Multimedia Tools and Applications (2019) 78:31035–31055
https://doi.org/10.1007/s11042-018-7081-3

Color image chaos encryption algorithm combining CRC and nine palace map

Zenggang Xiong [1,2] · Yuan Wu [1,2] · Conghuan Ye [1,2] · Xuemin Zhang [1] · Fang Xu [1,2]

Received: 21 September 2018 / Revised: 29 November 2018 / Accepted: 12 December 2018 /
Published online: 9 January 2019

Abstract
The color image encryption algorithm based on the chaos theory is not strong enough. In this paper, we proposed a color image chaos encryption algorithm combining Cyclic Redundancy Check (CRC) and nine palace map. Firstly, the pixel data of the plain image were moved and shuffled based on the theory of nine palace map. And the R, G and B components were extracted and converted into a binary sequence matrix that was then cyclically shifted based on the technology of generating CRC code. Finally, the encrypted image was derived from the XOR operation with random key matrix. The average entropy of encrypted image by our algorithm is 7.9993, which is slight improved compared with the coupled hyper chaotic Lorenz algorithm in previous studies. In addition, the algorithm has the advantages of large key space, high key sensitivity, anti-robust attack, and feasible encryption efficiency.

Keywords Color image encryption · Cyclic redundancy check (CRC) · Nine palace map · Logistic map · Cross shift

1 Introduction

With the advent of the information age, multi-media technology has been developing rapidly. People are enjoying the convenience of technology while accessing information on the Internet. However, frequent data breaches have also brought problems in

✉ Zenggang Xiong
xzg@hbeu.edu.cn; jkxxzg2003@163.com

✉ Yuan Wu
yuanywu@126.com

[1] School of Computer and Information Science, Hubei Engineering University, Xiaogan 432000 Hubei, China

[2] School of Computer Science and Information Engineering, Hubei University, Wuhan 430062 Hubei, China

people's lives. For example, in September 2017, Equifax, one of the largest credit reporting agencies in the United States, was subject to unauthorized access, and the information of approximate 143 million consumers have been leaked; in February 2016, $81 million of the Central Bank of Bangladesh was stolen and $31 million of the central bank of Russian was stolen. And excessive data collection can also cause the personal information to be leaked [14]. In order to create a comfortable, healthy and safe social environment, it is necessary to encrypt the transmitted images.

In the early days, people designed encryption methods such as DES, RSA and AES to encrypt textual data. However, these conventional cipher algorithms are not suitable for the multimedia data with a high correlation between adjacent pixels and a large data capacity, more and more new technologies have been being proposed. Martin et al. [18] introduced a method to achieve the compression and partial encryption by selecting the significance bits for each wavelet coefficient and combining Color-SPIHT(C-SPIHT) compression. Kanso and Ghebleh [11] proposed a lossless visually meaningful image encryption scheme using existing cipher images as watermarks to embed wavelet coefficients to make the encrypted content more difficult to detect. Mahesh et al. [17] presented an encryption algorithm based on Arnold Cat Map and discrete Haar wavelet transform. In addition to combining wavelet transforms, DNA coding is also a common technique[1, 25]. Kumar et al. [12] introduced an algorithm of the color image encryption. In this algorithm, each component of the color image was encoded by doing deoxyribonucleic acid (DNA) sequence operations and symmetric encryption based on elliptic curve cryptography. Chai et al. [4] introduced a scheme of the image encryption based on chaotic system and DNA encoding. Zhang et al. [34] presented a new algorithm, which was mainly based on the spatiotemporal chaos of the Mixed Linear-Nonlinear Coupled Map Lattices (MLNCML). This algorithm not only can enhance the sensitivity to the plain image, but also can resist different attacks by employing the strategy of DNA computing and one time pad encryption policy. In 1989, the chaos-based encryption thought was proposed by the British mathematician Matthews. After that, the application of chaos theory has been being paid more and more attention. Chaos technology [2, 3, 8, 13, 24] has become a common technique in the field of image encryption due to its inherent characteristics. Logistic maps [10, 30], Arnold transformation [37], 3-D Cat maps [6], and Henon maps [5, 21, 27] are representative chaotic map techniques.

Although the technology of image encryption has become more and more sophisticated in recent years [19, 31, 32], chaos encryption algorithms for color images still have some shortcomings. Kadir et al. [9] used a coupled hyper chaotic Lorenz systems to encrypt color images, but the encryption algorithm was simple. Liu et al. [16] introduced a stream-cipher scheme, which was a combination of a one-time key and robust chaotic maps. Through scrambling and diffusion processes, it can achieve high security and dynamic degradation. The correlation between adjacent pixels was not well eliminated because the pixel disruption is not thorough enough.

In order to improve the encryption performance of the color images, we combine the CRC and the nine palace map. With the help of the principle of the nine palace map, the pixels of the plain image are initially scrambled. The attackers need to know not only the rules of the initial transformation, but also which one of them is scrambled. It can improve the security of encryption. The CRC generation mechanism is used to generate the chaotic key in the cyclic shift phase, and the pixel values are completely disorganized at the bit-level. The results of the experiments show that the algorithm has superior high

security property and can resist different attacks. However, the simple encryption algorithms will eliminate spatial redundancy as much as possible, if the encrypted images need to be compressed then stored, it will inevitably affect the compression ratio, and the image quality will be affected. At the same time, because lossy compression of images such as JPEG format will selectively lose a portion of the data, the algorithm proposed in this paper is mainly performed at the bit-level, and slightly changing the bit values will make the compressed image difficult to recover. Therefore, the images in lossless compressed or uncompressed fit the algorithm studied in this paper.

In recent years, researchers have been actively looking for new applications for encryption [38, 40, 41]. One of the methods is to introduce the encryption technology into cloud storage, which can solve the problem of privacy leakage caused by daily sharing of social photos. For example, Sosa et al. [26] designed the Privacy-enhancing Image-based Collaborative File System and proved its practicality in the real world.

This paper is organized as follows: Section 2 introduces theoretical analysis; Section 3 introduces the proposed algorithm in detail; Section 4 shows a variety of simulation results; In Section 5, we make a summary of our work.

2 Theoretical analysis

2.1 Nine palace map system

The Nine palace map, named the rearrangement of the Nine Palaces, is a digital game. The rule of the which is to put 1~9 numbers into a 3 × 3 grid so that the sums of the numbers in each row, each column and either diagonal should be 15.

In the encryption algorithm mentioned in this paper, the principle of nine palace map is applied to the initial scrambling of image pixels. Each component of RGB images can be evenly divided into 9 blocks numbered 1 to 9, as shown in Fig. 1 (a). According to the layout of the nine palace map, the blocks in Fig. 1(a) are re-disrupted in a certain order to form a new map as shown in (c): the numbered block 1 is placed in the block 2 of (b); the numbered block 2 is placed in the block 9 of (b); numbered block 3 is placed in block 4 of (b); numbered block 4 is placed in block 7 of (b); numbered block 6 is placed in block 3 of (b); numbered block 7 is placed in block 6 of (b); numbered block 8 is placed in block 1 of (b); numbered block 9 is placed in block 8 of (b). When all nine blocks in (a) are rearranged, the nine palace map is completed as shown in (c). Since the position

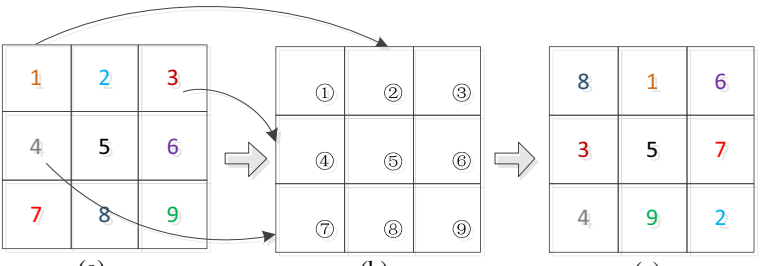

Fig. 1 Nine palace map scrambling

of the numbered block 5 has not changed, the block 5 is transposed in the original position to change the position of pixel data. To make the pixels scrambled in their entirety, each channel of the plain image follow placement rules above to achieve the scrambling of the initial position of the pixels.

