Image scrambling algorithm based on image block and zigzag transformation

Changjiu Pu*

Network Center, Chongqing University of Education, Chongqing 400067, Chongqing, China

Received 1 June 2014, www.cmnt.lv

Abstract

Image scrambling has wide application field in the protection of image information and secret. To achieve a satisfactory level of security, this paper introduces a color image scrambling algorithm based on image block, extended zigzag transformation and bit exchange technology. First, the algorithm converts the three-dimensional color image into two-dimensional gray image using matrix transformation according to the order of each component of image, then divides the image into blocks and completes block matching in couple, finally, the image is converted to cipher image using matrix transformation after permutation, substitution and bit exchange. Experiment simulations and theoretical analysis show that the algorithm can completely reach good scrambling effect and has the advantages of a large space of keys, high security, strong robustness and high sensitivity.

Keywords: zigzag transformation, bit exchange, image block, image scrambling

1 Introduction

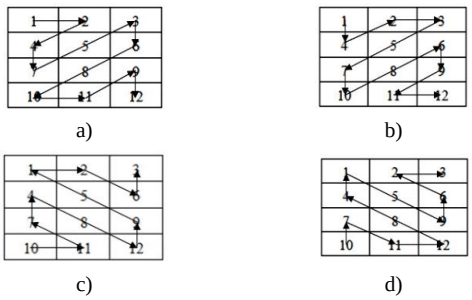
Digital multimedia is easy to illegally obtain, tampering and communication. So, how to complete the security transmission of digital image in the network, which has became a hot research in the field of information security. Image scrambling can break the correlation of pixels of image, which looks like a meaningless noise image and enhance the ability to resist malicious attacks in a certain extent. The scheme of image scrambling can be divided into three categories: scheme in space domain (including the position and color space) [1-3], scheme in frequency domain [4-6] and combination of them. A large amount of literatures have proposed many image scrambling algorithms in space domain. Literature [7] mentioned the Arnold transformation algorithm, which need take several times to attain the satisfied effect or combination with other algorithm [8]. Because the Arnold algorithm itself has a periodic [9], it was later extended to 3-dimensional Arnold transformation to improve security [10]. Literature [11] mentioned Knight's tour algorithm which has large amount of key space, higher security, but time complexity is high, but it also need take many times to attain the satisfied scrambling effect. Currently, there are many other scrambling algorithms, such as Rubik cube transformation [12], Hilbert curve transformation [13], S-box [14], Gray conversion, but the implementations of these algorithms are complex. Compared with these algorithms, the zigzag transformation has the advantages of simple realization, low time complexity and large key space. So it has been gradually adopted in image scrambling in recent years [15-16].

This paper presents a color image scrambling algorithm. The algorithm converses color image to gray image to complete pixels scrambling in space domain and color space using image block, extended zigzag scanning, matrix transformation, Henon map and bit exchange technology. The algorithm achieves 4 pixels scrambling, 2 pixels substitution, which achieves good effect of scrambling and high safety.

2 The extended zigzag transformation

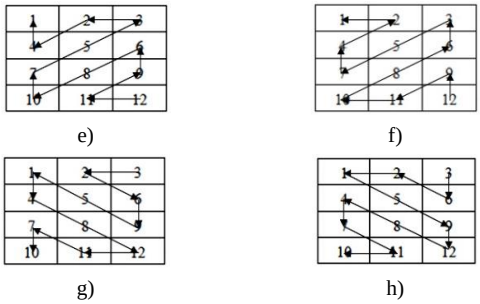
Zigzag transformation [15] is a scrambling algorithm. It starts from the upper left corner of the matrix, saves every element to a one-dimensional array in the scanning order according to the "Z" word shape, and then rearranges the one-dimensional array to a two-dimensional matrix in a certain way. The zigzag transformation begins only in the block $2^n \times 2^n$, which is also called the standard zigzag transformation.

