Yunpeng ZHANG[1, 2], Lifu HUANG[1], Yasin Hasan KARANFIL[2, 3], Zhenzhen WANG[1]
Northwestern Polytechnical University (1), Imperial College London (2), University of Wales (3)

A New Digital Image Hiding Encryption Algorithm Based on Dual Chaotic Systems

***Abstract**. To find new algorithms with good security, the authors proposed a new encryption algorithm based on dual chaotic systems. Firstly, the block of each pixel in the carrier image is determined with the Arnold mapping. And then with the chaotic sequence generated by Lorenz mapping, the specific location of each pixel in the corresponding block is determined. Finally, the pixels are hidden into the appropriate location by the chaotic sequence generated by Lorenz mapping. Theoretical analysis and practical tests show that this algorithm has enough key space and better key sensitivity.*

Streszczenie:** Poszukując nowych algorytmów z dobrym poziomem bezpieczeństwa, autorzy zaproponowali nowy algorytm szyfrowania oparty o podwójne systemy chaotyczne. Najpierw, blok każdego piksela w obrazie nośnika określany jest przez odwzorowanie Arnolda. Następnie, określane jest właściwe położenie każdego piksela w odpowiednim bloku przy pomocy sekwencji chaotycznej generowanej przez odwzorowanie Lorenza. W końcu piksele ukrywane są w odpowiednim położeniu przy pomocy sekwencji chaotycznej generowanej przez odwzorowanie Lorenza. Teoretyczne analizy i praktyczne testy wykazują, że zastosowany algorytm ma wystarczającą przestrzeń kluczy i lepszą czułość klucza. **Nowy algorytm szyfrowania z ukryciem obrazu cyfrowego oparty o podwójne systemy chaotyczne

Keywords: Cryptography; Chaos; Image; Hiding.
Słowa kluczowe: Kryptografia, Chaos, Obraz, Ukrycie

Introduction

Currently, digital image encryption and hiding technology are two basic way to protect digital image information[1]. In order to meet higher security requirements, digital image hiding encryption algorithm based on chaotic systems became popular in recent years.

The hiding encryption technique based on chaos is still a new thing. Researchers around the world are dedicated to the study, but the results are not satisfactory[2-4], especially in the security, restore steps and high performance requirements[5], but research literatures about hiding confidential information are rarely reported[6].

The authors have researched and explored the digital image hiding encryption techniques based on chaotic systems, and have proposed a new digital image hiding encryption algorithm based on dual chaotic systems and determined the feasibility and effectiveness of this program through theoretical analysis and practical testing.

1. A New Digital Image Hiding Encryption Algorithm Based on Dual Chaotic Systems

1.1 Pretreatment of Encryption before Hiding

In this paper, Logistic chaotic system is used to make pre-encryption on the image waiting for encryption.

Assuming the size of picture O waiting for encryption is $m \times n$, the chaotic encryption algorithm is based on gray value's transform. The basic idea is as follows:

1. The image gray value matrix was written in the form of sequence in length $m \times n$ and the pixel gray scale value range from 0 to 255. The gray value of each pixel was conversed from algorithm to the eight-bit binary system to get the binary image pixel value sequences $\{a_1, a_2, \dots, a_{m\times n}\}$

2. Using Logistic chaotic map, setting the initial parameters as u, x_1 , iterating $m \times n$ times, and it may give rise to a length of $m \times n$ chaotic pseudo-random sequence $\{x_1, x_2, \dots, x_{m\times n}\}$, to multiply the sequence elements by 256 and select the integer part. Then we get the sequence $\{x_1', x_2', \dots, x_{m\times n}'\}$, ($0 \le x_i' \le 256$).

3. Conversing $\{x_1', x_2', \dots, x_{m\times n}'\}$ from decimal system to the binary system and correspond it with binary pixel values $\{a_1, a_2, \dots, a_{m\times n}\}$ .Then to make x_i' XOR a_i and you will get the sequence $\{a_1', a_2', \dots, a_{m\times n}'\}$.

Conversing $\{a_1', a_2', \dots, a_{m\times n}'\}$ to decimal system and converting it into a matrix $m \times n$ form .Then the picture with encryption after pretreatment can be obtained.