2.2 Logistic map

1D Logistic map, widely used in image encryption, is the simplest nonlinear chaotic system, which can be expressed by

$$\mathbf{x}_{i+1} = \mu \mathbf{x}_i (1 - \mathbf{x}_i) \tag{1}$$

Where x_0 defines the initial value of the condition,parameterμdenotes the control coefficient of the system, subscript $i$denotes the number of iterations of the system. When$\mu \in (3.57, 4)$, the system is in chaos, $\mathbf{x}_{i+1} \in (0, 1)$.

2.3 Cyclic redundancy check (CRC)

CRC is the cyclic redundancy check, which is the most commonly used method of checking errors in the communication field. It can perform polynomial calculation on data and attach the obtained result to the back of the frame. The entire process ensures the correctness and completeness of the data transmission.

When this encoding is used, the sender and the receiver must agree in advance to create a multi-pattern whose first and last bits must be '1' at the same time. Any code consisting of a series of binary bits can correspond to multiple styles where the coefficients only take values of '0' and '1'. Table 1 lists several CRC versions of commonly used international standards.

The definition is as follows:

$$\frac{C(x) \times 2^r}{G(x)} = R_0(x) + X(x) \tag{2}$$

$$CRCcode = \mathrm{C}(x) \times 2^r + R_0(x) \tag{3}$$

Where$X(x)$is quotient. Suppose the information to be transmitted is represented by polynomial$C(x)$, and $C(x)$ is shifted to the left by $r$bit (can be expressed as$C(x) \times 2^r$).

Table 1 CRC polynomial of commonly used international standards

Name	Polynomial $G(x)$	Corresponding binary number $F(x)$
CRC-4	$x^4 + x + 1$	'10011'
CRC-8	$x^8 + x^5 + x^4 + 1$	'100110001'
CRC-8	$x^8 + x^2 + x + 1$	'100000111'
CRC-8	$x^8 + x^6 + x^4 + x^3 + x^2 + x$	'101011110'
CRC-12	$x^{12} + x^{11} + x^3 + x + 1$	'1100000001011'
CRC-16	$x^{16} + x^{15} + x^2 + 1$	'11000000000000101'

Then, r bit is vacated on the right side of$C(x)$, which is the position of the check code. The modulo 2 division (high-alignment) of $C(x) \times 2^r$is performed using the binary number $F(x)$of the generator polynomial, and the remainder, the redundant bit $R_0(x)$, is the check code. For example, the sent information polynomial $C(x)$is '1010001', the binary number$F(x)$ corresponding to the polynomial $G(x)$is '10111'. Move $C(x)$ to the left by $r$bit (the number of bits in $F(x)$is reduced by 1) to get '10100010000', and then the binary number is used to divide the moved information polynomial, which is equivalent to XOR, and the remainder $R_0(x)$ is '1101'. Finally, $R_0(x)$will be placed in the position where the information code $C(x)$ is moved, and the complete CRC code can be obtained.

3 Image encryption algorithm

The process of the encryption algorithm is shown in Fig. 2. Each specific process is described as follows.

Step 1: Import the plain RGB image $P(M \times N \times 3)$, where$M$and$N$are the size of row and column. Each component of the RGB is placed based on the same rule of the nine palace map as described in Section 2.1.

Step 2: Let the 1D Logistic map iterate $(l+m)$ times, and only keep the latter $m$elements to obtain a new sequence$\mathbf{sp} = \{\mathbf{x}_{l+1}, \mathbf{x}_{l+2}, \mathbf{x}_{l+3}, ..., \mathbf{x}_{l+m}\}$, where$m = 24$.

Step 3: Obtain three 1D sequences with the length of $MN$by converting each color component in row-major order,$\mathbf{R_P}=\{R_1,R_2,...,R_{MN}\}$,$\mathbf{G_P}=\{G_1,G_2,...,G_{MN}\}$ and$\mathbf{B_P}=\{B_1, B_2,...,B_{MN}\}$.

Each sequence is equally divided into 3 blocks. r_i is denoted as the i-th block of $\mathbf{R_P}$,g_i is denoted as the i-th block of $\mathbf{G_P}$,$b_i$is denoted as the i-th block of $\mathbf{B_P}$, where $i = \{1, 2, 3\}$.

Step 4: Obtain the matrix**rowRGB**with the size of $(3 \times MN)$.

$$\begin{pmatrix} r_1 & r_2 & r_3 \\ g_1 & g_2 & g_3 \\ b_1 & b_2 & b_3 \end{pmatrix} \rightarrow \mathbf{rowRGB} = \begin{pmatrix} \mathbf{g_3} & \mathbf{r_3} & \mathbf{b_3} \\ \mathbf{g_1} & \mathbf{r_1} & \mathbf{b_1} \\ \mathbf{g_2} & \mathbf{r_2} & \mathbf{b_2} \end{pmatrix} \tag{4}$$

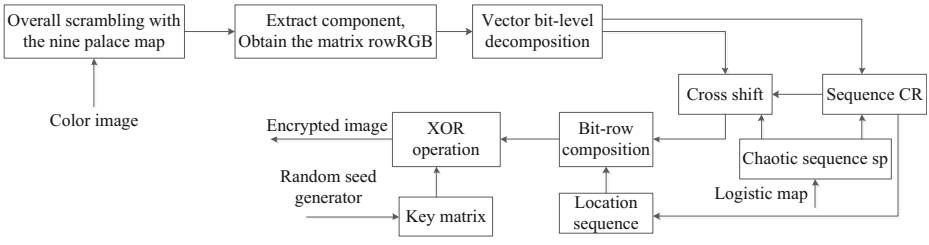

Fig. 2 Structure of the proposed encryption algorithm

Step 5: Obtain the binary matrix **Q**with the size of $(3 \times 8) \times MN$ by doing bit-level decomposition with each row of elements of**rowRGB**, where$n$in$\mathbf{Q}_i^n$represents the$n$-th bit-level.

$$\mathbf{rowRGB} = \begin{pmatrix} g_3 & r_3 & b_3 \\ g_1 & r_1 & b_1 \\ g_2 & r_2 & b_2 \end{pmatrix} \rightarrow \mathbf{Q} = \begin{pmatrix} g_3^1 & r_3^1 & b_3^1 \\ \vdots & \vdots & \vdots \\ g_3^8 & r_3^8 & b_3^8 \\ g_1^1 & r_1^1 & b_1^1 \\ \vdots & \vdots & \vdots \\ g_1^8 & r_1^8 & b_1^8 \\ g_2^1 & r_2^1 & b_2^1 \\ \vdots & \vdots & \vdots \\ g_2^8 & r_2^8 & b_2^8 \end{pmatrix} \tag{5}$$

Step 6: Obtain a sequence **CR** according to each row of **Q**. The pseudo code of the corresponding step is shown in Fig. 3.

for $(i = 1; i \leq 3 \times 8; i++)$.

