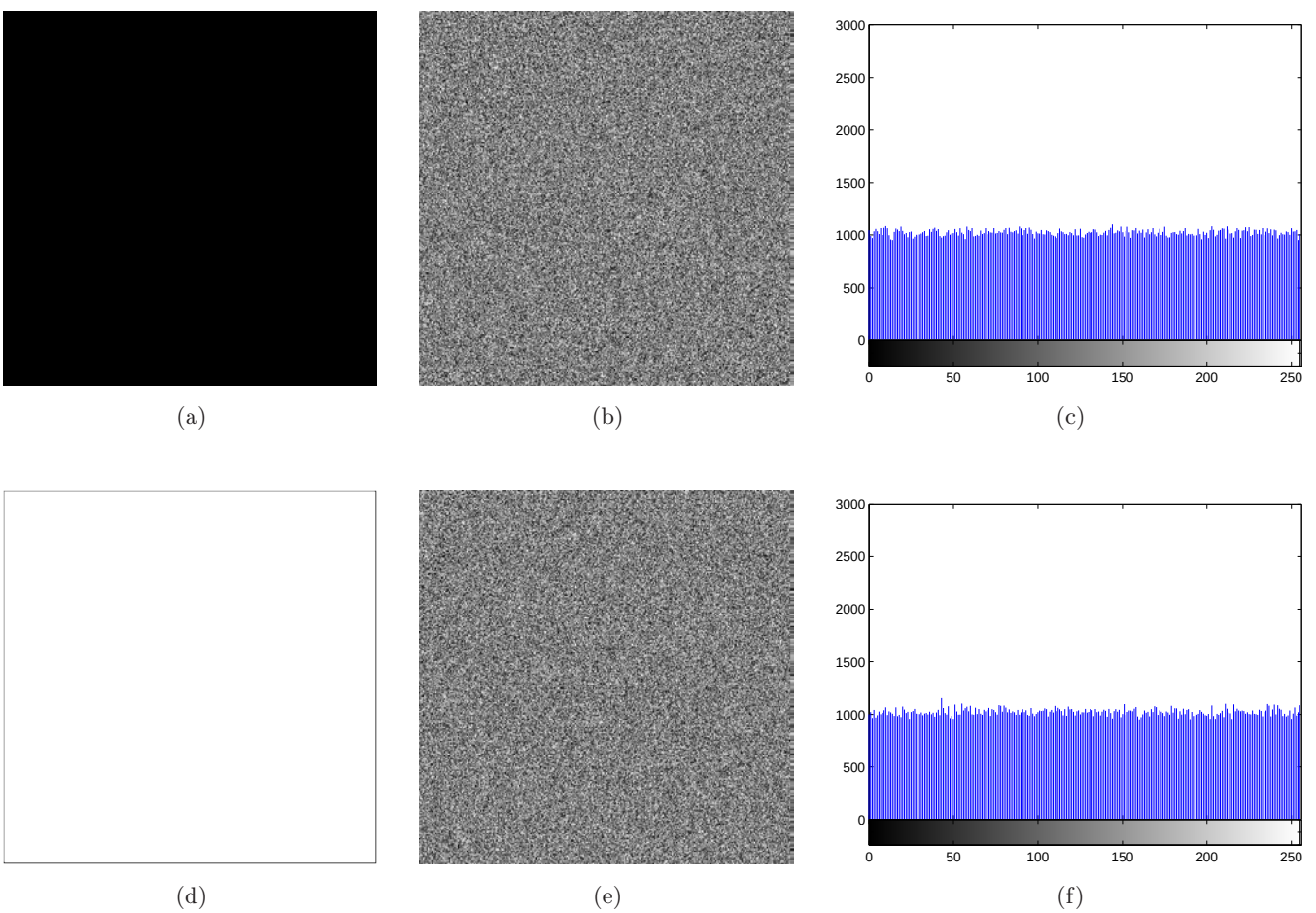
The results in Table 6 show that the values of NPCR and UCAI are approaching the ideal theoretical values 99.61% and 33.46% only through one round of encryption. In other words, our scheme can withstand the differential attack effectively.