1.2 Carrier Image's Unblocking Processing

Carrier image's unblocking should be made in according to the size of encrypted image and the carrier image. Assuming the size of image O waiting for encryption is $m \times n$, the size of gray carrier image I is $M \times N$. In the design of the algorithm, it is needed to embed the pixels of image waiting for encryption in each sub-block of carrier image. So the number of sub-blocks should be $m \times n$, the size of each block is $M \times N/m \times n$. For simplicity, we only consider the algorithm and images to be encrypted when the images are square and $M \bmod m = 0$ and $N \bmod n = 0$. When we do simulation experiments, we use 128×128 Image Lena as showed in figure 1-1 as to be encrypted and use the gray scale (Figure 1-2) as the carrier image. Based on the above analysis, the encrypted image can be divided into 128×128 blocks, each block size is 4×4 .

Figure 1-1. Image Lena Waiting for Encryption

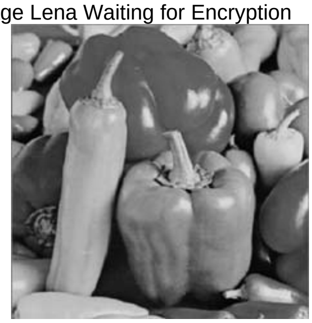


Figure 1-2. Carrier Image

1.3 Achieve the Hiding Program

1.3.1 Determine the Carrier Image Block

After a chaotic encryption pretreatment to the image waiting for encryption, unblocking the image and determining the size of the carrier image, we use the same matrix size as the image waiting for encryption to represent by using image scrambling. Arnold chaotic system is used to make each pixel in the image waiting for encryption with the corresponding pixel of a block inside carrier image.

For a square digital image, we use the discrete Arnold transform.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \bmod N \quad x, y \in \{0, 1, \ldots, N-1\}$$

Put the original coordinates (x, y) of each pixel of image (x_0, y_0) and make the block number expressed as matrix form after the carrier image being unblocked. The block number is made by the matrix row and column coordinates (x', y'), with a given coefficient matrix and the number of iterations as the decryption key. The iterative result (x_n, y_n) is used as the specific location where the original image pixel point (x, y) locates in the carrier image block. In this way, all of the carrier image pixel points were randomly and evenly scrambled into the entire block space of the carrier image.

1.3.2 To Determine the Location Where to Hide the Pixels

Lorenz system needs to use three-dimensional numerical integration to obtain real-valued chaotic sequence. In order to improve the computing speed, this paper will introduce the first-order Euler algorithm. In a given interval to determine the integration step, the integration step is set as h.

According to the system initial values x_0, y_0, z_0, the control parameters σ, γ, b ,the integration step size as h and the number of iterations n_{Lorenz}, we can get three chaotic sequences $\{lx_i\}$, $\{ly_i\}$, $\{lz_i\}$, $0 \le i < n_{Lorenz}$.To facilitate this application, it is needed to pretreat the achieved three chaotic sequence as the following:

By removing the integer part of all the real values, we get the chaotic sequence $\{lx_i'\}$, $\{ly_i'\}$, $0 \le i < n_{Lorenz}$.

Then we calculate the ratio of the size of the carrier image $I_{M\times N}$ and the image $O_{M\times N}$ waiting for encryption is t, where $t = M \times N / m \times n$. Now we get a chaotic sequence of integers $\{ly_i''\}$ $(0 \le i < n_{Lorenz})$ in the range of $-t \sim t$ and the other chaotic sequence of integers $\{lz_i''\}$ $(0 \le i < n_{Lorenz})$, range is $0 \sim t$.

An ideal chaotic sequence should have the statistical properties of uniform distribution and that the self-correlation is a function δ. Under these conditions, we can get the three chaotic sequence $\{lx_i'\}$, $\{ly_i''\}$, $\{lz_i''\}$ ($0 \le i < n_{Lorenz}$) after pretreatment. We can get the result that each treated chaotic sequence is an ideal function δ.

With the resulting chaotic sequence, we can determine the position where each pixel of the image waiting for encryption locates in the carrier image block and then make all the pixel gray value of each block of size t in the carrier image into sequence form. First to use the chaotic sequence $\{ly_i''\}$, considering a certain pixel point in the image waiting for encryption, if $\{ly_i''\}$ is a positive integer, then $\{ly_i''\}$ is the position number in which this pixel point locates in the carrier image block. If $\{ly_i''\}$ is less than or equal to 0, then we take $\{lz_k''\}$. If $\{lz_k''\}$ is 0, then we let $k = k+1$ to continue until lz_k'' is not equal to 0, at the time that pixel point's position number in the corresponding carrier image block is lz_k''.Finally we let $k = k+1$, then we can determine the origination location corresponding to the carrier image block of all the pixels in the image waiting for encryption.

