Application of Chaos Theory in Image Encryption

Yanhong Wei*

Weinan Normal University, Shaanxi, China, 714099

*Corresponding author e-mail: shenzhou5201@126.com

Abstract. In order to protect the digital image, people have come up with many methods, digital image encryption is one of the most important means. Based on the features of chaotic mapping and the strong correlation between adjacent pixels of images, an image encryption algorithm inspired by chaos theory is proposed. In the specific operation process, chaotic mapping is used to perform chaotic encryption on the displaced image using. The image encryption algorithm is easy to operate, safe and stable, and has been confirmed by many successful examples.

Keywords: chaotic mapping, chaotic mapping, image encryption

1. Introduction
At present, image encryption mainly adopts four methods, which apply different principles separately and achieve the same purpose. The four principles are sequential permutation, secret segmentation and sharing, modern cryptography, and chaotic dynamical systems. Each principle has its own characteristics [3-4]. Among them, the image encryption technology based on replacement can be easily implemented. However, its confidentiality is not high, and it does not conform to the modern cryptosystem, i.e., the Kerckhoffs specification, while the image encryption technology based on chaotic dynamic systems [5-6] is a password encryption technology developing quickly in recent years. The encryption process is not very complicated, firstly, the image information that needs to be encrypted is encrypted as binary data stream by the predetermined coding method, and the data stream is encrypted again by chaotic signal. Chaotic mapping can overcome the characteristics of traditional optimization algorithms that are prone to falling into the local optimum during chaos optimization. In this paper, an image encryption algorithm based on the chaos theory is proposed. The algorithm is simple in thinking and easy to implement, which has a huge key space, relatively good encryption effect, strong resistance to attack, and high security.

2. Chaos mapping and chaotic sequences
Chaos is a certainty that occurs in nonlinear dynamical systems. It is similar to a stochastic process. This process is neither periodic nor convergent and has a highly sensitive alternative to the initial value. From the perspective of the time domain, the sequence obtained by the chaotic mapping is a random serial sequence with weak correlation but has better white noise-like characteristics. Hence, it can be used to generate some pseudo-random signals or pseudo-random codes. In principle, it is only

[End of document]
necessary to increase the number of times of alternation. The period of the pseudo-random code can be very long, and it is very simple to generate a long code. In this way, through the chaotic system's sensitive substitution of initial values and structural parameters, a small number of uncorrelated, random-like, yet reproducible signals can be provided. Chaos has been widely replicated in secure communications and can also be used as an encrypted sequence.

There are many influence factors in the chaotic optimization process, each of which has different effects on it, and the chaotic mapping operator is a factor. After careful comparison of several methods, we can know that the results of chaotic mapping are better. Chaotic optimization methods based on chaotic mapping have higher optimization efficiency.

Chaotic mapping is also called tent mapping, and its mathematical expression is:

\[
x_{k+1} = \begin{cases} 
2x_k, & 0 \leq x_k \leq 1/2 \\
2(1-x_k), & 1/2 \leq x_k \leq 1 
\end{cases}
\]  

(1)

According to chaotic mapping, particle \(i\) generates chaotic point order in a specific domain according to the following algorithm steps:

1. Map each dimension \(x_{ij}, \ j=1,\ldots,n\) of the position \(x_i\) where variable is located among 0 and 1 according to the formula (1). In the following formula (2), \([a_j, b_j]\) represents the domain of the \(j\)-th dimension \(x_j\)

\[
cx_{ij} = \frac{x_{ij} - a_j}{b_j - a_j} 
\]  

(2)

2. Use the above equation (1) for \(M\) iterations to generate the chaotic mapping sequence \(cx_{ij}^1, cx_{ij}^2, \ldots, cx_{ij}^M\).

3. Map the points in the chaotic sequence back to the original space according to the following formula (3),

\[
x_{ij}^\prime = a_j + cx_{ij}^\prime (b_j - a_j) 
\]  

(3)

4. Through these chaotic sequences, the chaotic sequence of \(x_i\) after chaotic mapping can be obtained:

\[
x_i^\prime = (x_{i1}, x_{i2}, \ldots, x_{in})^\prime, s = 1, \ldots, M 
\]  

(4)

3. Image encryption

A grayscale image can be expressed in the form of an \(M \times N\) matrix, with pixel values \(\in [0,255]\), and neighboring pixels have a strong correlation in the spatial domain. The method used in this paper is to use global scrambling transform to perform image transformation Encrypt, then use the chaotic sequence generated by the chaotic system to act on the original image, perform pixel position transformation, and change the arrangement order of pixels in the original image, so that the original
image becomes a chaotic image and achieve a good encryption effect.

3.1. Description of the encryption algorithm
The specific algorithm steps are:
Step 1: Input the square digital image XinImage of the file, the matrix is represented by \( I_{N \times N} \), and the image height and width are both \( N \).
Step 2: Input the natural number \( n \) and perform \( n \) Arnold transforms on the original image to get the scrambled image YZhImage,
Step 3: When you enter a variable \( a \) and an initial value \( x_0 \), and use the Tent equation to generate a sequence \( \{ h_k \mid k = 1, 2, 3, \ldots, N^2 \} \) of length \( N^2 \). To ensure chaos and security, part of the chaotic trajectory needs to be hidden, and the value of \( h_k \) can be taken from the interval of chaotic trajectory.
Step 4: Finally, the scrambled image YZhImage is calculated with the chaotic sequence \( \{ h_k \mid k = 1, 2, 3, \ldots, N^2 \} \) to obtain the final encrypted image ZOutImage. Let the pixel value in YZh Image is expressed in \( z h[i, j] \). At the meanwhile, pixel value in ZOutImage be expressed out of \([i, j]\).

