A THREE-DIMENSIONAL IMAGE ENCRYPTION APPROACH BASED ON CHAOTIC MAPS

Feng HUANG[1,2 *], Xiongzhi LI[1], Shixiong BAI[1], Hui HU[1]

Some chaotic maps were used in image encryption. But most of them only processed square image and couldn't change the pixels' value. The paper proposes a rectangle image encryption approach which can change the statistical characteristics of images with pixels permutation. Firstly, it introduces new chaotic maps utilizing the stretch-and-fold mechanism. An image is divided into three parts, each pixel is inserted between two adjacent pixels in next column. Repeating the process and all pixels are stretched and joined to a pixel line. Then the line is folded to a cipher image. Secondly, it split a rectangle image into eight layers by pixel's value. The layers can be stretched and joined to an eight times bigger new image. It uses the new chaotic maps to encrypt the new image pixels. At last, the cipher image is combined to an image with the size as original image. The process quickly changes the pixel' values and positions. Analyses prove it has good confusion and diffusion characteristics.

Keywords: image encryption, chaotic map, confusion, image security

1. Introduction

Today people are accustomed to surfing the Internet every day. People upload personal images or download favorite images. It can be said that Internet images have become a part of life. Thus, image security become more important than ever before. As we know some traditional encryption algorithms, DES, RSA etc., can be used in image encryption. To some extent, the algorithms can protect images as well as text information. However, in fact, images have many inherent characteristics different from text: strong correlation in adjacent pixels, big bulky data capacity redundancy and so on. Image encryption using the classic encryption algorithms may bring some security risks [1].

Butterfly effect makes chaos systems well known. In recent years, researchers proposed some image encryption technologies based on chaos systems [2-5]. In fact, some characteristics of chaos system can connect with the properties of encryption technologies, act as sensitive on initial conditions and so on [6].

[1] College of Electrical & Information Engineering, Hunan Institute of Engineering, Xiangtan, Hunan, China
[2] The Cooperative Innovation Center of Wind Power Equipment and Energy Conversion, Hunan Institute of Engineering, Xiangtan, Hunan, China
*Corresponding Author: e-mail: hf7825@qq.com

The adjacent pixels in an image are strongly correlated. Permutation can shuffle the positions of image pixels. Some 2D chaotic maps, baker map, cat map and tent map, are used for square image permutation. In 1998 a classic symmetric image encryption approach was proposed by Fridrich who introduced a general architecture for chaos-based image cipher [7]. In [8] a square plain image was permuted based on a sequence of pseudo random number generated by 2D baker map. In [9], a new simple 2D chaotic map was proposed for image encryption, which used basic stretch-and-fold mechanism of chaos. Compared with baker map, the new map has a simpler formulation, larger key space. Obviously square images were required in all the papers.

The permutation-only ciphers are vulnerable to plaintext attack and statistical cryptanalysis. For instance, the key is not so sensitive that a little change of key is not able to affect the whole process of encryption or decryption in [7]. To overcome the inherent security weaknesses in permutation-only ciphers, people extended some 2D chaotic maps to 3D or hyper chaotic maps which could change the pixels' value [10-13]. In [10,11], two image encryption approaches were proposed based on 3D baker map and 3D cat map which used logistic or Chen's chaotic system to confuse the relationship between a cipher image and a plain image. Encryption techniques employ chaotic scrambling together with an XOR operation in [12]. In [13] was presented a chaos-based image encryption method using logistic map and hyperchaotic system for confusion and diffusion. In [14], the image encryption combined confusion and diffusion into one stage. Four logistic maps were used to generate the encryption key stream. A fast-chaotic image encryption scheme with block permutation and block diffusion was proposed in [15] which included a plain image-related swapping block permutation strategy.

A bit-level image cryptosystem was introduced in [16]. The binary plain image was scrambled globally by a 2D hyper-chaos system. The images could be extended to 3D ones by the pixels' value. Act as each pixel value could be expressed as a binary number, the corresponding bits of the binary number would form a rectangular image layer. The chaotic maps in [17] were unable to handle rectangular image. So it was necessary to design improve chaotic maps which could permutated every rectangular layer. Because the pixel bits of each layer were almost random distribution, the process changed the statistical characteristics of images finally.