Table 6. Performance of NPCR and UACI.

Algorithms	Encryption Round	NPCR	UACI
Ours	1	0.9961	0.3347
Li's [Li *et al.*, 2018]	1	0.9961	0.3344
Liu's [Liu *et al.*, 2016]	2	0.9961	0.3333
Wang's [Wang *et al.*, 2015]	2	0.9961	0.3340

5.5. *Analysis of some typical attacks*

As well known, the anti-attack ability is an important criterion for evaluating the safety of encryption algorithms, and chosen plaintext attack is the most effective attack in modern cryptography, so the ability of anti-chosen plaintext attack needs to be improved when designing encryption scheme [Chai *et al.*, 2019]. In our scheme, the initial keys are related to MD5 hash value of the original image. Therefore, the different key streams will be used



(a) (b) (c)

(d) (e) (f)

Fig. 9. Experimental results of special images: (a) and (d) Special images, (b) and (e) encrypted images and (c) and (f) histogram of encrypted images.

Table 7. The encryption results of special images.

	Entropy	Correlation Coefficient		
		Horizontal	Vertical	Diagonal
Cipher image for all black image	7.9993	−0.0021	0.0044	−0.0069
Cipher image for all white image	7.9992	−0.0200	0.0131	0.0031

when encrypting different images. Hence, the key stream retrieved with one chosen-plain image cannot decrypt other cipher images correctively, which illustrates that chosen plaintext attacks are not valid.

5.5.1. *Experimental results' analysis of special plain images*

Some cryptanalysts try to attack an encryption algorithm by using all black or white images as plain image. We select special images as input images, and the size of images for testing are 512×512. In Fig. 9 and Table 7, the histograms with uniform distribution show that attackers cannot obtain available information by performing the encryption scheme with special images.

5.5.2. *Attack test*

Some image encryptions have good performance analysis' results, but they are easily attacked by chosen plaintext [Wang *et al.*, 2012; Pak & Huang, 2017; Zhen *et al.*, 2015; Song *et al.*, 2013; Ye, 2010]. In order to verify the anti-chosen plaintext attack ability of the proposed encryption scheme, we adopted chosen plaintext attack to test the proposed algorithm and Yin's algorithm [Yin & Wang, 2018]. As shown in Fig. 10, the image of Lena is encrypted by using our algorithm and Yin's algorithm, respectively. The recovered images of cipher images are obtained after using chosen plaintext