Due to uncertainty and diversity of matrix, many scholars have studied the standard zigzag transformation to extended zigzag transformation, which can be applied to non-standard matrix. The extended zigzag transformation has 8 kinds. Figure 1 shows the scanning process of the extended zigzag transformation of 4 × 3 non square matrix.



a) b) c) d)

* ***Corresponding author's*** e-mail: pcj8880289@sina.com



e) f) g) h)

FIGURE 1 The extended zigzag transformation of 4×3 matrix

The algorithm in this paper will implement different scrambling scanning mode in different image block according to the key parameter.

In order to strengthen the security of the algorithm, the origin of conversion is from any pixel of image block, not the upper left corner of the matrix. As Figure 1a an example, assuming starting from the point (2, 2), so the cycle is the following:

$5\to3\to6\to8\to10\to11\to9\to12\to1\to2\to4\to7$

3 Algorithm

3.1 THE CONVERSION OF COLOR IMAGE TO GRAY IMAGE

In the processing of conversion, the three-dimensional color image I need convert to two-dimensional grey image I1. Assuming that the size of the plain color image is M×N, the R, G and B component of the color image respectively are I(:, :, 1), I(:, :, 2) and I(:, :, 3). Because each pixel point of color image has three pixel value, the number of pixel value of plain color image I is M×N×3. According to the knowledge of probability theory, the number of the order of R, G and B component is 3!. So the size of new grey image is 2M × 1.5N in the paper.

3.2 IMAGE BLOCK AND MATCHING

In order to achieve better effect in image scrambling, image I1 generated by the method described in section 3.1 need be divided into blocks and matched for each two block.

The block size is set $m\times n$ (m and n should be respectively less than M and $0.75N$, M can be divisible by m, $0.75N$ can be divisible by n and the number of block is even numbers), so there is a division of $K=M/m$ blocks in the row direction and there is a division of $T=N/n$ blocks in the column direction. Each block are numbered in row-major, $B(u,v)$ is numbered as $Block((u-1)\times K+v)$, and the pixel of i-th row and j-th column is denoted as $X_{B((u-1)\times K+v)}(i,j)$. So, the process of image block and matching is the following:

At first $K\times T$ different values $h\{h_1,h_2,\cdots h_{K\times T}\}$ are generated by Henon map [17], the Henon map defined by the following equation:

$$\begin{cases} x_{n+1}=y_n+1-ax_n^2 \\ y_{n+1}=bx_n \end{cases}, \tag{1}$$

where system appears strange attractor when $a=1.4$ and $b=0.3$. According to the numerical size of $h\{h_1,h_2,\cdots h_{K\times T}\}$ and Equation (2), they are converted into a positive integers sequence $h'\{h_1',h_2',\cdots h_{K\times T}'\}$ from 1.

$$h_i'=\mathrm{mod}(round(h_i\times10000000000),K\times T). \tag{2}$$

Then, matching sequence of each block is determined by the index value and the sequence value of $h'\{h_1',h_2',\cdots h_{K\times T}'\}$. The process is the following:

1) According to the first index value 1 and sequence value h_1', the first pair of 1 and h_1' respectively is saved into sequence $x'\{x_1,x_2,x_3,\cdots x_{(K\times T)/2}\}$ and $y'\{y_1,y_2,y_3,\cdots y_{(K\times T)/2}\}$, $x_1=1$, $y_1=h_1'$ and then choose the next index value and sequence value.

2) If the selected index value or sequence value is more than $(K+T)/2$, the program goes to step 5; otherwise go to step 3.

3) Is the index value present in sequence $y'\{y_1,y_2,y_3,\cdots y_{(K\times T)/2}\}$? If yes, choose the next index value and go to step 2. Otherwise, go to step 5.

4) According to the selected index value, traversals the sequence h' to find the first value which is more than index value and the value is not present in $y'\{y_1,y_2,y_3,\cdots y_{(K\times T)/2}\}$. If finding it, the program stores the index value and sequence value to sequence $x'\{x_1,x_2,x_3,\cdots x_{(K\times T)/2}\}$ and $y'\{y_1,y_2,y_3,\cdots y_{(K\times T)/2}\}$, and then chooses the next index value and sequence value and goes to step 2;

5) The end of the program.