2. The principle of the chaotic maps

Suppose that a rectangular image consisted of $N \times M$ ($M > N$) pixels with L gray levels. The chaotic map was realized by the stretch-and-fold mechanism. For left-map permutation in Fig. 1(a), firstly, the plain image was divided into three

parts: two right-angled triangles (D, E) and a rectangle (C). In a part, from the direction as shown in Fig.1(a), each pixel of first column were inserted respectively between two adjacent pixels in second column, each pixel of third column were inserted respectively between two adjacent pixels in 4th column and so on. Repeating the process, the pixels could be stretched to a line. Then the line was folded over to a new image whose size as the plain image. For right-map permutation in Fig. 1(b), the plain image also was divided into two right-angled triangles (G, H) and a rectangle (H). Pixels was inserted respectively between two adjacent pixels from different directions.

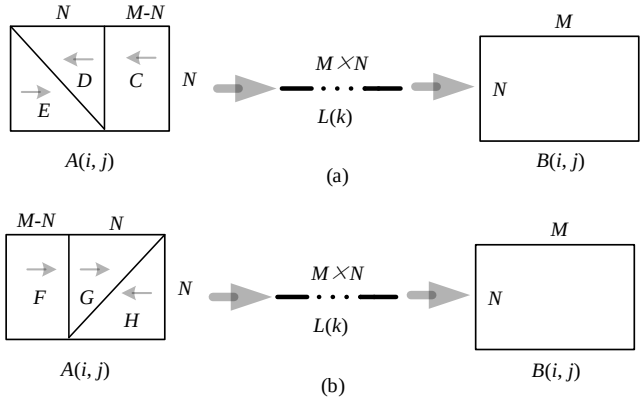

Fig. 1. The principle of the maps. (a) left-map permutation; (b) right-map permutation

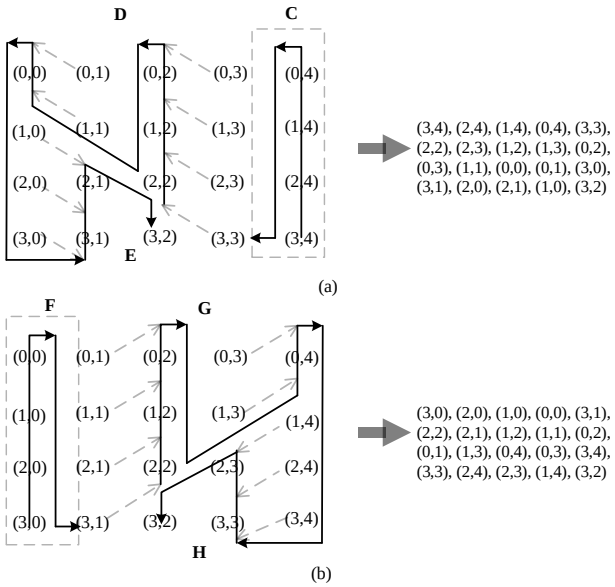

Fig. 2. The process of the map. (a) left-map permutation; (b) right-map permutation

Two examples were given here. The process of the left-map permutation was shown in Fig. 2(a). The image with 4×5 pixels, that was N=4, M=5. From the principle of the left-map permutation, the plain image firstly was divided into two right-angled triangles (D, E) and a rectangle(C). Then every pixel in each column of the three parts was inserted respectively between two adjacent pixels in next column from right to left. Because there was only a column in rectangle(C), the pixels didn't insert. In right-angled triangles(D), the pixel (3,3) was placed in front of pixel (2,2), pixel (2,3) was inserted between pixel (2,2) and pixel (1,2), pixel (1,3) was inserted between pixel (1,2) and pixel (0,2) and so on. Repeating the process for the rest of the image, the pixels of plain image joined to a line of pixels: (3,4), (2,4), (1,4), (0,4), (3,3), (2,2), (2,3), (1,2), (1,3) and so on. Lastly it was mapped to a new rectangular image whose size as same as the plain image. The right-map was symmetric with the left-map shown in Fig. 2(b).