{Suppose the vector of the i-th row of **Q**is**L**. Sum the values of the elements in**L**by the following equation.

$$\mathbf{Sum}(i) = sum(L); \tag{6}$$

Obtain the**q**by the following.

$$\mathbf{q}(i) = \mathbf{Sum}(i) + \mathbf{sp}(i) \times 10^6 \bmod 2^8; \tag{7}$$

Let the binary of $\mathbf{q}(i)$be the information polynomial$C(x)$, which is done modulo-2 division operation by$F(x)$, and the decimal number $\mathbf{CR}(i)$ of the CRC code is obtained.

}

Step 7: Cross shift

for $(i = 1; i \leq 3 \times 8; i++)$// row shift.

$$\{\mathbf{len}(i) = \mathrm{mod}\left(floor\left(\mathbf{sp}(i) \times 10^6 + \mathbf{CR}(i) \times 10^2\right), MN\right) + 1 \tag{8}$$

If i is an odd, all elements of the row are cyclic shifted to the.

left by $\mathbf{len}(i)$ elements;

Else i is an even, all elements of the row are cyclic shifted to the right by $\mathbf{len}(i)$ elements;

}

for $(j = 1; i \leq MN; j++)$// column shift.

$$\{leng = \mathrm{mod}(j, 3 \times 8) + 1 \tag{9}$$

$$\mathbf{len}(j) = \mathrm{mod}(\mathbf{CR}(leng), 3 \times 8) + 1 \tag{10}$$

If the$j$-th column of matrix **Q** is odd, all elements of the column are cyclic shifted to the up by **len**(j)elements;

Else all elements of the column are cyclic shifted to the down by **len**(j) elements;

}

Step 8: According to the location sequence of$sort$(**CR**), the rows of the **Q**are rearranged. Convert 1st~8th bit-rows, 9th~16th bit-rows and 17~24th bit-rows to three scrambled 1D vectors. Reshape them to three matrices$\mathbf{I}_R$,$\mathbf{I}_G$,$\mathbf{I}_B$with the size of $M \times N$.

Step 9: Generate a random key matrix $\mathbf{S} \in [1255]$ with the size of $M \times N$using the pseudo-random seed generator$k$. Obtain three matrices$\mathbf{C_R}$,$\mathbf{C_G}$ and $\mathbf{C_B}$with key matrix**S**.

$$\begin{aligned} \mathbf{C_R} &= \mathbf{I_R} \oplus \mathbf{S} \\ \mathbf{C_G} &= \mathbf{I_G} \oplus \mathbf{S} \\ \mathbf{C_B} &= \mathbf{I_B} \oplus \mathbf{S} \end{aligned} \tag{11}$$

Matrix$\mathbf{C_R}$,$\mathbf{C_G}$ and $\mathbf{C_B}$are combined and the color cipher image is obtained.

The steps of decryption can be seen as the reverse of encryption. As shown in Fig. 4.

Input: Matrix Q of size $(r \times MN)$, sequence sp of size (1×24), secret key $F(x)$

Step 6 for $i = 1$ to r do

$Sum(i) \leftarrow sum(Q(i,:))$;

$q(i) \leftarrow Sum(i) + sp(i) \times 10^6 \bmod 2^8$;

$C(x) \leftarrow dec2bin(q(i))$; // $C(x)$ is the information polynomial

CRC code ← ($F(x)$ modulo-2 division $C(x)$); // $F(x)$ is the generating polynomial (binary).

$CR(i) \leftarrow bin2dec(CRC\ \text{code})$;

end

Step 7 for $i = 1$ to r do // The row shift of matrix Q

$len(i) \leftarrow \mathrm{mod}(floor(sp(i) \times 10^6 + CR(i) \times 10^2), MN) + 1$;

if i =odd, $circshift(Q(i,:), shift\ len(i)\ lengths\ to\ the\ left)$;

else $circshift(Q(i,:), shift\ len(i)\ lengths\ to\ the\ right)$;

end

end

for $j = 1$ to MN do // The column shift of matrix Q

$leng \leftarrow \mathrm{mod}(j, r) + 1$;

$len(j) \leftarrow \mathrm{mod}(CR(leng), r) + 1$;

if j =odd, $circshift(Q(i,:), shift\ len(i)\ lengths\ to\ the\ up)$;

else $circshift(Q(i,:), shift\ len(i)\ lengths\ to\ the\ down)$;

end

end

Output: Matrix Q' obtained by cyclically shifting matrix Q.

Fig. 3 Pseudo code for steps 6 and 7 of the encryption algorithm

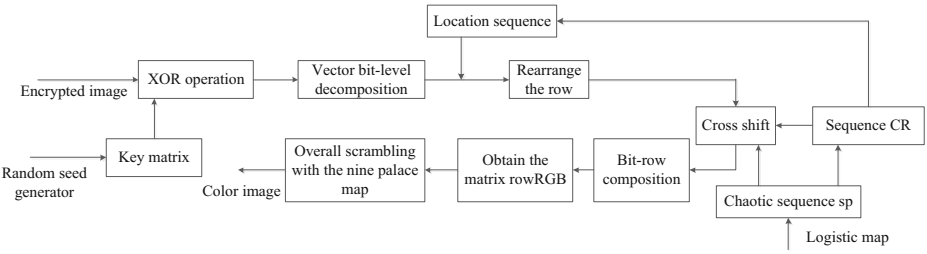

Fig. 4 Structure of the proposed decryption algorithm

4 Simulation results

For the algorithm proposed in this paper, we made various simulation experiments with MatLab R2009a. Except for special statement in the experiment, the size of the tested images is (512× 512 × 3). The initial valuex_0=0.4, the control parameter μ=3.6,the key l = M,$F(x)$='11101' and k=5.

JPEG is a lossy compression format. By removing redundant data, the image can be compressed into a small storage space. Although the visual loss can be difficult to detect, it is easy to cause damage to the image data. Generally, simple encryption algorithms try to remove the spatial redundancy, which will affect the compression ratio and the image quality might be affected as well. Moreover, by observing the images before and after compression, it is found that the data at the bit-level is very different. So, when the encryption algorithm is operated at the bit-level, it is difficult to recover the encrypted image after compression. In Fig. 5 (a), the same image is encrypted. When the encrypted image is stored in PNG format, it can be successfully decrypted as shown in (d); when

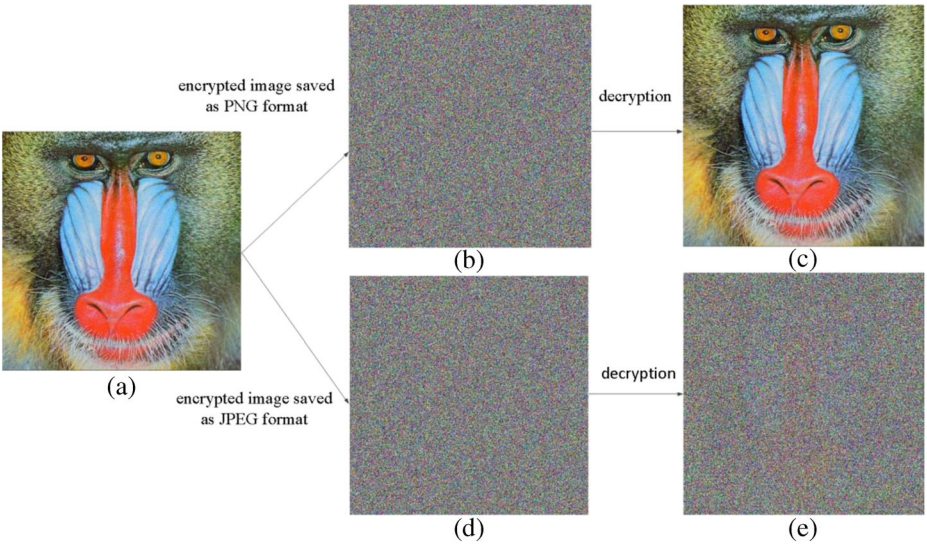

Fig. 5 Encrypted image saved as PNG and JPEG format. (**a**) Plain image Baboon, (**b**) Encrypted image saved as PNG format, (**c**) Corresponding decrypted (**b**), (**d**) Decrypted image saved as JPEG format, (**e**) Corresponding decrypted (**d**)

the encrypted image is stored in JPEG format, the cipher image cannot be successfully decrypted as shown in (e). Therefore, the images in lossless compressed or uncompressed, such as PNG images or TIFF images, fit the algorithm proposed in this paper. The color RGB test images used in the following simulation experiments are all PNG images, and the corresponding encrypted image is saved also in PNG format. Figure 6 shows the simulation results.

