A Fast Image Encryption Algorithm Based on Self-confusion Method

Yong ZHANG*, Ai-guo CHEN, Ben CHEN and Han-cheng LIAO

School of Software and Internet-of-Things Engineering, Jiangxi University of Finance and Economics, Nanchang, 330013, P.R. China

*Corresponding author

Keywords: Image encryption, Self-confusion, Diffusion, Chaos.

Abstract. A novel image encryption algorithm based on self-confusion was proposed in this paper. The image encryption process is composed of covering operation, self-confusion and diffusion. The chaotic system was employed to generate one pseudo-random matrix and two pseudo-random sequences for image encryption. The matrix is used to cover the image, and the sequences are used to diffuse the image. The image is shuffled in the self-confusion by its own data, making the proposed image encryption algorithm be plaintext-related. The simulation results show that the proposed scheme has the merits of fast speed and good security, and can be a candidate for image information security.

Introduction

In recent years, with the development of computer network and wireless communication technology, image encryption has become a hotspot in the field of information security [1-4]. The image encryption is the most effective means to protect the security of image information transmitted and stored on the internet.

However, some image cryptosystems based on chaotic systems were proved to be unsecure [5-8]. Those image cryptosystems having been cracked used the plaintext-unrelated encryption algorithms, failing to resist the known/chosen plaintext attacks. Therefore, a good image encryption scheme must be plaintext-related, i.e. the equivalent secret keys of the scheme are not only related to the secret key, but also to the plain images [9-11].

Recently, the plaintext-related confusion algorithms were deeply studied. Since the confusion process only changes the position of each pixel, the sum or information entropy of all the pixels in the image are invariants in the confusion process. These invariants can be used to select the confusion modes [12-13]. In this paper, a new plaintext-related image confusion algorithm was proposed, which was called self-confusion because it only related to its own data.

Image Encryption Scheme

The proposed image encryption scheme is shown in Fig. 1. As can be seen from Fig. 1, the proposed image encryption system includes two parts: the pseudo-random number generator based on chaotic system and the encryption process composed of covering, confusion and diffusion. Here, the Lorenz system is chosen as the chaotic system of pseudo-random generator.

![Figure 1. The proposed image cryptosystem.](image-url)
The Lorenz system is shown as follows.

\[
\begin{align*}
    x' &= a(y - x) \\
    y' &= cx - xz - y \\
    z' &= xy - bz
\end{align*}
\]  

where, \(a=10\), \(b=8/3\), \(c=28\). The size of discretization step used in simulation test is \(dt=0.02\).

Now, denote the plain image as \(P\) of size \(M\times N\), ciphered image as \(C\) of size \(M\times N\), and secret key as \(K\) of length 240 bits.

### Pseudo-Random Sequence Generator

The structure of the key \(K\) is shown in Fig. 2.

![Figure 2. The structure of secret key K.](image)

According to Fig. 2, let

\[
\begin{align*}
    x_{01} &= -19.6 + 40.7 \cdot \frac{x_{01}}{2^4} \\
    y_{01} &= -27.2 + 57.1 \cdot \frac{y_{01}}{2^4} \\
    z_{01} &= 0.8 + 50.2 \cdot \frac{z_{01}}{2^4}
\end{align*}
\]

(2)

Then, take \(x_{01}, y_{01}\) and \(z_{01}\) as the initial states of Eq. 1. After iterating Eq. 1 for \(n_{01}\) times, the current states of Eq. 1 are recorded as \(x_{01}, y_{01}\) and \(z_{01}\). Now, let

\[
\begin{align*}
    x_{0B} &= (0.5 + \frac{0.3}{255}) x_{0A} + (0.5 - \frac{0.3}{255}) x_{02} \\
    y_{0B} &= (0.5 + \frac{0.3}{255}) y_{0A} + (0.5 - \frac{0.3}{255}) y_{02} \\
    z_{0B} &= (0.5 + \frac{0.3}{255}) z_{0A} + (0.5 - \frac{0.3}{255}) z_{02}
\end{align*}
\]

(3)

Take \(x_{0B}, y_{0B}\) and \(z_{0B}\) as the initial states of Eq. 1. After iterating Eq. 1 for \(n_{02}\) times, the current states of Eq. 1 are recorded as \(x_{0B}, y_{0B}\) and \(z_{0B}\). Then, let

\[
\begin{align*}
    x_{0D} &= (0.5 + \frac{0.3}{255}) x_{0C} + (0.5 - \frac{0.3}{255}) x_{03} \\
    y_{0D} &= (0.5 + \frac{0.3}{255}) y_{0C} + (0.5 - \frac{0.3}{255}) y_{03} \\
    z_{0D} &= (0.5 + \frac{0.3}{255}) z_{0C} + (0.5 - \frac{0.3}{255}) z_{03}
\end{align*}
\]