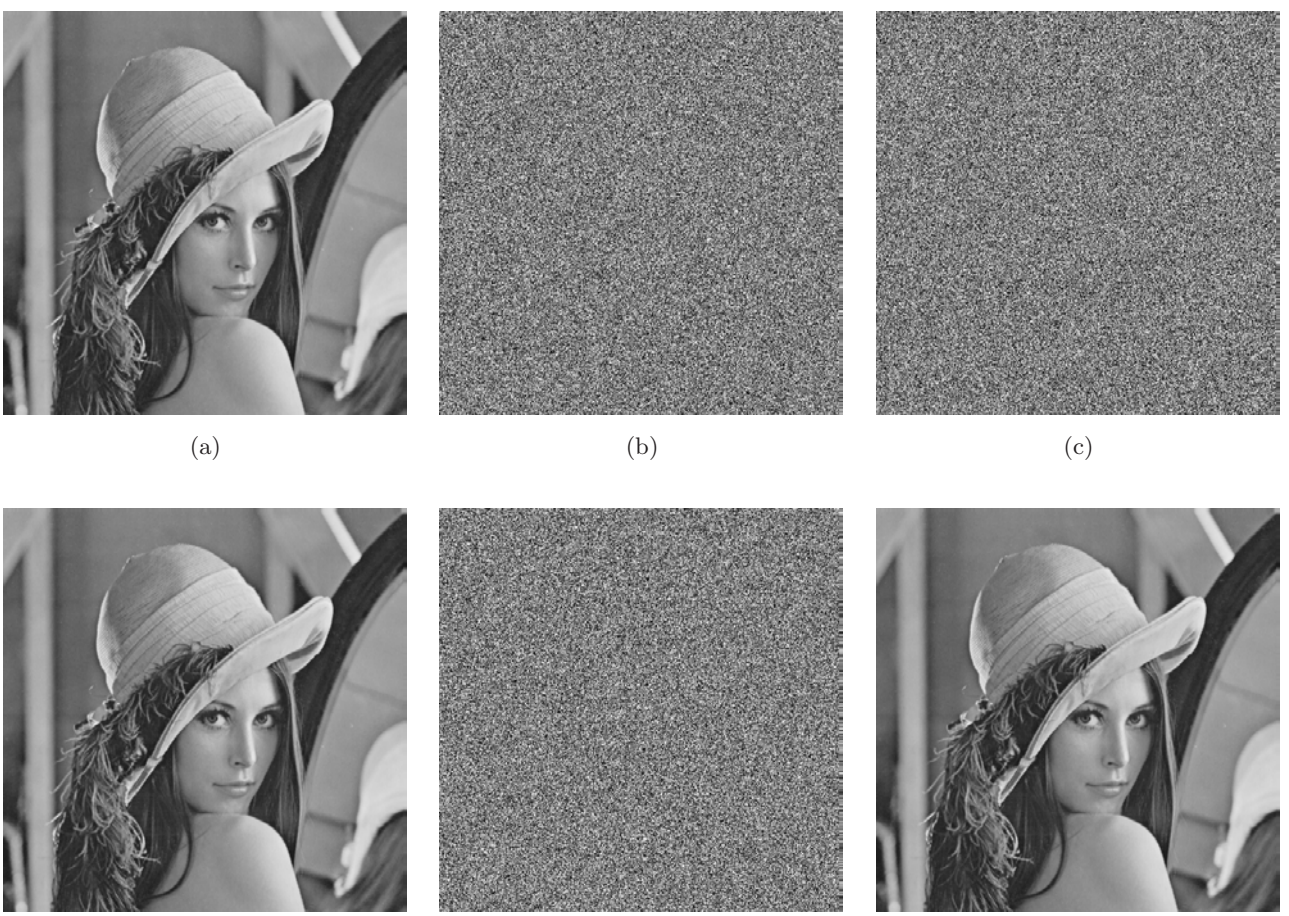


Fig. 10. Attack test: (a) and (d) Lena image (512×512), (b) and (e) cipher image of Lena using our proposed algorithm and Yin's algorithm, respectively and (c) and (f) recovered images of (b) and (e), respectively.

Table 8. The encryption time comparison results.

Algorithms	Images	Time(s)
Ours	Lena (256×256)	0.132
	Butterfly (512×768)	0.592
	Terrace (1200×1280)	0.943
Yin's [Yin & Wang, 2018]	Lena (256×256)	0.417
Zhu's [Zhu *et al.*, 2017]	Lena (256×256)	1.532
Chai's [Chai *et al.*, 2018]	Lena (256×256)	0.580

Table 9. Results of comparison.

Comparing Parameter	Ours	Li's [Li *et al.*, 2018]	Wang's [Wang *et al.*, 2018b]	Sukalyan's [Sukalyan *et al.*, 2019]
Attack analysis	Yes	No	Yes	Yes
Key space	2^{448}	2^{428}	4.5×10^{114}	10^{45}
Entropy	7.9993	7.9993	7.9971	7.9965
NPCR	0.9961	0.9961	0.9959	0.9897
UACI	0.3347	0.3348	0.3345	0.3218
Execution time(s)	0.425	0.926	1.243	0.494

attack to crack the proposed algorithm and Yin's algorithm, respectively. The results show that the plain image can be successfully recovered when using chosen plaintext attack on Yin's scheme, while the chosen plaintext attack on the proposed algorithm was failed. Conclusively, the proposed encryption scheme has the ability of anti-chosen plaintext attack.