Supposing the size of a rectangular image was $N \times M$, where N, M were integers and $M>N$. $A(i, j)$ was the matrix of the plain image, in which each element corresponds to a gray-level value of the pixel (i, j), i=0, …, N-1, j=0, …, M-1; $L(k)$ was a line of pixels mapped from A.

2.1 The left-map permutation

The left-map was shown in Fig.1 (a). The algorithm of the image was described with the following formula:

$$\boldsymbol{L}[(M-N)\times N+\frac{N^2+N+(2N-j-1)\times j}{2}+2(N-i-1)]=\boldsymbol{A}(i, j) \tag{1}$$

while $j<i$, j is even number.

$$\boldsymbol{L}[(M-N)\times N+\frac{N^2+N+(2N-j)\times (j-1)}{2}+2(N-i)-1]=\boldsymbol{A}(i, j) \tag{2}$$

while $j<i$, j is odd number.

$$\boldsymbol{L}[(M-N)\times N+\frac{(N+j+2)(N-j-1)}{2}+2(j-i)]=\boldsymbol{A}(i, j) \tag{3}$$

while $N>j\geq i$, N-j is odd number.

$$\boldsymbol{L}[(M-N)\times N+\frac{(N+j+3)(N-j-2)}{2}+2(j-i)+1]=\boldsymbol{A}(i, j) \tag{4}$$

while $N>j\geq i$, N-j is even number.

$$\boldsymbol{L}[(M-j-1)\times N+2(N-i)-1]=\boldsymbol{A}(i, j) \tag{5}$$

while $j\geq N$, M-j is odd number.

Among (5), when *M-N is odd number* and $j=N$:

$$\boldsymbol{L}[(M-N-1)\times N+N-i-1]=\boldsymbol{A}(i, j) \tag{6}$$

$$\boldsymbol{L}[(M-j-1)\times N+2(N-i)-2]=\boldsymbol{A}(i, j) \tag{7}$$

while $j\geq N$, M-j is even number.

2.2 The right-map permutation

The right-map was shown in Fig.1 (b). The algorithm of the image was described with the following formula:

$$\boldsymbol{L}[(M-N)\times N+\frac{N^2+N+(2N-j-1)\times j}{2}+2(N-i-1)]=\boldsymbol{A}(i,M-1-j) \quad (8)$$

while $j<i$, j is even number.

$$\boldsymbol{L}[(M-N)\times N+\frac{N^2+N+(2N-j)\times(j-1)}{2}+2(N-i)-1]=\boldsymbol{A}(i,M-1-j) \quad (9)$$

while $j<i$, j is odd number.

$$\boldsymbol{L}[(M-N)\times N+\frac{(N+j+2)(N-j-1)}{2}+2(j-i)]=\boldsymbol{A}(i,M-1-j) \quad (10)$$

while $N>j\geq i$, $N-j$ is odd number.

$$\boldsymbol{L}[(M-N)\times N+\frac{(N+j+3)(N-j-2)}{2}+2(j-i)+1]=\boldsymbol{A}(i,M-1-j) \quad (11)$$

while $N>j\geq i$, $N-j$ is even number.

$$\boldsymbol{L}[(M-j-1)\times N+2(N-i)-1]=\boldsymbol{A}(i,M-1-j) \quad (12)$$

while $j\geq N$, $M-j$ is odd number.

Among (12), when *M-N is odd number* and $j=N$:

$$\boldsymbol{L}[(M-N-1)\times N+N-i-1]=\boldsymbol{A}(i,M-1-j) \quad (13)$$

$$\boldsymbol{L}[(M-j-1)\times N+2(N-i)-2]=\boldsymbol{A}(i,M-1-j) \quad (14)$$

while $j\geq N$, $M-j$ is even number.