3.2. Decryption and Restore
The decryption algorithm is the inverse process of the encryption algorithm, which is to restore the image protected by encryption. Firstly, the user shall have permission to obtain the encrypted chaotic sequence \( \{ h_k \mid k = 1, 2, 3, \ldots, N^2 \} \). The Arnold transforms for \( n \) times. The inverse operation is performed on Steps 4 and 2 in the encryption algorithm to restore the original image as it used to be.

4. Experimental results and analysis

4.1. Examples of digital image encryption and decryption recovery
In this paper, the encryption preservation and decryption of the image under the algorithm are discussed in detail by example. The Lena diagram is used as the original image, and the chaotic sequence is generated using chaotic mapping, and the key is \( n=150, a=0.62, x_0=0.50000 \). Since the influence of initial conditions on chaotic sequence is the most sensitive, although there is only a minor deviation of the encryption and decryption keys, the decryption key in this example is only \( x_0=0.50001 \). However, no trace of the original image can be identified in the decrypted image, and a completely wrong image is obtained.

4.2. Objective evaluation criteria for images
When the degree of information loss can be expressed as a function of the original input image and the decrypted recovered image, this measurement is an objective fidelity criterion. Commonly used objective fidelity criteria have a root mean square error [14]

Mean square error (RMSE): Take \( f(i, j) \) as an original image. After decryption, the image is restored to \( f'(i, j) \). When an error occurs in the process, the error image is \( e(i, j) = f(i, j) - f'(i, j) \),
then both the square error (MSE: Mean Squared Error) can be expressed as:

$$e_{mse} = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} e^2(i, j)$$  \hspace{1cm} (5)$$

Smaller RMSE indicates that two images are more similar.

The peak signal-to-noise ratio (PSNR) can be expressed as follows:

$$PSNR = 10 \log_{10} \left( \frac{M \cdot N \cdot 255^2}{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left[ f(i, j) - \hat{f}(i, j) \right]^2} \right)$$  \hspace{1cm} (6)$$

The unit is dB. The larger the value of PSNR, the better the image fidelity, and the more similar the two images.

The peak signal-to-noise ratio PSNR of the recovered image and the original secret image is 55.34 using the algorithm in this paper. It can be seen that the fidelity of the two images is excellent, and the two images are very similar, the RMSE is the root mean square error of 0.1901. The decryption effect is relatively good.

4.3. Anti-noise performance

Image data will inevitably encounter various noises during transmission. Figure 1 shows the results of image restoration after adding Gaussian noise with a density of 0.02% and 0.5% to the original image, Figure 2 adds 0.02% and 0.5 to the encrypted recovered image. As a result of image restoration after% salt and pepper noise, the algorithm has better resistance to noise attacks.

![Salt and pepper image](image1.png)  ![Gaussian noise image](image2.png)  ![Restore the salt and pepper noise image](image3.png)  ![Restore Gaussian Noise Image](image4.png)

**Figure 1.** Add noise to the original image and restore the image

![Salt and pepper image](image5.png)  ![Gaussian noise image](image6.png)  ![Restore the salt and pepper noise image](image7.png)  ![Restore Gaussian Noise Image](image8.png)

**Figure 2.** Add noise to the decrypted image and restore the image
4.4. Key sensitivity analysis

According to the chaos map, the particles generate a chaotic point order. When the initial iteration of the chaotic point order is $x_0 = 0.50000$, and the decrypted key becomes $0.50001$, although the value changes only $0.00001$, by observing its key sequence, the analysis results are The random sequence is 99.56% different from the original random sequence on average. Although the key has only changed slightly, the decrypted image completely deviates from the original image, and the histogram of the decrypted image is entirely different from that of the original image. (The specific effect is shown in Figure 1), which shows that the algorithm has excellent sensitivity to keys.

5. Conclusion

Chaotic image encryption, as an important technology in the field of computer image security, is being widely studied. In this paper, the author mentions two basic principles of image encryption algorithm, namely Arnold transform and chaotic mapping. The encryption algorithm of image is simple, easy to program and implement, with the advantages such as huge key space, relatively good security and noise resistance, etc., which is in line with the modern cryptosystem, i.e., Kerckhoffs hypothesis., and the image reconstruction quality is relatively high.

Acknowledgment

The Application of Cross Chaotic System in Digital Image Encryption.Scientific Research Funds of Education Department of Shaanxi Provincial Government (Grant Number: 18JK0282)

References

[1] Amnah Firdous, Aqeel ur Rehman, Malik M. Saad Missen. A highly efficient color image encryption based on linear transformation using chaos theory and SHA-2[J]. Multimedia Tools and Applications, 2019(1).
[2] Wang J, Fei L, Ou W. CNN-based color image encryption algorithm using DNA sequence operations[C]/ 2018.
[3] Cao G, Chunqiang L I. Image encryption based on combination of spactial and wavelet domain[J]. 2017.
[4] Balakrishnan Ramalingam, Dhivya Ravichandran, Arun Adhithiya Annadurai, et al. Chaos triggered image encryption - a reconfigurable security solution[J]. Multimedia Tools & Applications, 2017(10):1-24.
[5] Qin Y, Wang Z, Wang H, et al. Binary image encryption in a joint transform correlator scheme by aid of run-length encoding and QR code[J]. 2018, 103.
[6] Junxin Chen, Zhi-liang Zhu, Li-bo Zhang, et al. Exploiting self-adaptive permutation-diffusion and DNA random encoding for secure and efficient image encryption[J]. Signal Processing, 2017, 142:340-353.