5.6. *Analysis of speed performance*

Regardless of the security performance, encryption speed is also important, especially in real-time internet applications. The speed test of the schemes are operated using Matlab (R2014a) on a PC with Windows 7 64-bit operation system, 2.2 GHz CPU and 4 GB RAM. The images of Lena, Butterfly and Terrace are tested, the comparison results of one round of encryption execution time are listed in Table 8. It is clear that the execution time of our scheme is shorter than the other schemes. Therefore, the proposed scheme is efficient.

6. Comparison with State-of-the-Art Schemes

In order to demonstrate the good encryption performance, the proposed scheme is compared with some state-of-the-art schemes. The comparison results are listed in Table 9. For key space analysis, the proposed scheme offers a reasonably large key space. In contrast to the encryption schemes in [Wang *et al.*, 2018b; Sukalyan *et al.*, 2019], the information entropy is more close to the ideal value. The NPCR and UACI values in this paper are also pretty close to the ideal values. In addition, the attack analysis illustrated the proposed scheme has the ability of anti-chosen plaintext attack. Moreover, execution time of the proposed scheme is faster than the schemes in [Li *et al.*, 2018; Wang *et al.*, 2018b; Sukalyan *et al.*, 2019]. Therefore, the proposed scheme is feasible and effective.

7. Conclusion

An encryption scheme based on random walk and two hyperchaotic systems is proposed. It is divided into three phases: permutation within block, permutation between blocks and diffusion. In the proposed permutation within block, pixels of each block image are scrambled based on random walk matrix. The permutation method can effectively break the high correlation between neighborhood pixels in plain image. Additionally, the initial keys are related to MD5 hash value of original image. We carry out many experimental tests and analyses, the results indicate that our encryption scheme provides high security.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61971185), and

the Open Fund Project of Key Laboratory in Hunan Universities (No. 18K010).

References

Alvarez, G. & Li, S. J. [2006] "Some basic cryptographic requirements for chaos-based cryptosystems," *Int. J. Bifurcation and Chaos* **16**, 2129–2151.

Chai, X. L., Chen, Y. R. & Broyde, L. [2017a] "A novel chaos-based image encryption algorithm using DNA sequence operations," *Opt. Lasers Eng.* **88**, 197–213.

Chai, X. L., Gan, Z. H., Chen, Y. R. *et al.* [2017b] "A visually secure image encryption scheme based on compressive sensing," *Sign. Process.* **134**, 35–51.

Chai, X. L., Gan, Z. H., Yang, K. *et al.* [2017c] "An image encryption algorithm based on the memristive hyperchaotic system, cellular automata and DNA sequence operations," *Sign. Process.*: *Image Commun.* **52**, 6–19.

Chai, X. L., Zheng, X. Y., Gan, Z. H. *et al.* [2018] "An image encryption algorithm based on chaotic system and compressive sensing," *Sign. Process.* **148**, 124–144.

Chai, X. L., Fu, X. L., Gan, Z. H. *et al.* [2019] "A color image cryptosystem based on dynamic DNA encryption and chaos," *Sign. Process.* **155**, 44–62.

Chen, L., Ma, B., Zhao, X. H. & Wang, S. H. [2017] "Differential cryptanalysis of a novel image encryption algorithm based on chaos and Line map," *Nonlin. Dyn.* **87**, 1797–1807.

Chen, J. X., Han, F. F., Qian, W., Yao, Y. D. & Zhu, Z. L. [2018] "Cryptanalysis and improvement in an image encryption scheme using combination of the 1D chaotic map," *Nonlin. Dyn.* **93**, 2399–2413.

Enayatifar, R., Sadaei, H. J., Abdullah, A. H. *et al.* [2015] "A novel chaotic based image encryption using a hybrid model of deoxyribonucleic acid and cellular automata," *Opt. Lasers Eng.* **71**, 33–41.