2.3 Folding algorithm

The line of pixels L was further mapped to a new $N\times M$ rectangular image, B. The map from line L to image B was described with the following formula:

$$B(i,j)=L(i\times M+j) \quad (15)$$

3. Extension image to three-dimension

The rectangular image consists of $N\times M$ pixels with K gray levels. The gray level value of each pixel A was in decimal which can be expressed as a binary number.

$$A(i,j)=\sum A_n(i,j)\times 2^n \qquad n=0,1,\cdots,\ log_2K-1 \quad (16)$$

For example, if K=512, then $log_2K-1=8$,

$$\begin{aligned}A(i,j)=\sum A_k(i,j)\times 2^i &= A_0(i,j)\times 2^0+A_1(i,j)\times 2^1+A_2(i,j)\times 2^2+A_3(i,j)\times 2^3\\ &+A_4(i,j)\times 2^4+A_5(i,j)\times 2^5+A_6(i,j)\times 2^6+A_7(i,j)\times 2^7\end{aligned} \quad (17)$$

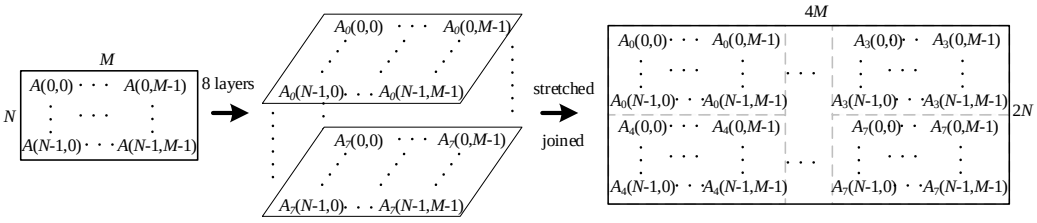

Fig.3. Extension to three-dimension

The plain image could be divided into eight layers. The first layer was composed by the lowest coefficients of the binary number of image' values, the second layer was composed by the second coefficients…and so on. Lastly, the eight layers were stretched and joined to a new $2N\times4M$ image as shown in Fig.3.

4. An image encryption approach

In this section a 3D rectangular image encryption approach was presented based on the improved chaotic maps, shown in Fig.4.

Firstly, the plain image was extended to a 3D image whose size was $2N\times4M$, as shown in Fig.1.

The pixels permutation was achieved by the improved chaotic maps. The iteration number of the right-map permutation or left-map permutation could be used as secret keys in image encryption. When a secret key is in decimal, from the least significant digit to the most significant digit, each digit (0-9) corresponds to the iteration number of the different maps alternately. Supposing a secret key was "123" which meant the image was firstly mapped through one iteration of left-map permutation, then 2 iterations of the right-map permutation, and lastly 3 iterations of the left-map permutation as shown in Fig. 1.

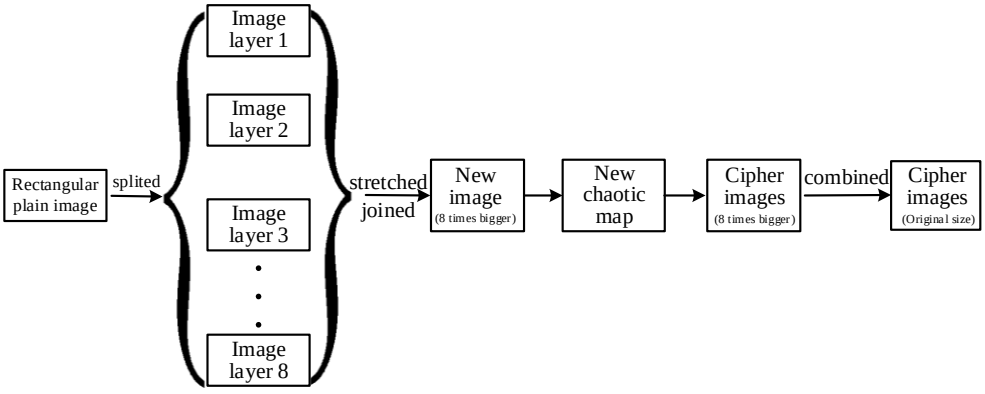

Fig.4. An image encryption approach base on 3D mapping

Lastly, the 8 times bigger cipher images were combined to the original size ($N \times M$) cipher image. The image was unrecognizable.

An image encryption approach was carried out. Four plain images were encrypted by the maps using the key "123456". The size of plain image was 240×256 (there is no limit to image size). By Fig.5, it could be seen that the plain images had been encrypted. Running time of optimized software programs were 0.09s, 0.09s, 0.09s, 0.09s and the time of four decryptions were 0.08s, 0.08s, 0.07s, 0.08s (the CPU of PC was Intel's core i5 1.6Ghz with 4GB RAM, and the operating system was Windows 10). The 3D map had the same speed as 2D map in [9].