1.3.3 Achieving Image Hiding

Each pixel's gray value of the carrier image and the image waiting for encryption after pretreatment is converted into the binary form. Then the 8-bit binary of each pixel's gray value of the image waiting for encryption after pretreatment is divided into three parts, first two, middle three and last three. Three binary sequences $\{O_{i1}\}$ $\{O_{i2}\}$ $\{O_{i3}\}$ $\{lo_i\}$ $(i = \{0, 1, \ldots, m\times n-1\})$were obtained.

Using the pseudo-random sequence $\{x_i\}$ ($i = \{0, 1, \ldots, m\times n-1\}$) received from the Lorenz chaotic system, $\{x_i\}$, the specific location sequence $\{lo_i\}$ ($i = \{0, 1, \ldots, m\times n-1\}$) corresponding to each pixel the of carrier image in the image waiting for encryption after pretreatment and then using following formula to hide,

$$\begin{cases} E[lo_i] = I[lo_i] + fabs(x_i) * O_{i3} \\ E[lo_i + 1] = I[(lo_i + 1) \bmod t] \wedge O_{i2} \\ E[lo_i + 2] = I[(lo_i + 2) \bmod t] \wedge O_{i1} \end{cases}$$

$$i = \{0, 1, \ldots, m\times n-1\}, \quad t = M \times N / m \times n$$

Transforming the binary sequence of the hiding results into a decimal sequence, then the Final image after hiding can be achieved.

1.4 Decryption Process

Decryption algorithm is shown as follows.

We unblock the carrier image and determine the number, using the chaotic sequence generated by Arnold chaotic systems to determine the specific block number of each pixel in the carrier image, then we use the chaotic sequence generated by the Lorenz chaotic system to determine starting position of each pixel in the corresponding block.

We use the chaotic sequence generated by Lorenz chaotic system, combining the corresponding image pixel's gray value at each point of the carrier image and the hidden image. Then we can obtain each pixel's gray value of pre-encrypted original image.

We use chaotic sequences generated by Logistic chaotic systems to pre-encrypt the gray value of each pixel's value XOR chaotic sequence ， then each pixel's gray value of original image can be obtained.

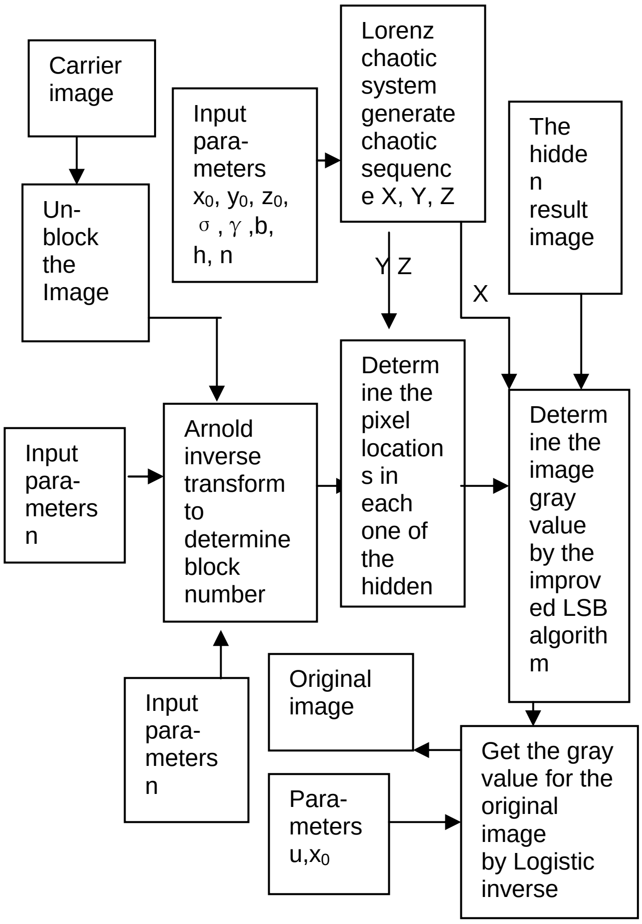
As a result, the original image can be restored.