3.3 ZIGZAG TRANSFORMATION AND BIT EXCHANGE

According to extended zigzag scanning mode of each block, matching and the exchange bit length of each matching block; the process of bit exchange is given by the following (Assuming the block p and block h is a pair of the exchange block), please see the Figure 1.

First, pixels of each block make Xor operation with the value generated by Henon map according to the order of scanning mode for the selected zigzag and Equation (3):

$$\begin{cases} List_{B(p)}'(i,j)=List_{B(p)}(i,j)\oplus Chaohenon_x(\mathrm{k}) \\ List_{B(h)}'(i,j)=List_{B(h)}(i,j)\oplus Chaohenon_y(\mathrm{k}) \end{cases}, \tag{3}$$

where, $List_{B(p)}(i,j)$ and $List_{B(h)}(i,j)$ respectively are the i-th pixel values of block p or block h according to the j-th transformation order shown in Table 1. $List_{B(p)}'(i,j)$ and $List_{B(h)}'(i,j)$ respectively are the i-th new pixel values,

which is saved into new block $List'_{B(p)}(i, j)$ and $List'_{B(h)}(i, j)$ according to the *j*-th transformation order shown in Table 1.

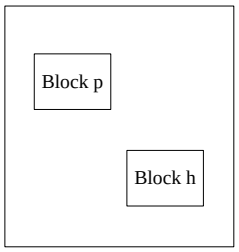


FIGURE 2 The sketch map of image block

TABLE 1 The *j*-th zigzag transformation

j	Transformation
1	Figure 1a
2	Figure 1b
3	Figure 1c
4	Figure 1d
5	Figure 1e
6	Figure 1f
7	Figure 1g
8	Figure 1h

The function of $Chaohenon_x(i)$ and $Chaohenon_y(i)$ generate sequence value to make the xor operation with the pixels of the image. The method is that it chooses several bits of them generated by the Henon map to composite the positive integer, and then uses the positive and 256 to make the complementation operation, according to Equation (4):

$$\begin{cases} Chaohenon_x(i) = mod\left(round\left(x*10000000000\right), 256\right) \\ Chaohenon_y(i) = mod\left(round\left(y*10000000000\right), 256\right) \end{cases}, \quad (4)$$

In this processing, the front part sequence value may be discarded to gain better value.

Then, implementation bit exchange according to the matching and the length of the exchange form inputted. Assuming the length of the bit exchange is *len*. So the new pixel value of block p can be obtained according to Equation (5):

$$List1'_{B(p)}(i, j) = bitxor(bitand(List'_{B(p)}(i, j), len1), bitand(List'_{B(h)}(i, j), len2)), \quad (5)$$

where, $List'_{B(p)}(i, j)$ and $List'_{B(h)}(i, j)$ respectively are the *i*-th pixel in the block p or block h according to the *j*-th transformation order. $List1'_{B(p)}(i, j)$ is the *i*-th new pixel after completing bit exchange and stored in block p according to the *j*-th transformation order. The relationship between *len*, *len*1 and *len*2 as shown in Table 2:

TABLE 2 The relationship between *len*, *len*1 and *len*2

len	*len*1	*len*2
1	128	127
2	192	63
3	224	31
4	240	15
5	248	7
6	252	3
7	254	1

For example:

$$List1'_{B(p)}(i,1) = bitxor(bitand(List'_{B(p)}(i,1), 240), bitand(List'_{B(h)}(i,2), 15)), \quad (6)$$

where $List'_{B(p)}(i,1)$ is the i-th pixel value in the block p transformed order as Figure 1a. $List'_{B(h)}(i,2)$ is the *i*-th pixel value in the block h transformed order as Figure 2b. $List1'_{B(p)}(i,1)$ is the *i*-th new pixel value in the block p composed by the left 4 bit of *i*-th pixel value of block p and the right 4 bit of i-th pixel value of block h and stored in block p transformed order as Figure 1a. So, the new generation method of block *h*:

$$List1'_{B(h)}(i, j) = bitxor(bitand(List'_{B(p)}(i, j), len2), bitand(List'_{B(h)}(i, j), len1)), \quad (7)$$