The time of encryptions algorithm with a 256×256 plain image in Dagadu's [18], Slimane et al.'s [19], Khan et al.'s [20] and Essaid et al.'s [21] was 0.3457s, 0.4139s, 0.2854s and 0.0933s. It showed that the new encryption algorithm was relatively fast.

4.1 Key space

The key space could be calculated according to the length of the key. Suppose the key was represented in decimal. The relationship between the key space size and the key length was shown in Table 1.

The key space of new encryption algorithm was much bigger than the key space of baker map in [7] which was no more than 2^{n-1}. The total key space in [22], [23] and [24] approximately reckoned as 2^{532}, 2^{512} and 2^{372}. Thus, the key space of the new encryption algorithm was enough large to resist brute force attacks. But '0' was a weak keys in the new encryption algorithm and the whole encryption time would disclosure the iteration number of map. To solve this problem, each bit of the key could be mapped to a value in a certain interval [25].

Table 1

Key space size VS keys length

Key length (digit)	6	7	8	9
Key space size	10^6	10^7	10^8	10^9

4.2 Correlation

The new image encryption algorithm had very good confusion properties without diffusion mechanism. The plain images have high correlation among pixels, but the correlation was completely destroyed in cipher images which could be seen in Table 2.

Correlation of two adjacent pixels in a cipher image, $cov(x,y) = E(x - E(x))(y - E(y))$, $r_{xy} = \dfrac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}$, Where x and y were gray-scale values of two adjacent pixels in the image.Fig.6 showed the correlations of two

horizontally adjacent pixels in plain image and cipher image: the correlation coefficients were shown in Table 2.

Table 2

Correlation coefficients of two adjacent pixels

	Plain image1	Cipher image1	Plain image2	Cipher image2	Plain image3	Cipher image3	Plain image4	Cipher image4
horizontal	0.9404	0.0053	0.9325	0.0043	0.8085	0.0050	0.7202	0.0028
vertical	0.9683	0.0001	0.9338	0.0033	0.7489	0.0021	0.7372	0.0067
diagonal	0.9024	0.0031	0.8910	0.0099	0.7132	0.0019	0.5493	0.0038