(4)

Now, take \(x_{0D}, y_{0D}\) and \(z_{0D}\) as the initial states of Eq. 1. After iterating Eq. 1 for \(n_{03}\) times, continue to iterate it for \(MN\) times to obtain three state sequences of length \(MN\), denoted by \(\{x_i\}, \{y_i\}\) and \(\{z_i\}\), \(i=0,1,2,\ldots, MN-1\). Get a pseudo-random matrix \(D\) and two pseudo-random vectors \(E\) and \(F\) from the three state sequences as follows:

\[
\begin{align*}
    D(k,l) &= floor \left( 10^8 \cdot (x_{kN+l} - floor(x_{kN+l})) \right) \mod 256 \\
    E(i) &= floor \left( 10^8 \cdot (y_i - floor(y_i)) \right) \mod 256 \\
    F(i) &= floor \left( 10^8 \cdot (z_i - floor(z_i)) \right) \mod 256
\end{align*}
\]

(5)

where, \(k=0,1,2,\ldots, M-1, i=0,1,2,\ldots, N-1, i=0,1,2,\ldots, MN-1\).

### Encryption Process

In Fig. 1, the encryption process includes the covering, confusion and diffusion operations. The inputs of covering are \(P\) and \(D\), and its output is \(A\). The covering algorithm is as follows:

\[
A(i,j) = P(i,j) \oplus D(i,j), i = 0,1,\ldots, M-1, j = 0,1,\ldots, N-1
\]

(6)

The confusion algorithm is in Section 2.3. In Fig. 1, the inputs of diffusion-I are \(B, E\) and \(F\), and its output is \(Q\). The matrix \(B\) is divided into two equal-size vectors, satisfying \(B_1(i)=B(i/N, i \mod N)\), \(B_2(j-MN/2)=B(j/N, j \mod N)\), \(i=0,1,\ldots, MN/2-1, j=MN/2, MN/2+1,\ldots, MN-1\). Also, the matrix \(Q\) is
equally divided into two vectors, i.e. \( Q_1(i) = Q(iN, i \mod N) \), \( Q_2(j-MN/2) = Q(jN, j \mod N) \), \( i=0,1,\ldots, MN/2-1 \), \( j=MN/2,MN/2+1,\ldots,MN-1 \). Then, let

\[
\begin{align*}
Q_1(0) &= (B_1(0) + E(0)) \mod 256 \\
Q_1(i) &= (Q_1(i-1) + B_1(i) + E(i)) \mod 256 \\
Q_2(\frac{MN}{2} - 1) &= (B_2(\frac{MN}{2} - 1) + F(MN - 1)) \mod 256 \\
Q_2(j) &= (Q_2(j+1) + B_2(j) + F(j + \frac{MN}{2})) \mod 256
\end{align*}
\]  

(7)

where, \( i=1,2,\ldots,MN/2-1 \), \( j=MN/2-2,MN/2-3,\ldots,2,1,0 \). Here, Eq. 7 can be parallelly executed.

Based on the \( Q_1 \) and \( Q_2 \) in Eq. 7, diffusion-II performs the following operations.

\[
\begin{align*}
C_1(\frac{MN}{2} - 1) &= (Q_1(\frac{MN}{2} - 1) + F(\frac{MN}{2} - 1)) \mod 256 \\
C_1(i) &= (C_1(i+1) + Q_1(i) + F(i)) \mod 256 \\
C_2(0) &= (Q_2(0) + E(MN/2)) \mod 256 \\
C_2(j) &= (C_2(j-1) + Q_2(j) + E(j + \frac{MN}{2})) \mod 256
\end{align*}
\]  

(8)

where, \( i= MN/2-2,MN/2-3,\ldots,2,1,0 \), \( j=1,2,\ldots,MN/2-1 \). Here, Eq. 8 can be parallelly executed.

Combine \( C_1 \) and \( C_2 \) into the matrix \( C \) such that \( C(i,j)=C_1(iN+j) \), \( C(k,j)=C_2(kN+j) \), \( i=0,1,\ldots,M/2-1 \), \( k=M/2,M/2+1,\ldots,M-1 \), \( j=0,1,2,\ldots,N-1 \). \( C \) is the ciphered image.