Encryption and decryption algorithm process overall flow chart is shown Figure 1-3.

2 Experiments of Simulation and Its Analysis

The standard Lena image (figure 1-1) in 128×128 size is the image waiting for encryption. The image in size of 512×512 (Figure 1-2) is the carrier image. We set the initial value of Logistic chaotic system, $u = 3.9$, $x_0 = 0.50000000$, the effect of pre-encryption on Lena image is shown in Figure 2-1.

Assuming Arnold chaotic systems iterations $n_{Arnold}=100$, the initial value of the Lorenz chaotic system $x_0=1.1840$, $y_0=1.3627$, $z_0=1.2519$, Control parameters $\sigma=10$, $\lambda=28$, $b=8/3$, Integration step $h=0.001$, Iterations $n_{Lorenz}=128\times128$, Final result of the image hiding encryption is shown in Figure 2-2.



Carrier image
Un-block the Image
Input para-meters x0, y0, z0, σ, γ, b, h, n
Lorenz chaotic system generate chaotic sequence X, Y, Z
The hidden result image
Y Z
X
Input para-meters n
Arnold inverse transform to determine block number
Determine the pixel locations in each one of the hidden
Determine the image gray value by the improved LSB algorithm
Input para-meters n
Original image
Para-meters u,x0
Get the gray value for the original image by Logistic inverse

Figure 1-3. Encryption and Decryption Flow Chart

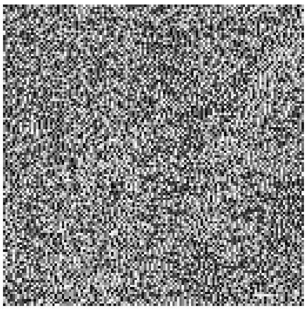


Figure 2-1. Pre-encrypted Lena image

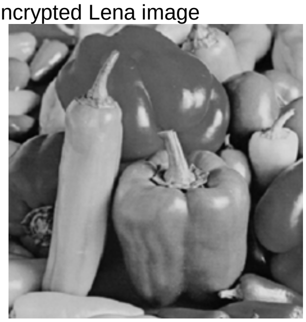


Figure 2-2. The Resulting Encrypted Image with Hidden Image Embedded

2.1 Invisible Analysis

We use the PSNR and PMSE to measure the objective fidelity between carrier images and mixed resulting images in lossless case. In this case, if PSNR is greater, the deviation of the average square root will be smaller. It shows that if the objective fidelity of the image is too high, it will have a better hiding effect. As to the carrier image $I_{M\times N}$ and resulting encrypted image $F_{M\times N}$, the formulas is:

$$PSNR=10\lg\left[\frac{M\times N\times255^2}{\sum_{i=1}^{M}\sum_{j=1}^{N}(I(i,j)-F(i,j))}\right]$$

$$PMSE=\left[\frac{1}{M\times N}\sum_{i=1}^{M}\sum_{j=1}^{N}(I(i,j)-F(i,j))^2\right]^{\frac{1}{2}}$$

M And N respectively express the width and height of the carrier image and the image waiting for encryption. $I(i,j)$ And $F(i,j)$ respectively express the gray value at the point of (i,j) the carrier image and the image waiting for encryption. The result is shown as follows:

Table 2-1. The Comparison between PSNR Values and PMSE Values

	PSNR	PMSE
The program	37.89	1.03
Algorithm [7]	22.187329	3.584322
Algorithm [8]	34.257269	5.550945

As we can see from the results that the hiding effect of the proposed algorithm in this paper is better.

2.2 Key Space Analysis

The algorithm produces the encrypted keys u, x_0 in the pre-encryption process and uses the encrypted key n_{Arnold} as loop iterations in the process of determining the carrier image block. With the increase in the number of loop iterations, the encrypted key space increases geometrically. In the application of the Lorenz chaotic system, we get the encrypted key x_0, y_0, z_0, σ, λ, b, h, n_{Lorenz} (the number of iterations). Regarding 32-bit computer, the key space of the algorithm above can be achieved to 2^{100}.