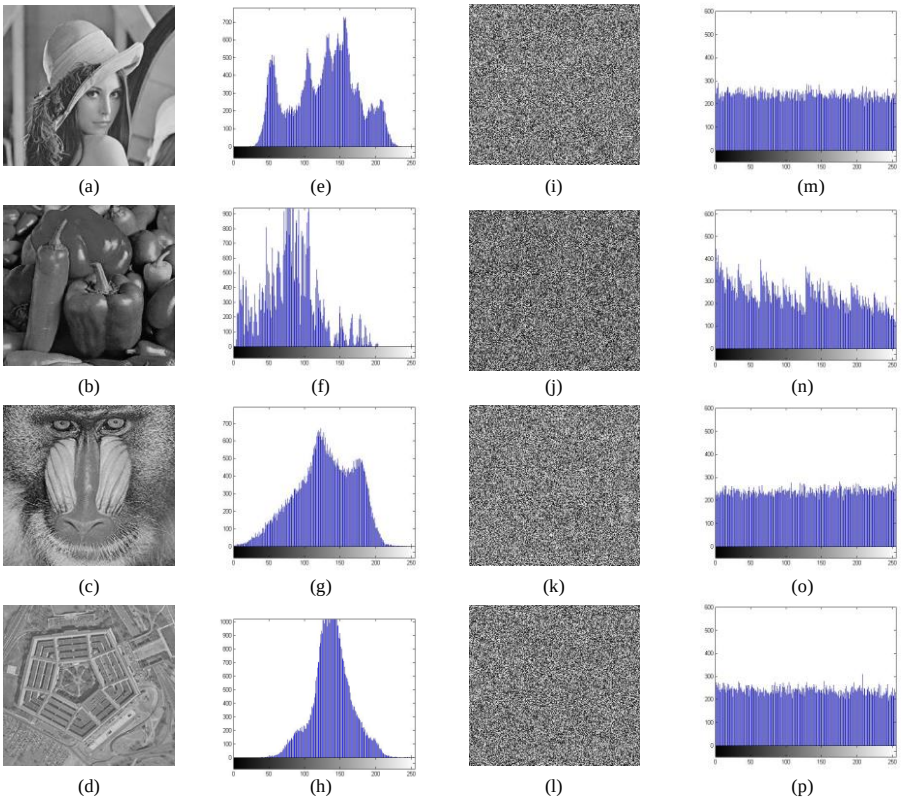

Fig. 5. Image, (a)-(d) plain images, (e)-(h) histogram of plain images (i)-(l)cipher images, and (m)-(p) histogram of cipher images

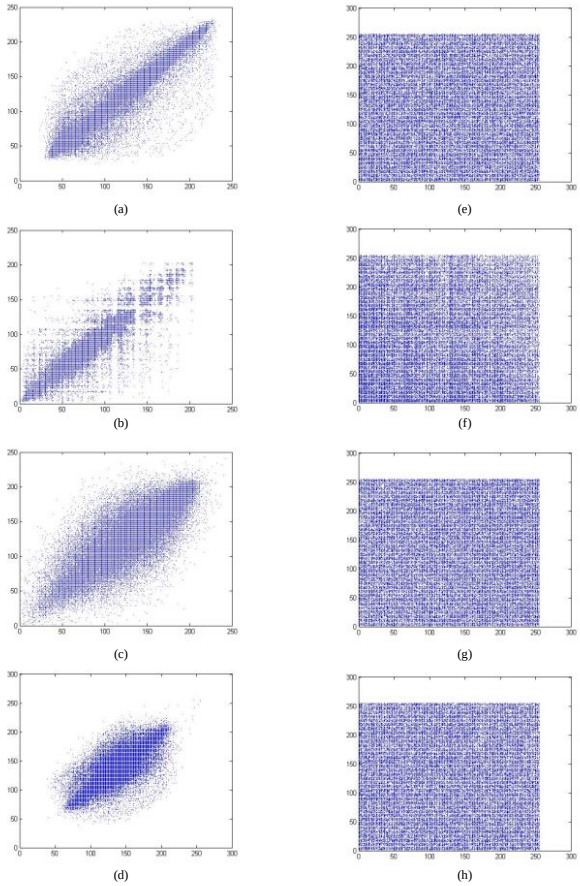

Fig. 6. Correlations of adjacent pixels, (a)-(d) plain images, (e)-(h) cipher images

4.3. Fixed point ratio

Fixed point ratios could be seen in Table 3. It meant the position of over 99% pixels in plain image were changed comparing with the cipher images.

Table 3

Fixed point ratios

	Cipher image1	Cipher image2	Cipher image3	Cipher image4
Fixed point ratio	99.66%	99.57%	99.66%	99.64%

4.4. Change of the gray

Changes of the gray could be seen in Table 4. It meant the average values of the pixels in plain image were changed over 63% comparing with the pixels in the cipher images.

Table 4

Changes of the gray

	Cipher image1	Cipher image2	Cipher image3	Cipher image4
Changes of the gray	67.5119	68.9256	66.4440	63.2617

4.5. r-m self-relevance

Where r=1, the r-m self-relevance could see in Table 5. It could be seen the self-relevance of cipher image significantly reduced compared with the plain image. Those meant the correlation between each pixel and adjacent pixels was destroyed and effect of permutation was very good.