**Self-Confusion Process**

The self-confusion process does not require the pseudo-random numbers. The input of self-confusion is the matrix \( A \), and its output is the matrix \( B \), as shown in Fig. 1. The self-confusion performs the following operations.

\[
B(i,j) = A \left( \left( \sum_{k=1}^{M} A(k,j) - A(i,j) \right) \mod M, \left( \sum_{k=1}^{N} A(i,k) - A(i,j) \right) \mod N \right)
\]  

(9)

where, \( i=0,1,2,\ldots,M-1 \), \( j=0,1,2,\ldots,N-1 \). If the element of \( A \) calculated by Eq. 9 is in the \( i \)-th row or in the \( j \)-th column, then the assignment operation in Eq. 9 is not performed, but \( B(i,j)=A(i,j) \).

**Decryption Process**

The decryption process is the inverse of the encryption process, as shown in Fig. 3.

In Fig. 3, the inverse algorithm of the confusion process is as follows.

\[
A(i,j) = B \left( \left( \sum_{k=1}^{M} B(k,j) - B(i,j) \right) \mod M, \left( \sum_{k=1}^{N} B(i,k) - B(i,j) \right) \mod N \right)
\]  

(10)

where, \( i=M-1,M-2,\ldots,2,1,0 \), \( j=N-1,N-2,\ldots,2,1,0 \). If the element of \( B \) calculate by Eq. 10 is in the \( i \)-th row or in the \( j \)-th column, then let \( A(i,j)=B(i,j) \).

**Simulation Results**

The computer is configured with Intel® Core™ i7-8650U, 16GB RAM, Windows 10. The tested images are Lena, Baboon and Pepper of all size 256×256, as shown in Figs. 4a-4c, respectively. Without losing generality, the secret key \( K='826AED5A4440EA974B69DE949B7E96CDC887983193D7AF5DAE605E97E9D5' \) (in hexadecimal). The ciphered images are as shown in Figs. 4d-4f. And the recovered images are as shown in Figs. 4g-4j, identical to their corresponding plain images.
Performance Analysis

Encryption and Decryption Speed

In the computer in Section 3, under the environment of MATLAB 2018a for the image of size 256×256, the encrypted time is 0.2054s, and the decrypted time is 0.1808s. So the encryption speed is about 2.5525Mbps, and the decryption speed is about 2.8998Mbps. Considering the interpretive execution of MATLAB program, the encryption/decryption speed is quite fast.

Key Space

The secret key of proposed image cryptosystem is 240 bits, so the size of key space is $2^{240} \approx 1.7669 \times 10^{72}$. Using the computer in Section 3, it takes at least $5.0648 \times 10^{63}$ years to try the keys in the half key space. So the key space is large enough to resist exhaustive attack.

Correlation Analysis

Generally, the correlation coefficients between adjacent pixels in an image are calculated by the following formula:

$$ r = \frac{\sum_{i=0}^{n-1}(x_i-E(x))(y_i-E(y))}{\sqrt{\sum_{i=0}^{n-1}(x_i-E(x))^2 \sum_{i=0}^{n-1}(y_i-E(y))^2}} $$

(11)

where, $n$ represents the number of adjacent pixels selected from the image, and $E(x)$ returns the average value of $x$.

Here, take Figs. 4a-4f as examples to calculate their correlation coefficients. Let $n=2000$, and list the calculated results in Table 1.

| Image      | Horizontal | Vertical | Diagonal | Counter-diagonal |
|------------|------------|----------|----------|------------------|
| Fig. 4a    | 0.9671     | 0.9358   | 0.9200   | 0.9371           |
| Fig. 4b    | 0.8293     | 0.8674   | 0.7679   | 0.7964           |
| Fig. 4c    | 0.9711     | 0.9621   | 0.9369   | 0.9490           |
| Fig. 4d    | 0.0408     | -0.0009  | 0.0249   | 0.0300           |
| Fig. 4e    | -0.0630    | 0.0207   | 0.0451   | 0.0245           |
| Fig. 4f    | 0.0031     | -0.0013  | 0.0068   | -0.0428          |

Table 1 shows that the correlation coefficients in each direction of plain images are close to 1, while the correlation coefficients in each direction of ciphered images tend to 0. So the ciphered images can resist the correlation analysis.

Information Entropy Analysis

The information entropy of an image is calculated by the following formula:

$$ H = -\sum_{i=0}^{255} p(i) \log_2(p(i)) $$

(12)

For the 8-bit grayscale image, the maximum value of entropy is 8 bits. Here, the values of entropy of Figs. 4a-4f are calculated and then listed in Table 2.

| Image      | Entropy |
|------------|---------|
| Fig. 4a    | 7.2971  |
| Fig. 4b    | 7.3226  |
| Fig. 4c    | 7.5383  |
| Fig. 4d    | 7.9971  |
| Fig. 4e    | 7.9975  |
| Fig. 4f    | 7.9976  |
Table 2 shows that the information entropies of ciphered images (Figs. 4d-4f) are larger than those of plain images (Figs. 4a-4c) and are close to 8. Therefore, the ciphered images approximate the noise image and have no visual information.