Gao, T. G., Chen, Z. Q., Yuan, Z. Z. *et al.* [2006] "Hyperchaos generated from Chen's system," *Int. J. Mod. Phys. C* **17**, 471–478.

Hu, T., Liu, Y., Gong, L. H. *et al.* [2017a] "An image encryption scheme combining chaos with cycle operation for DNA sequences," *Nonlin. Dyn.* **87**, 51–66.

Hu, G. Q., Xiao, D., Wang, Y. & Li, X. Y. [2017b] "Cryptanalysis of a chaotic image cipher using Latin square-based confusion and diffusion," *Nonlin. Dyn.* **88**, 1305–1316.

Li, S. J., Chen, G. R., Cheung, A., Bhargava, B. & Lo, K. T. [2007] "On the design of perceptual MPEG-video encryption algorithms," *IEEE Trans. Circuits Syst. Video Technol.* **17**, 214–223.

Li, Z., Peng, C. G., Li, L. R. *et al.* [2018] "A novel plaintext-related image encryption scheme using hyper-chaotic system," *Nonlin. Dyn.* **94**, 1319–1333.

Lin, H. R. & Wang, C. H. [2020] "Influences of electromagnetic radiation distribution on chaotic dynamics of a neural network," *Appl. Math. Comput.* **369**, 124840.

Liu, W. H., Sun, K. H. & Zhu, C. X. [2016] "A fast image encryption algorithm based on chaotic map," *Opt. Lasers Eng.* **84**, 26–36.

Mao, Y., Chen, G. R. & Lian, S. G. [2004] "A novel fast image encryption scheme based on 3D chaotic baker maps," *Int. J. Bifurcation and Chaos* **14**, 3613–3624.

Matthews, R. [1989] "On the derivation of a 'chaotic' encryption algorithm," *Cryptologia* **13**, 29–42.

Norouzi, B., Mirzakuchaki, S., Seyedzadeh, S. *et al.* [2014] "A simple, sensitive and secure image encryption algorithm based on hyper-chaotic system with only one round diffusion process," *Multimed. Tools Appl.* **71**, 1469–1497.

Pak, C. & Huang, L. L. [2017] "A new color image encryption using combination of the 1D chaotic map," *Sign. Process.* **138**, 129–137.

Pearson, K. [1905] "The problem of the random walk," *Nature* **72**, 294.

Song, C. Y., Qiao, Y. L. & Zhang, X. Z. [2013] "An image encryption scheme based on new spatiotemporal chaos," *Optik* **124**, 3329–3334.

Sukalyan, S., Abhijit, M., Sarbani, P. *et al.* [2019] "A selective bitplane image encryption scheme using chaotic maps," *Multimed. Tools Appl.* **78**, 10373–10400.

Wang, X. & Wang, M. J. [2008] "A hyperchaos generated from Lorenz system," *Physica A* **387**, 3751–3758.

Wang, X. Y., Yang, L., Liu, R. *et al.* [2010] "A chaotic image encryption algorithm based on perceptron model," *Nonlin. Dyn.* **62**, 615–621.

Wang, X. Y., Teng, L. & Qin, X. [2012] "A novel colour image encryption algorithm based on chaos," *Sign. Process.* **92**, 1101–1108.

Wang, X. Y., Wang, Q. & Zhang, Y. Q. [2015] "A fast image algorithm based on rows and columns switch," *Nonlin. Dyn.* **79**, 1141–1149.

Wang, L. Y., Song, H. J. & Liu, P. [2016] "A novel hybrid color image encryption algorithm using two complex chaotic systems," *Opt. Lasers Eng.* **77**, 118–125.

Wang, C. Q., Zhang, X. & Zheng, Z. M. [2017a] "An efficient image encryption algorithm based on a novel chaotic map," *Multimed. Tools Appl.* **76**, 24251–24280.

Wang, C. H., Liu, X. M. & Xia, H. [2017b] "Multi-piecewise quadratic nonlinearity memristor and its $2N$-scroll and $2N + 1$-scroll chaotic attractors system," *Chaos* **27**, 033114.