2.3 Sensitivity Analysis of Key

We use the 128×128 standard Lena image as encryption test object to test the sensitivity of the encrypted key space. The correct encrypted keys are that $u=3.9$, $x_0=0.50000000$, $n_{Arnold}=100$,the initial values of Lorenz mapping are that $x_0=1.1840$, $y_0=1.3627$, $z_0=1.2519$.The control parameters are that $\sigma=10$, $\lambda=28$, $b=8/3$,the integration step is $h=0.001$, the number of iterations is $n_{Lorenz}=128\times128$, the decrypted image taking the correct key is shown in Figure 2-3.

When the decryption key is $x_0=0.50000000$, and the other parameters are unchanged, the recovered image from the resulting hidden image is shown in Figure 2-4 (a). When the decryption key is $n_{Arnold}=101$, the decryption effect is shown in Figure 2-4(b).

We can see from the resulting decrypted image using the wrong key that this algorithm is sensitive to the decryption key.

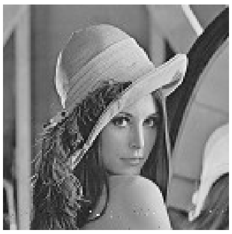
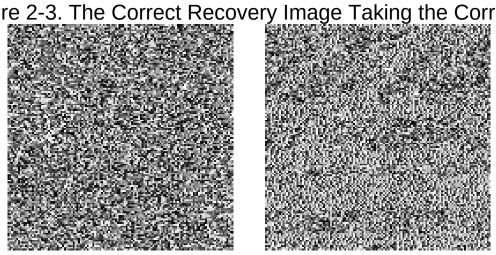


Figure 2-3. The Correct Recovery Image Taking the Correct Key



(a). $x_0 = 0.50000000$ (b). $n_{Arnold} = 101$

Figure 2-4. The Decrypted Image with Wrong Key

3 Conclusions

This paper analyzes some defects of the current digital image hiding and encryption schemes based on several chaotic systems. We propose a new digital image hiding and encryption algorithm. Through the way of theoretical analysis and practical test, we confirm its feasibility and effectiveness.

Acknowledgments: This work is supported by Science and Technology Development Project of Shaanxi Province Project (2010K06-22g), Industrial application technology research and development project of Xi'an (CXY1118(1)), Basic research fund of Northwestern Polytechnical University(GAKY100101), and R Fund of College of Software and Microelectronics of Northwestern Polytechnical University (2010R001).

REFERENCES

[1] ZHOU Zhigang, LI Su-gui. Digital Image Hiding Technology Based on Chaos System changing in parameters[J].Computer Application， 2009,11(05):2972-2976.

[2] I.J.cox, J.Kilian, T.Leighton, T.Shamoon. Secure spectrum watermarking for multimedia[J]. IEEE Trans. on Image Processing, 1997, 6(12):1673-1687.

[3] Yu Hong Heather, Peng Y. Multimedia Data Recovery Using Information Hiding[J]. IEEE Trans Image Processing, 2000, 6(4): 1344-1348.

[4] CaiW T, Turner Stephen J, Gan Boon Ping. Architecture for Information Hiding[J]. IEEE Trans Image Processing, 2001, 11(4): 67-74.

[5] XIE xie, Luo zujun, Wang Hui. An Image Information Hiding Algorithm Based On Chaotic Permutation[J].Information Security and Communications Privacy.2007.6 :187-191

[6] WANG Xian-min, GUAN Ze-qun, WU Chen-han. Information Authorized Hiding Algorithm for Remote ensign Image Based on Image Fusion[J], Journal of Remote Sensing ,2005,9(5):576-582.

[7] LI Peng, TIAN Dongping, ZHANG Nan. Digital Image Hiding Method Based on Chaotic Sequence[J]. Information Security and Communications Privacy, 2007, 06(04):222-225.

[8] Hongxing Yao, Meng Li. An approach of image hiding and encryption based on a new hyper-chaotic system[J]. International Journal *of Nonlinear Science*, 2009, 07(3) 379-384.

***Authors**: prof. Yunpeng Zhang, School of Software and Microelectronics, Northwestern Polytechnical University, Xi'an, 710072, China, E-mail: Poweryp@163.com; Lifu Hang, School of Software and Microelectronics, Northwestern Polytechnical University, E-mail: huanglifu_1989@163.com; Yasin Hasan Karanfil, Imperial College London, London, SW3 6NP, UK, E-mail:* yasinkaranfil@gmail.com*; Zhenzhen Wang, School of Software and Microelectronics, Northwestern Polytechnical University, Xi'an, 710072, China, E-mail:rjxyjs@nwpu.edu.cn.*