3.4 THE PROCEDURE OF THE ALGORITHM

The flow chart of the algorithm is shown as follows:

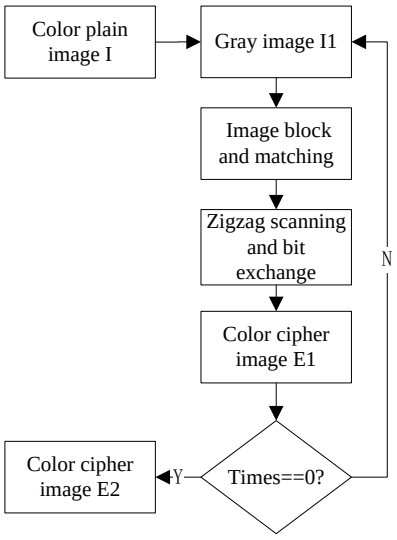
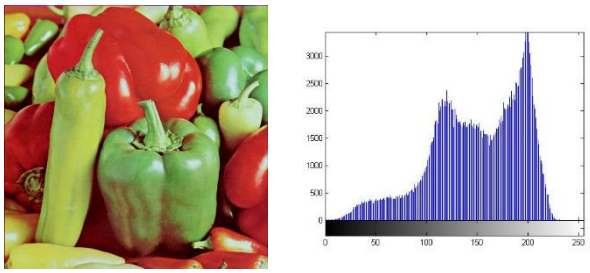


FIGURE 3 The flow chart of program

4 Experiment and analysis

In order to check the effect of scrambling, the color plain image Baboon (the size is 512 × 512) is used in the experimental (Figures 4a and 4b) is the histogram of R component of Figure 4a, others are not listed.). The platform of experimental is Windows 7 and Matlab.



a) b)

FIGURE 4 Plain image

4.1 KEY SPACE ANALYSIS

In order to analyze the key space of the algorithm, algorithm is divided into 3 stages according to described in section 3: In the process of converting to gray image, key parameters have the order of each component of the color image and the origin of each component. In the process of image block and matching, key parameters mainly lies in the size of image block, matching for each two block. In the process of zigzag transformation and bit exchange, key parameters mainly have different extended zigzag transformation of each block at different stages, the initial key of Henon map and the length of the bit exchange. So, the algorithm has a large key space.

4.2 SCRAMBLING EFFECT

Poor initial condition is used to experiment: In the process of conversing to 2D gray image, the permutation of RGB of each component is determined in column major way. Then the image is divided into blocks accord to Figure 5. And the matching of block is also determined (block 1 and block 4, block 2 and block 3).

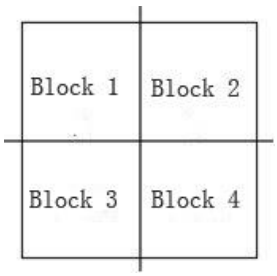


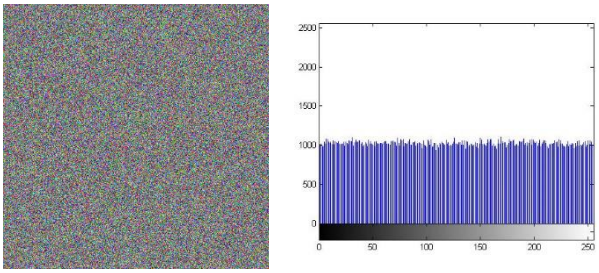
FIGURE 5 Block and number of image in the experiment