**Sensitivity Analysis**

The key sensitivity and plaintext sensitivity of proposed image cryptosystem are analyzed with the help of NPCR, UACI [1] and BACI [4]. The results are shown in Tables 3-4.

| Index | Lena | Baboon | Pepper | Theoretical value |
|-------|------|--------|--------|-------------------|
| NPCR  | 99.6078 | 99.6121 | 99.6107 | 99.6094          |
| UACI  | 33.4741 | 33.4708 | 33.4730 | 33.4635          |
| BACI  | 26.7700 | 26.7567 | 26.7622 | 26.7712          |

| Index | Lena | Baboon | Pepper | Theoretical value |
|-------|------|--------|--------|-------------------|
| NPCR  | 99.6105 | 99.6122 | 99.6133 | 99.6094          |
| UACI  | 33.4443 | 33.4951 | 33.4728 | 33.4635          |
| BACI  | 26.7507 | 26.7915 | 26.7763 | 26.7712          |

Tables 3-4 show that the results of sensitivity analysis are very close to their respective theoretical values, which indicate that the proposed image encryption system has both strong key sensitivity and plaintext sensitivity, and can resist the cryptanalysis method based on differential operations.

**Conclusion**

In this paper, a new fast image encryption system with 240-bit long external key was proposed. The Lorenz system is employed to generate the pseudo-random equivalent keys. Then, the covering, self-confusion and diffusion operations are employed to convert plain images into the noise-like ciphered images. The simulation results show that the proposed image cryptosystem has the characteristics of fast speed and strong security, and can be applied to practical network image information security.

**Acknowledgement**

This research was financially supported by the National Natural Science Foundation of China (Grant Nos. 61562035 and 61762043).

**References**

[1] G. Chen, Y. Mao, C.K. Chui, A symmetric image encryption scheme based on 3D chaotic cat maps, Chaos Soliton. Fract. 21 (2004) 749-761.

[2] M. Khan, Z. Asghar, A novel construction of substitution box for image encryption applications with Gingerbreadman chaotic map and S8 permutation, Neural Comput. Appl. 29 (2018) 93-999.

[3] Y. Zhang, X. Jia, The fast image encryption algorithm based on substitution and diffusion, KSII T. Internet Inf. 12 (2018) 4487-4511.

[4] Y. Zhang, The unified image encryption algorithm based on chaos and cubic S-Box, Inform. Sciences 450 (2018) 361-377.

[5] W. Wen, Security analysis of a color image encryption scheme based on skew tent map and hyper chaotic system of 6th-order CNN against chosen-plaintext attack, Multimed. Tools Appl, 75 (2016) 3553-3560.
[6] E.Y. Xie, C. Li, S. Yu, J. Lv, On the cryptanalysis of Fridrich’s chaotic image encryption scheme, Signal Process. 132 (2017) 150-154.

[7] A. Akhavan, A. Samsudin, A. Akhshani, Cryptanalysis of an image encryption algorithm based on DNA encoding, Opt. Laser Technol. 95 (2017) 94-99.

[8] M. Li, Y. Guo, J. Huang, Y. Li, Cryptanalysis of a chaotic image encryption scheme based on permutation-diffusion structure, Signal Process. Image Commun. 62 (2018) 164-172.

[9] Y. Luo, R. Zhou, J. Liu, Y. Cao, X. Ding, A parallel image encryption algorithm based on the piecewise linear chaotic map and hyper-chaotic map, Nonlinear Dyn. 93 (2018) 1165-1181.

[10] Y. Zhang, Y. Tang, A plaintext-related image encryption algorithm based on chaos, Multimed. Tools Appl. 77 (2018) 6647-6669.

[11] P. Li, K.T. Lo, Joint image compression and encryption based on order-8 alternating transforms, J. Vis. Commun. Image R. 44 (2017) 61-71.

[12] G. Ye, C. Pan, X. Huang, Z. Zhao, J. He, A chaotic image encryption algorithm based on information entropy, Int. J. Bifurc. Chaos 28 (2018) 1850010 (11 pages).

[13] X. Huang, G. Ye, An image encryption algorithm based on irregular wave representation, Multimed. Tools Appl. 77 (2018) 2611-2628.