4.1 Key space analysis

As we all know, for an algorithm of image encryption, the enough key space has advantages of resisting brute-force attacks. The key space of the encryption algorithm proposed in this paper is closely related to the initial value$x_0$and parameter$\mu$of the Logistic map, the seed of the pseudo-random generator$k$, the polynomial$F(x)$, the integer$l$=512. Supposing that the computer precision is 10^{-16}, the number of available$x_0$andμ is $10^{16} \times 10^{16} = 10^{32}$. The$k$is an integer between 0 and 2^{32} -1. So, the number of available is2^{32}. As for the polynomial$F(x)$, the available number is 2^{29}. Therefore, our key space is $10^{32} \times 2^{32} \times 2^{29} \times 2^9 > 2^{96} \times 2^{32} \times 2^{29} \times 2^9 = 2^{166}$, as shown in Table 2. It is large enough to not only provide a high security space, but also resist brute-force attacks.

4.2 Statistical analysis

(1) Histogram analysis

The histograms [36] of cipher images generated by the ideal algorithm of encryption are uniform. Figure 7 shows the histograms of the distributions of pixel values corresponding to pre-encrypted and encrypted Baboon image. Observing the histograms in the two states shows that the distribution of pixel values before encryption is regular, but after encryption the pixel values are almost distributed on the same horizontal line, and no information in the image is displayed. It can be concluded that this algorithm can effectively resist attacks based on frequency analysis of pixel values in encrypted images. Table 2.

(2) Correlation analysis

The adjacent pixels have similar strength in the digital image and therefore there has a strong correlation. In order to avoid statistical attacks, these strong correlations must be broken. We randomly select$N$=5000 pairs of adjacent pixels$x_i$andy_i in the horizontal, vertical and diagonal directions of Baboon image, respectively. And the calculation of the correlation coefficients can be expressed by

$$r_{xy} = \frac{\mathrm{cov}(x,y)}{\sqrt{D(x)D(y)}} \tag{12}$$

where

$$\mathrm{cov}(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E[x])(y_i - E[y]) \tag{13}$$

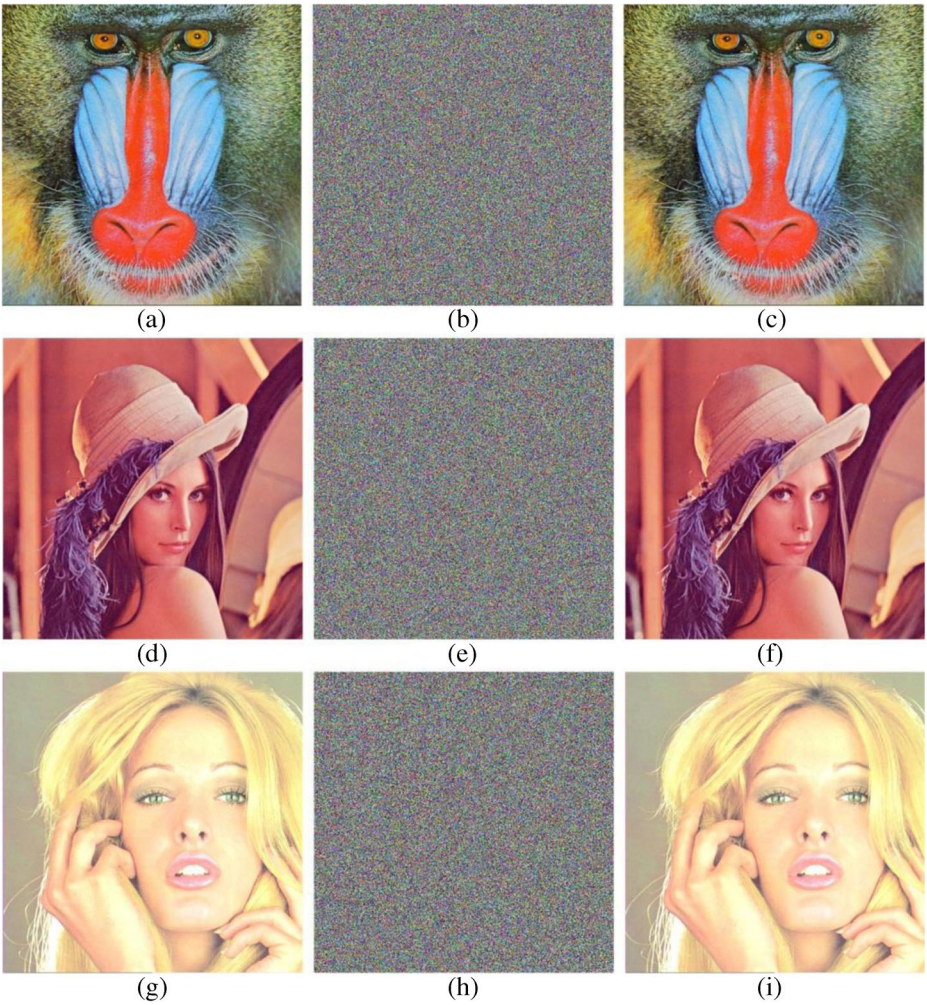

Fig. 6 Test results. (**a**) Plain image Baboon, (**b**) Baboon after encryption, (**c**) Corresponding decrypted (**b**), (**d**) Plain image Lena, (**e**) Lena after encryption, (**f**) Corresponding decrypted (**e**), (**g**) Plain image Tiffany, (**h**) Tiffany after encryption, (**i**) Corresponding decrypted (**h**)

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{14}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E[x])^2 \tag{15}$$

Figure 8 analyzes the horizontal correlation of "Baboon" image before and after encryption. From the six sub-graphs in the Fig. 8, we can see that the distributions of the horizontal pixels of the pre-encrypted and encrypted images are significantly different. The pixel values before encryption are distributed on one diagonal of the sub-graphs with high correlation, while the