In the processing of xor operation and bit exchange, block 1 and block 2 use the extended zigzag transformation shown in Figure 1a, Figure 1a, Figure 1a and Figure 1b. Block 3 and block 4 use the extended zigzag transformation shown in Figure 1b, Figure 1b, Figure 1b and Figure 1a. The 4 bit of the pixel value is set to bit exchange in the algorithm. Meanwhile, the initial value x and y of Henon map are 0.7778889999 and 0.9998887777 and the first 99 values of the Henon map are discarded. In the processing of rearranging gray image I2 to the scrambling image E1, the details of operation is the following:

$$\begin{aligned} &LS1 = I2(M+1:1.5M,1:2N);\\ &LS2 = reshape(LS1,M,N);\\ &E1(:,:,1) = LS2;\\ &E1(:,:,2) = I2(1:M,N+1:2N);\\ &E1(:,:,3) = I2(1:M,1:N); \end{aligned} \tag{8}$$

where, LS1 and LS2 is the provisional matrix, which the size is M×N.

The scrambling result is shown in Figure 6. Comparison with Figure 4a, the distribution of pixels of Figure 6a is closely, which effectively hides the information of the plain image. So the effect of scrambling is very good. Meanwhile, the histogram of R component is shown in Figure 6b, the distributed is even and is entirely different to the plain image (Figure 4b).



a) b)

FIGURE 6 The effect of scrambling

4.3 SENSITIVITY TESTING

In the testing of sensitivity, 2 scenarios are applied to the following test cases:

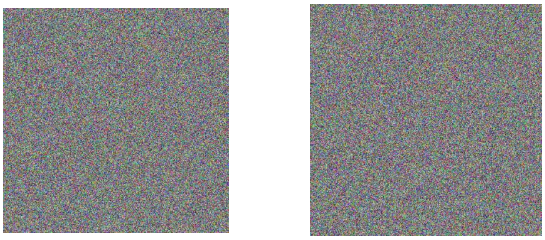
Scenario 1: The zigzag scanning mode of block 1 all use Figure 1b described in section 4.1;

Scenario 2: changing the initial key y of Henon map form 0.9998887777 to 0.9998887776.

From the Figure 7a (result of scenario 1) and Figure 7b (result of scenario 2), the result are also entirely different. The histogram of them are smoothed (histogram is slightly).

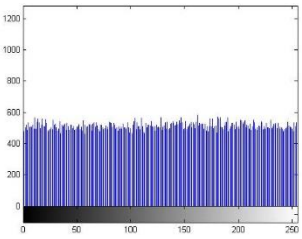
In order to check their scrambling effect on the image, the following measure is usually used: number of pixels change rate (NPCR) [18].

The NPCR of R, G and B components in Figures 6a and 7a is 0.466110，0.232933，0.689735. The NPCR of R, G and B components in Figures 6a and 7b is 0.996208, 0.996231, 0.995922. This is mainly because that the scenario 1 only changes the zigzag scanning mode in part region of image, and the scenario 2 change all pixels. Although the scheme of scenario 1 is only change some pixel values of image, the effect is also perfect. Certainly, the effect will be much better if increasing the number of blocks or execution times of algorithm. From the respective NPCR of them, the pixels values are almost entirely different, which embodied the key sensitivity is very good, and also reached the purpose of confusion and diffusion in cryptography.



a) b)

FIGURE 7 The effect of scrambling of different scenario

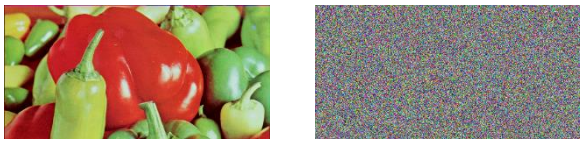


c)

FIGURE 8 The scrambling effect of shear image

4.4 THE SCRAMBLING EFFECT OF NON-STANDARD IMAGE

In order to check the compatibility of scrambling algorithm, the sheared image (Figure 8a), the size is 512×256) is select to test. Accord to the initial keys described in section 4.1, the scrambling effect and histogram are shown in Figure 8b and Figure 8c. The scrambling scheme is also very good in the different width and height (the size is non standard).



a) b)

5 Conclusion

In this paper, a novel color image scrambling algorithm based on extended zigzag scanning, bit exchange and matrix transformation has been introduced in detail. The algorithm has achieved satisfactory scrambling effect, and has the advantages of easy implementation, high security and high sensitivity. In order to further increase the algorithm's security and convenience, the next step of the research work is as follows: different blocks with different initial values, the introduction of high dimension chaotic map and hash sequence to bring more transformation algorithm and reduce the key input.