Wang, H., Xiao, D., Chen, X. & Huang, H. Y. [2018a] "Cryptanalysis and enhancements of image encryption using combination of the 1D chaotic map," *Sign. Process.* **144**, 444–452.

Wang, X. Y., Zhu, X. Q. & Zhang, Y. Q. [2018b] "An image encryption algorithm based on Josephus traversing and mixed chaotic map," *IEEE Access* **6**, 23733–23746.

Xu, L., Li, Z., Li, J. & Hua, W. [2016] "A novel bit-level image encryption algorithm based on chaotic maps," *Opt. Lasers Eng.* **78**, 17–25.

Ye, G. D. [2010] "Image scrambling encryption algorithm of pixel bit based on chaos map," *Patt. Recogn. Lett.* **31**, 347–354.

Yin, Q. & Wang, C. H. [2018] "A new chaotic image encryption scheme using breadth-first search and dynamic diffusion," *Int. J. Bifurcation and Chaos* **28**, 1850047-1–13.

Zhang, Y. Q., Wang, X. Y., Liu, J. *et al.* [2016] "An image encryption scheme based on the MLNCML system using DNA sequences," *Opt. Lasers Eng.* **82**, 95–103.

Zhang, X. & Wang, C. H. [2019a] "A novel multi-attractor period multi-scroll chaotic integrated circuit based on CMOS wide adjustable CCCII," *IEEE Access* **7**, 16336–16350.

Zhang, X. & Wang, C. H. [2019b] "Multiscroll hyperchaotic system with hidden attractors and its circuit implementation," *Int. J. Bifurcation and Chaos* **29**(9), 1950117.

Zhao, Q., Wang, C. H. & Zhang, X. [2019] "A universal emulator for memristor, memcapacitor, and meminductor and its chaotic circuit," *Chaos* **29**, 013141.

Zhen, P., Zhao, G., Min, L. Q. *et al.* [2015] "Chaos-based image encryption scheme combining DNA coding and entropy," *Multimed. Tools Appl.* **75**, 6303–6319.

Zhou, Y., Bao, L. & Chen, C. L. P. [2014] "A new 1D chaotic system for image encryption," *Sign. Process.* **97**, 172–182.

Zhou, L., Wang, C. H. & Zhou, L. L. [2016a] "Generating hyperchaotic multi-wing attractor in a 4D memristive circuit," *Nonlin. Dyn.* **85**, 2653–2663.

Zhou, N. R., Pan, S. M., Cheng, S. *et al.* [2016b] "Image compression-encryption scheme based on hyperchaotic system and 2D compressive sensing," *Opt. Laser Technol.* **82**, 121–133.

Zhou, L., Wang, C. H. & Zhou, L. L. [2017] "Generating four-wing hyperchaotic attractor and two-wing, three-wing, and four-wing chaotic attractors in 4D memristive system," *Int. J. Bifurcation and Chaos* **27**, 1750027-1–14.

Zhou, L., Wang, C. H., Zhang, X. & Yao, W. [2018a] "Various attractors, coexisting attractors and antimonotonicity in a simple fourth-order memristive twin-T oscillator," *Int. J. Bifurcation and Chaos* **28**, 1850050-1–18.

Zhou, L., Wang, C. H. & Zhou, L. L. [2018b] "A novel no-equilibrium hyperchaotic multi-wing system via introducing memristor," *Int. J. Circuit Th. Appl.* **46**, 84–98.

Zhou, M. J. & Wang, C. H. [2020] "A novel image encryption scheme based on conservative hyperchaotic system and closed-loop diffusion between blocks," *Signal Process.* **171**, 107484.

Zhu, C. X. [2012] "A novel image encryption scheme based on improved hyperchaotic sequences," *Opt. Commun.* **285**, 29–37.

Zhu, H. G., Zhang, X. D., Yu, H. *et al.* [2017] "An image encryption algorithm based on compound homogeneous hyper-chaotic system," *Nonlin. Dyn.* **89**, 61–79.