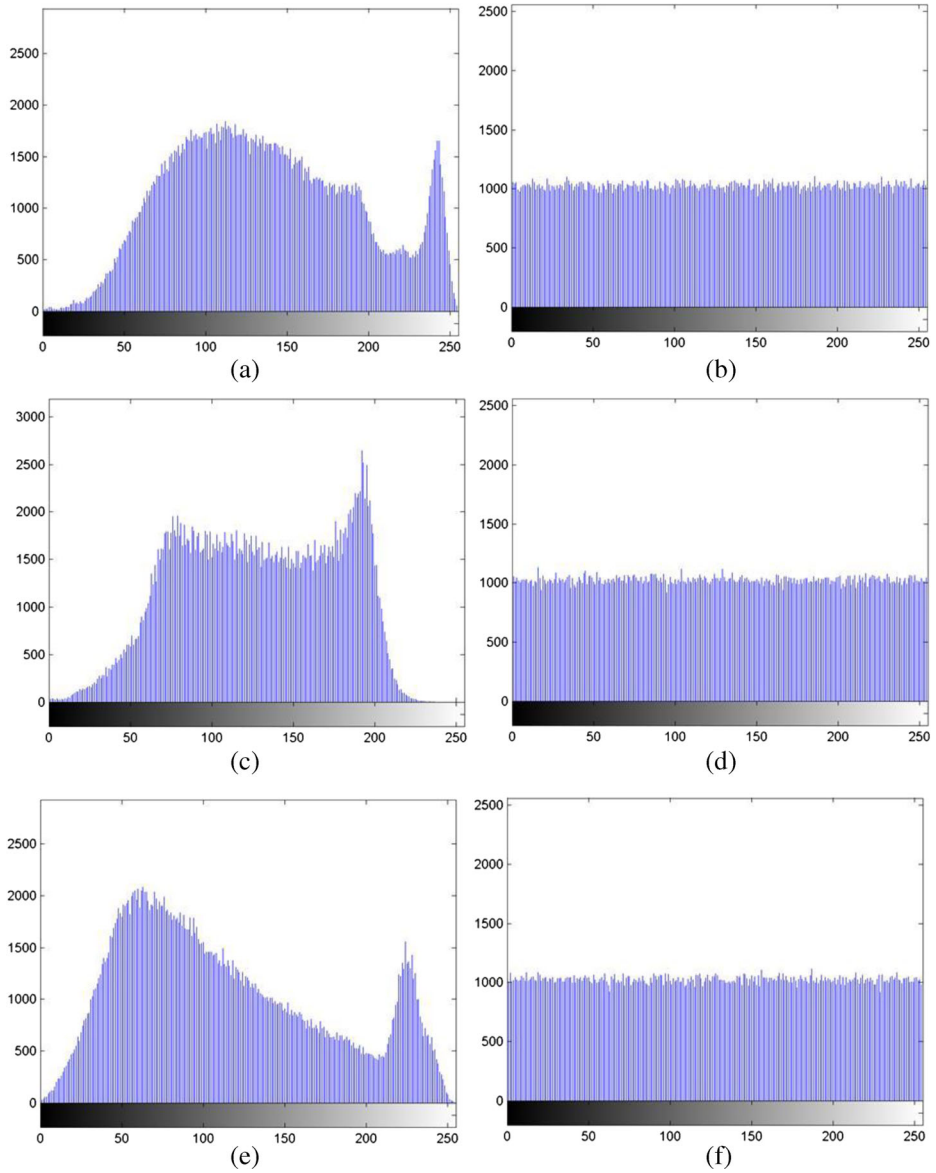

Fig. 7 Histograms analysis for image "Baboon". (**a**) R component before encryption, (**b**) R component after encryption, (**c**) G component before encryption, (**d**) G component after encryption, (**e**) B component before encryption, (**f**) B component after encryption

Table 2 Comparison of Key space

Algorithm	Key Space
Proposed	2^{166}
Ref. [12]	$2^{143.9322} \sim 2^{144.0674}$
Ref. [7]	2^{100}
Ref. [15]	2^{128}
Ref. [28]	2^{128}

encrypted pixel values are evenly distributed in the entire gray space with extremely low correlation. Table 3 lists the correlation coefficients of the groups of the test images. Compared to the plain images with the correlation coefficients close to 1, the encrypted images are effectively eliminated the correlation between adjacent pixels by pixel diffusion, and the correlation coefficients of the pixels are almost close to 0 in all three directions. Table 4 compares the algorithm proposed in this paper with the adjacent coefficients of the other algorithms. The encryption algorithm of this paper can resist statistical attacks.

4.3 Information entropy analysis

There is no doubt that the information entropy [33] is a significant consideration for measuring the randomness of the pixels in images. The closer to 8 the information entropy is, the more random the image is, which avoids the leakage of the information in the image. Suppose we define the source of the information as$m$, and the calculation formula for information entropy is as follows:

$$H(m) = \sum_{i=0}^{M-1} p(m_i)\log_2 \frac{1}{p(m_i)} \tag{16}$$

Where M denotes the total number of symbols$m_i \in m$, and $p(m_i)$ denotes the proportion of the occurrence of them_i.

The information entropy of six different test images is shown in Table 5. By analyzing the data for each set of images, we can conclude that the average value of the information entropy of the cipher images in the R, G and B components is

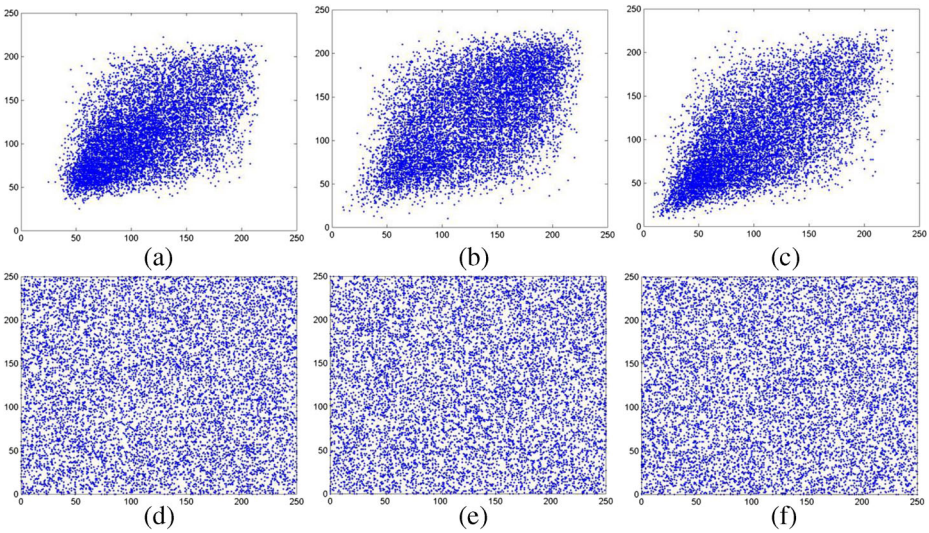

Fig. 8 Horizontal correlation of "Baboon" image. (**a**) The R component of "Baboon" before encryption, (**b**) The G component of "Baboon" before encryption, (**c**) The B component of "Baboon" before encryption, (**d**) The R component of "Baboon" after encryption, (**e**) The G component of "Baboon" after encryption, (**f**) The B component of "Baboon" after encryption

Table 3 Correlation coefficients in three directions

Images	Direction	Plain image			Cipher image		
		R component	G component	B component	R component	G component	B component
Baboon	Horizontal	0.9176	0.8650	0.9101	0.0123	0.0195	0.0049
	Vertical	0.8687	0.7664	0.8856	0.0050	0.0037	0.0143
	Diagonal	0.8612	0.7546	0.8463	−0.0055	−0.0048	0.0121
Lena	Horizontal	0.9788	0.9686	0.9289	−0.0032	−0.0076	−0.0240
	Vertical	0.9897	0.9820	0.9580	0.0141	−0.0060	−0.0084
	Diagonal	0.9725	0.9588	0.9205	0.0058	−0.0019	0.0065
Tiffany	Horizontal	0.9581	0.9039	0.9021	−0.0101	0.0156	−0.0043
	Vertical	0.9717	0.9614	0.9588	−0.0141	0.0216	0.0097
	Diagonal	0.9437	0.9003	0.8990	−0.0112	−0.0141	−0.0029
Peppers	Horizontal	0.9610	0.9803	0.9705	0.0108	−0.0012	0.0116
	Vertical	0.9643	0.9831	0.9708	0.0004	−0.0021	−0.0290
	Diagonal	0.9580	0.9740	0.9493	0.0029	0.0057	0.0071

about 7.9993, which is close to the ideal value of 8 and indicates that the algorithm of the encryption is also safe under the entropy attack.