Acknowledgments

The work was supported by the Chongqing University of Education (No. JG20132209).

References

[1] Zhang X, Liu F, Jiao L 2013 An image encryption arithmetic based on chaotic sequences *Journal of Image and Graphics* **8**(4) 374-8
[2] Ye G 2010 Image scrambling encryption algorithm of pixel bit based on chaos map *Pattern Recognition Letters* **31**(5) 347-54
[3] Liu G, Jiang T, Jiang W 2013 Color Image scrambling based on Zigzag Transformation *Computer Engineering & Science* **35**(5) 106-11
[4] Ansari S, Gupta N, Agrawal S 2012 An Image Encryption Approach Using Chaotic Map in Frequency Domain *International Journal of Emerging Technology and Advanced Engineering* **2**(8) 287-91
[5] Liu Z, Zhang Y, Zhao J, Ahmad M A, Liu S 2011 Optical multi-image encryption based on frequency shift *International Journal for Light and Electron Optics* **122**(11) 1010-3
[6] Zhang Y, Zhou M, Huang W 2009 Frequency-domain digital image watermarking algorithm based on image-scrambling *Computer Technology and Development* **19**(3) 49-51
[7] Ding W, Yan W, Qi D 2001 Digital Image Scrambling Technology Based on Arnold Transformation *Journal of Computer Aided Design & Computer Graphics* **13**(4) 338-41
[8] Xiang Y 2013 An improved hash encryption algorithm based on Arnold mapping *Journal of Chongqing normal university (Natural science)* **30**(4) 103-8 *(in Chinese)*
[9] Sun X, Zhang R 2008 A New Algorithm for Calculating Period of Arnold Transformation *Computer Technology and Development* **18**(11) 66-8
[10] Khade P N, Narnaware M 2012 Practical Approaches for Image Encryption/Scrambling Using 3D Arnolds Cat Map *Advances in Communication, Network, and Computing* **108** 398-404
[11] Paris L 2004 Heuristic strategies for the knight tour problem *Proceedings of IC-AI 2004 & MLMTA 2004* Athens Greece 1121-5
[12] Chen Q, Liao X, Chen Y (2005) Modified image encryption based on chaotic sequences and Rubik's cube transformation *Computer Engineering and Applic*ations **22** 138-9
[13] Lin X, Cai L 2004 Scrambling Research of Digital Image Based on Hilbert Curve *Chinese Journal of Stereology and Image Analysis* **9**(4) 224-7 *(in Chinese)*
[14] Sui X, Luo H 2004 Digital image scrambling based on S-box *Journal of Image and Graphics* **9**(10) 1223-7
[15] Dong H, Lu P, Ma X 2011 Image scrambling algorithm based on mixed chaotic systems and extended Zigzag transformation. *Computer Engineering and Design* **32**(4) 1241-5
[16] Lu P, Dong H, Ma X 2012 An image scrambling algorithm based on extended Zigzag transformation and bit exchange *Computer Applications and Software. Computer Applications and Software* **29**(10) 310-3
[17] Henon M 1976 A two-dimensional mapping with a strange attractor. *Communications in Mathematical Physics* **50**(1) 69-77
[18] Tong X, Cui M 2008 Image encryption with compound chaotic sequence cipher shifting dynamically *Image and Vision Computing* **26**(6) 843-50

Authors

Changjiu Pu, September 1980, China.

Current position, grades: researcher at Chongqing University of Education, China.
University studies: master's degree in computer application technology from Southwest University, China in 2009.
Scientific interests: information security and robot control and motion planning.