Further more, Table 6 shows the information entropy compared with other algorithms. Kadir et al. [9] performed bitwise operations of XOR and left or right cyclic shift inside pixel values, which essentially only changed the pixel values, but the three components were completely isolated in operation. Therefore, the overall pixels were not evenly distributed. Although Teng et al. [28] permuted the pixels at bit-level, the process of integrating three components into one bit-level image was simple, and the diffusion between the three components was insufficient. Similarly, Parvaz and Zarebnia [20] repeatedly used the cyclic shift and XOR operations, but the overall pixels did not reach a state of random distribution. However, the algorithm proposed in this paper strengthens the connection between the three components, and the pixels of each component are better diffused, so that it is evenly distributed over the entire image.

The average information entropy of the proposed algorithm is higher, which is 0.0976% higher than [9], 0.0366% higher than [28], 0.0300% higher than [20] and closer to 8, suggesting higher security.

Table 4 Comparison of correlation coefficients

Lena	Orientation	R component	G component	B component
Proposed	Horizontal	−0.0032	−0.0076	−0.0240
	Vertical	0.0141	−0.0060	−0.0084
	Diagonal	0.0058	−0.0019	0.0065
Ref. [12]	Horizontal	0.0035	−0.0097	0.0186
	Vertical	−0.0041	0.0053	0.0101
	Diagonal	−0.0410	−0.0085	−0.0175
Ref. [7]	Horizontal	0.0025	0.0314	0.0140
	Vertical	−0.0131	0.0241	0.0017
	Diagonal	0.0033	0.0272	0.0110
Ref. [29]	Horizontal	−0.0127	−0.0075	−0.0007
	Vertical	0.0067	−0.0068	0.0042
	Diagonal	0.0060	−0.0078	0.0026

Table 5 Information entropy

Test images	Plain image			Cipher image		
	R component	G component	B component	R component	G component	B component
Baboon	7.7067	7.4744	7.7522	7.9994	7.9994	7.9994
Lena	7.2531	7.5940	6.9684	7.9993	7.9993	7.9994
Tiffany	4.3372	6.6643	6.4288	7.9993	7.9993	7.9994
Peppers	7.3388	7.4963	7.0583	7.9992	7.9993	7.9993
Splash	6.9481	6.8845	6.1265	7.9993	7.9993	7.9993
Airplane	6.7178	6.7990	6.2138	7.9993	7.9992	7.9993
Average	**6.7170**	**7.1521**	**6.7580**	**7.9993**	**7.9993**	**7.99935**

4.4 Key sensitivity analysis

A good cryptographic system needs to have a dependency on the secret keys to be able to resist brute-force attacks [23]. In other words, the decrypted image cannot be recovered normally when the key used in the decryption process is slightly changed. Fig. 6(c) shows the reconstructed image obtained with the correct secret keys mentioned in this paper when decrypting Fig. 6(b). To test whether the encryption algorithm mentioned above is sensitive to the secret keys during the decryption process, only one of the keys is changed slightly at a time. And the remaining keys stay unchanged, and then the cipher image is decrypted.

The test results are shown in Fig. 9 by several different sets of data changes. Fig. 9 (a) shows the decrypted results when k=6; (b) shows the decrypted results whenx_0=0.4000001; (c) shows the decrypted results when$F(x)$='10011'. It can be clearly seen that these decrypted images have not been successfully restored to the plain images, and no useful information is obtained from them. Therefore, we can conclude that the algorithm proposed in this paper has a high dependence on the secret keys.

4.5 Robust analysis

(1) Noise attack analysis

During transmission through the physical channel, digital images will inevitably experience various noise disturbances. For a cryptographic system with good encryption performance, it should be very resistant to noise attacks. In order to test the anti-noise ability of the algorithm

Table 6 Information entropy of Lena image encrypted by different algorithms

Component	Proposed		Ref. [9]	Ref. [20]	Ref. [28]	Ref. [29]
	(512 × 512)	(256 × 256)	(512 × 512)	(512 × 512)	(256 × 256)	(256 × 256)
R component	7.9993	7.9973	7.9914	7.9967	7.994339	7.9974
G component	7.9993	7.9968	7.9915	7.9967	7.994273	7.9970
B component	7.9994	7.9975	7.9916	7.9972	7.994202	7.9971
Average	**7.9993**	**7.9972**	**7.9915**	**7.9969**	**7.994271**	**7.9972**
Improvement	–	–	**0.0976%**	**0.0300%**	**0.0366%**	–

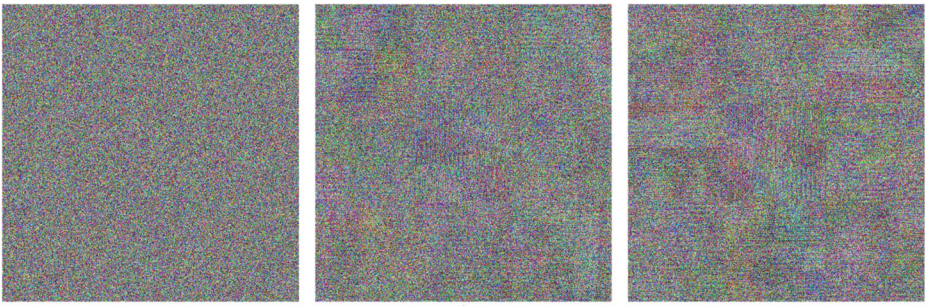

Fig. 9 Decryption with wrong system key

mentioned above, we chose to attack the cipher image with salt & pepper noise and calculate the PSNR [22] of the reconstructed image after noise attack. PSNR is computed as follows

$$PSNR = 10 \times \log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20 \times \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \tag{17}$$

where

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left\|I(i,j) - K\left(i,j\right)\right\|^2 \tag{18}$$

MAX_I denotes the maximum possible pixel value of the image,$I(i,j)$denotes the pixel values of the plain image, $K(i,j)$denotes the pixel values of the reconstructed image.

The noise of 0.01 density was firstly added to the cipher image and then decrypted according to the correct decryption algorithm. It is observed from Fig. 10 (a) that the decrypted image is substantially unaffected with the PSNR 27.8245 dB. Subsequently, the density of the noise is increased, it can be seen from (b) and (c) that while the quality of the reconstructed images is a little bit affected and PSNRs of those are decreasing from 20.8155 dB to 17.878 dB, the images subjected to noise attack can still be successfully recovered and the important information of the plain image is retained.

(2) Cropped attack analysis

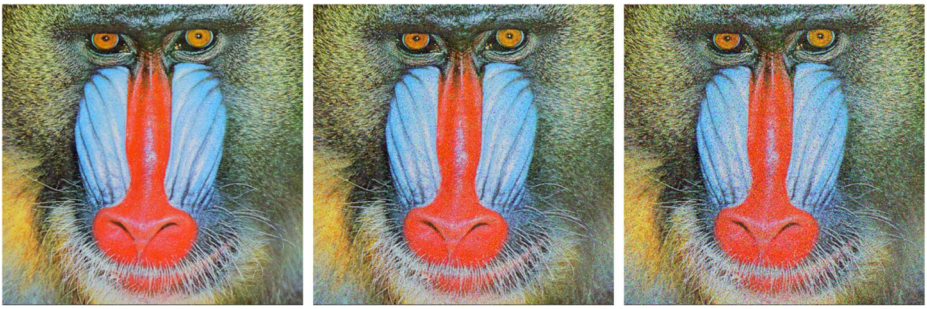

Fig. 10 The decryption results with salt & pepper noise:(**a**) 0.01 noise, PSNR = 27.8245 dB, (**b**) 0.05 noise, PSNR = 20.8155 dB, (**c**) 0.1 noise, PSNR = 17.8748 dB

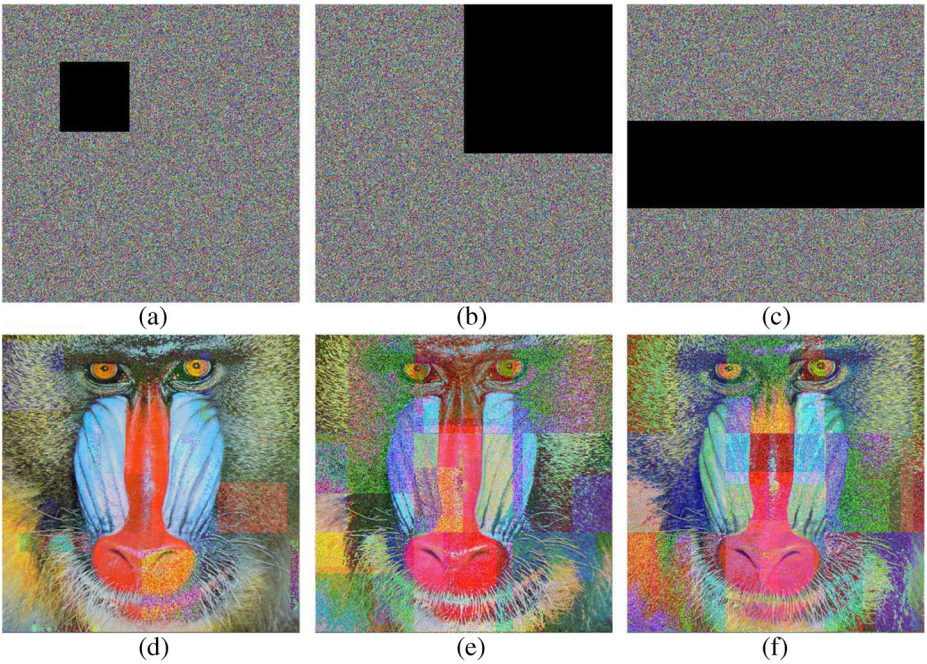

Fig. 11 Cropped attack: (**a**) 120 × 120 cropped, (**b**) 256 × 256 cropped, (**c**) 150 × 512 cropped, (**d**) Decrypted result of 120 × 120 cropped, PSNR = 20.3325 dB, (**e**) Decrypted result of 256 × 256 cropped, PSNR = 13.8291 dB, (**f**) Decrypted result of 150 × 512 cropped, PSNR = 13.1984 dB

In the process of transmission and storage of digital images, data loss occurs sometimes. We chose different degrees of data loss to test the degree of recovery of the encrypted image. As can be seen from the results in Fig. 11, when the size of the missing block of the encrypted image is only 120 × 120, it can be reconstructed according to the correct decryption algorithm with the PSNR 20.3325 dB; when the size of the missing block of the encrypted image is increased by 150 × 512, about 30% of the data of the cipher image is lost, the PSNR is 13.1984 dB, but the decrypted image can still be recognized. Therefore, from the above analysis, the algorithm can resist cropped attack.

Table 7 shows the MSE and PSNR of the reconstructed images for the encrypted images against noise attacks and cropped attacks. Based on the analysis of the noise attack and the cropped attack, it can be concluded that the algorithm proposed in this paper has strong robustness.

Table 7 MSE and PSNR values against noise attack and cropped attack

Noise attack	0.01 dB	0.05 dB	0.1 dB
MSE	107.3080	538.9267	1060.7
PSNR	27.8245	20.8155	17.8748
Cropped attack	120 × 120	256 × 256	150 × 512
MSE	602.3302	2692.6	3113.5
PSNR	20.3325	13.8291	13.1984

Table 8 Speed performance

Images	Encryption process/s	Decryption process/s
Baboon	21.044414	19.493248
Lena	20.915818	18.802360
Tiffany	21.011208	18.875521
Peppers	21.080197	18.895497
Splash	20.991757	18.999718
Airplane	20.995348	19.201820
Sailboat	21.268163	18.856172
House	20.980250	19.153423
SanDiego	21.071084	19.270790

4.6 Speed performance

In order to test the time taken by the algorithm mentioned above during the encryption and decryption phases, we run the program in a Lenovo computer with 1.90 GHz AMD A8-4500 M CPU and 4.0 GB RAM on Windows 7 system, and the execution times are listed in Table 8. When encrypting different RGB images with the size of 512×512, the encryption program and the decryption program in the computer run not fast, and the main reason is that it takes a lot of time in the process of cross-shifting. This may be a breakthrough point for future work.

5 Conclusion

In this paper, based on CRC and the nine palace map, we have proposed an algorithm of the chaos encryption for the color image. The blocks of the plain image input are scrambled, and each component is converted and decomposed into eight binary vectors. Then, based on the elements of the binary vector, the CRC code is obtained, which is used as shift step-length to change the order of elements of the binary vector. Subsequently, the operation of bitwise XOR is applied to the diffuse pixels. The experimental results and security analysis show that the space of the secret keys of the algorithm is very large, that the histogram of the cipher images is evenly distributed, and that the correlation of adjacent pixels of the cipher images is reduced. Further more, the average value of entropy is 7.9993, which is higher than the literature [9, 20, 28], indicating the efficiency for resisting the entropy attack. It proves that the encryption algorithm has a good effect on encryption by noise attack analysis, cropped attack and key sensitivity analysis. Good security makes the proposed algorithm suitable for the encryption of the color image. Today, Network Function Virtualization [35, 39] is a central topic. How to combine encryption technology with it is a future research direction and we will conduct more in-depth research. At the same time, we will further improve the security of the algorithm and research other encryption algorithms with higher security and faster speed.

Acknowledgements This work was supported by the National Natural Science Foundation of China (No.61502154, 61370092), Hubei Provincial Department of Education Outstanding Youth Scientific Innovation Team Support Foundation (T201410), the MOE (Ministry of Education in China) Project of Humanities and Social Sciences (17YJCZH203).

References

1. AqeelurRehman LX, Hahsmi MA et al (2018) An efficient mixed inter-intra pixels substitution at 2bits-level for image encryption technique using DNA and chaos[J]. Optik 153
2. Belazi A, El-Latif AAA, Belghith S (2016) A novel image encryption scheme based on substitution-permutation network and chaos[J]. Signal Process 128:155–170
3. Chai X (2017) An image encryption algorithm based on bit level Brownian motion and new chaotic systems[J]. Multimed Tools Appl 76(1):1159–1175
4. Chai X, Chen Y, Broyde L (2017) A novel chaos-based image encryption algorithm using DNA sequence operations[J]. Opt Lasers Eng 88:197–213
5. Chen W B, Xin Z (2006) Image encryption algorithm based on Henon chaotic system[C]// International Conference on Image Analysis and Signal Processing. IEEE 1–6
6. Chen J, Zhang Y, Qi L et al (2018) Exploiting chaos-based compressed sensing and cryptographic algorithm for image encryption and compression[J]. Opt Laser Technol 99
7. Fu C, Chen ZF, Zhao W et al (2017) A new fast color image encryption scheme using chen chaotic system[C]// Ieee/acis international conference on software engineering, artificial intelligence, NETWORKING and parallel/distributed computing. IEEE:121–126
8. Hua Z, Yi S, Zhou Y (2017) Medical image encryption using high-speed scrambling and pixel adaptive diffusion[J]. Signal Process 144
9. Kadir A, Aili M, Sattar M (2017) Color image encryption scheme using coupled hyper chaotic system with multiple impulse injections[J]. Optik – Int J Light Electron Opt 129:231–238
10. Kanafchian M, Fathi-Vajargah BA (2017) Novel image encryption scheme based on Clifford attractor and Noisy logistic map for secure transferring images in navy [J]. Int J e-Navig Maritime Econ 6:53–63
11. Kanso A, Ghebleh M (2017) An algorithm for encryption of secret images into meaningful images [J]. Opt Lasers Eng 90:196–208
12. Kumar M, Iqbal A, Kumar P (2016) A new RGB image encryption algorithm based on DNA encoding and elliptic curve Diffie–Hellman cryptography[J]. Signal Process 125(C):187–202
13. Lan R, He J, Wang S, et al. (2018) Integrated Chaotic Systems for Image Encryption [J]. Signal Processing
14. Li Y, Dai W, Ming Z et al (2016) Privacy protection for preventing data over-collection in Smart City[J]. IEEE Trans Comput 65(5):1339–1350
15. Liu H, Jin CA (2017) Novel color image encryption algorithm based on quantum Chaos sequence[J]. 3D Res 8(1):4
16. Liu H, Wang X (2010) Color image encryption based on one-time keys and robust chaotic maps[J]. Comput Math Appl 59(10):3320–3327
17. Mahesh M, Srinivasan D, Kankanala M et al (2015) Image cryptography using discrete Haar wavelet transform and Arnold cat map [C]// international conference on communications and signal processing. IEEE:1849–1855
18. Martin K, Lukac R, Plataniotis KN (2005) Efficient encryption of wavelet-based coded color images [J]. Pattern Recogn 38(7):1111–1115
19. Mollaeefar M, Sharif A, Nazari M (2015) A novel encryption scheme for colored image based on high level chaotic maps[J]. Multimed Tools Appl 76:1–23
20. Parvaz R, Zarebnia M (2017) A combination chaotic system and application in color image encryption[J]
21. Ping P, Xu F, Mao Y et al (2017) Designing permutation-substitution image encryption networks with Henon map[J]. Neurocomputing
22. Qiu H, Memmi G (2014) Fast Selective Encryption Method for Bitmaps Based on GPU Acceleration[C]// IEEE International Symposium on Multimedia. IEEE
23. Qiu H, Memmi G, Noura H (2017) An efficient secure storage scheme based on information fragmentation[C]// IEEE international conference on Cyber Security & Cloud Computing. IEEE
24. Sankpal PR, Vijaya PA (2014) Image encryption using chaotic maps: a survey[C]// international conference on Signal & Image Processing. IEEE:102–107
25. Sokouti M, Sokouti BA (2018) PRISMA-compliant systematic review and analysis on color image encryption using DNA properties[J]. Comput Sci Rev 29:14–20

26. Sosa C, Sutton B C, Huang H H (2010) PicFS: The Privacy-Enhancing Image-Based Collaborative File System[C]// IEEE International Conference on Parallel & Distributed Systems. IEEE Computer Society
27. Su Y, Tang C, Chen X et al (2017) Cascaded Fresnel holographic image encryption scheme based on a constrained optimization algorithm and Henon map[J]. Optics & Lasers in. Engineering 88:20–27
28. Teng L, Wang X, Meng J (2017) A chaotic color image encryption using integrated bit-level permutation[J]. Multimed Tools Appl 77(10):1–14
29. Wang X, Zhang HL (2015) A color image encryption with heterogeneous bit-permutation and correlated chaos[J]. Opt Commun 342:51–60
30. Ye G, Huang X (2017) An efficient symmetric image encryption algorithm based on an intertwining logistic map[M]. Elsevier Science Publishers B. V
31. Ye C, Xiong Z, Ding Y et al (2014) Joint fingerprinting and encryption in hybrid domains for multimedia sharing in social networks[J]. J Vis Lang Comput 25(6):658–666
32. Ye C, Xiong Z, Ding Y et al (2016) Joint fingerprinting and encryption in the DWT domain for secure M2M communication[J]. Int J Sec Appl 10(1):125–138
33. Zhang X, Wang X (2017) Multiple-image encryption algorithm based on mixed image element and chaos [J]. Optics & Lasers in. Engineering 92:6–16
34. Zhang YQ, Wang XY, Liu J et al (2016) An image encryption scheme based on the MLNCML system using DNA sequences[J]. Opt Lasers Eng 82:95–103
35. Zhang Y, Qiu M, Tsai CW et al (2017) Health-CPS: healthcare cyber-physical system assisted by cloud and big data[J]. IEEE Syst J 11(1):88–95
36. Zhou N, Chen W, Yan X et al (2018) Bit-level quantum color image encryption scheme with quantum cross-exchange operation and hyper-chaotic system[J]. Quantum Inf Process 17(6):137
37. Zhu L, Li W, Liao L et al (2006) A novel algorithm for scrambling digital image based on cat chaotic mapping[C]// international conference on intelligent information hiding and multimedia. IEEE Comput Soc: 601–604
38. Wu Y, Hu F, Min G et al(2017) Big Data and Computational Intelligence in Networking[M].Taylor & Francis/CRC, Boca Raton
39. Cheng X, Wu Y, Min G et al (2018) Network Function Virtualization in Dynamic Networks: A Stochastic Perspective[J]. IEEE J Sel Areas Commun 36(10):2218–2232
40. Ma Y, Wu Y, Ge J et al (2018) An Architecture for Accountable Anonymous Access in the Internet-of-Things Network[J]. IEEE Access 6:14451–14461
41. Huang C, Min G, Wu Y et al (2017) Time Series Anomaly Detection for Trustworthy Services in Cloud Computing Systems[J]. IEEE Transactions on Big Data (99):1–1

Zenggang Xiong received the Ph.D. degree from the University of Science and Technology Beijing, Beijing, China,in 2009. He is currently a Professor with the School of Computer and Information Science and Master Tutor, Hubei Engineering University, Xiaogan, China. His main research interests include cloud computing, big data, pattern recognition, and computer vision.

Yuan Wu received the Bachelor's degree from the Hubei Engineering University, Xiaogan City, China,in 2016. She is currently a graduate student at Hubei University. Her main research interests include cloud computing, big data, pattern recognition, and computer vision.

Conghuan Ye , received the B.S. and M.S. degree in computer science from Hubei Normal University, Hubei, China, in 2002, and University of Electronic Science and Technology of China, Chengdu, Sichuan, China, in 2005, respectively. Now, his research interests include digital fingerprinting, digital right management, complex network, and cloud computing. Dr. Ye received the scholarship from UESTC from 2003 to 2004.Dr. Ye has co-authored over 50 publications including book chapters, journal and conference papers. He received the Ph.D. degree in computer science and technology, Huazhong University of Science and Technology (HUST) in 2013, Wuhan, Hubei, China. Since 2013, he has been an associate professor with the college of computer science and technology, HBEU.

Xuemin Zhang received the the Bachelor degree in computer science from Hubei Normal University, China, in 2001, and the MA degree in computer science from Wuhan University of Technology, China, in 2009. She is now an professor in Hubei Engineering University. Her research interests are in the areas of Cloud computing, distributed systems, Service Computing. She is a member of the IEEE and the ACM.

Fang Xu received M.S. and Ph.D. degrees from Wuhan University, Wuhan, China, in 2009 and 2016, respectively. He is an Associate Professor in the School of Computer and Information Science, Hubei Engineering University, Xiaogan, China. His research interests include social computing, wireless mobile networks, and context aware computing. Dr. Xu is a member of the IEEE Computer Society, and a member of Association for Computing Machinery (ACM).