Table 5

Self-correlation of images

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Plain image1	0.38	0.38	0.43	0.48	0.52	0.55	0.58	0.60	0.62	0.64	0.66	0.67	0.69	0.70	0.71	0.72	0.73	0.74	0.75	0.76
Cipher image1	0.12	0.12	0.13	0.13	0.13	0.14	0.14	0.14	0.15	0.15	0.15	0.16	0.16	0.16	0.17	0.17	0.17	0.18	0.18	0.18
Plain image2	0.35	0.38	0.41	0.44	0.47	0.49	0.51	0.54	0.56	0.58	0.60	0.61	0.63	0.65	0.66	0.67	0.69	0.70	0.71	0.72
Cipher image2	0.12	0.12	0.13	0.13	0.13	0.14	0.14	0.15	0.15	0.15	0.16	0.16	0.16	0.17	0.17	0.17	0.18	0.18	0.18	0.19
Plain image3	0.16	0.18	0.20	0.22	0.24	0.26	0.28	0.29	0.31	0.33	0.34	0.36	0.37	0.38	0.40	0.41	0.42	0.43	0.44	0.46
Cipher image3	0.12	0.12	0.13	0.13	0.13	0.14	0.14	0.14	0.15	0.15	0.15	0.16	0.16	0.16	0.17	0.17	0.17	0.18	0.18	0.18
Plain image4	0.21	0.22	0.25	0.28	0.31	0.33	0.36	0.38	0.40	0.42	0.44	0.45	0.47	0.49	0.50	0.51	0.53	0.54	0.55	0.56
Cipher image4	0.12	0.12	0.13	0.13	0.13	0.14	0.14	0.14	0.15	0.15	0.15	0.16	0.16	0.16	0.17	0.17	0.17	0.18	0.18	0.18

5. Summary

Most chaotic maps only shuffle the positions of image pixel in image encryption. However, permutation-only encryption is not safe enough because it does not change the statistical characteristics of images. Some papers used diffusion mechanisms or high dimensional chaotic maps to encrypt image for avoiding potential safety issues. Nevertheless, most encryption algorithms were unable to handle rectangular image.

Here it designed a three-dimensional image encryption approach which could handle rectangular image and change the statistical characteristics of images. It firstly divided the rectangular images into eight layers by pixel' value. Then the layers jointed to an 8 times bigger image. It used a new chaotic map to permute pixels of image. Lastly, the cipher image was combined to the original size cipher image. The process achieved permutation and confusion at the same time. The paper designed an image encryption approach to encryption for

different images. Some analyses verified security of the encryption and the approach is also suitable for larger images. The new encryption algorithm had faster encryption speed and larger key space.

Acknowledgements

Authors gratefully acknowledge the projects supported by national natural science foundation of Hunan province of China (2018JJ4039) and the projects supported by scientific research fund of Hunan provincial education department (17A048).

R E F E R E N C E S

[1] *Elgendy, Fatma, et al,* "Chaos-based model for encryption and decryption of digital images", Multimedia Tools & Applications, **vol. 75**, no. 18, 2015, pp. 1-25.
[2] *J. M. Zheng, W. Z. Gao,* "Color image encryption algorithm based on chaotic map", Comp. Eng. Design., **vol. 32**, no. 9, 2011, pp. 2934-2937.
[3] *Y. Wang, K. Wong, X. Liao, G.Chen,* "A new chaos-based fast image encryption algorithm", Applied Soft Comput., **vol. 11**, no. 1, 2011, pp. 514-522.
[4] *J. X. Chen, Z. L. Zhu, C. Fu, H. Yu, L. B. Zhang,* "A fast chaos-based image encryption scheme with a dynamic state variables selection mechanism", Communications in Nonlinear Science and Numerical Simulation, **vol. 20**, no. 3, 2015, pp. 846-860.
[5] *X. J. Tong,* "The novel bilateral - Diffusion image encryption algorithm with dynamical compound chaos", Journal of Systems & Software, **vol. 85**, no. 4, 2012, pp. 850-858.
[6] *A. S. Menon, K. S. Sarila,* "Image encryption based on chaotic algorithms: An overview", Int. J. Science, Engineering and Technology Research, **vol. 2**, no. 6, 2013, pp. 1328-1332.
[7] *J. Fridrich,* "Symmetric ciphers based on two-dimensional chaotic maps", Int. J. Bifurcat Chaos, **vol. 8**, no. 6, 1998, pp. 1259-1284.
[8] *B. Mondal, P. Kumar, S. Singh,* "A chaotic permutation and diffusion-based image encryption algorithm for secure communications", Multimedia Tools & Applications, no. 1, 2018, pp. 1-22.
[9] *F. Huang, Y. Feng,* "A symmetric image encryption scheme based on a simple novel 2D map", Int J Innovative Computing, Information and Control, **vol. 3**, no. 6, 2007, pp. 591-1600.
[10] *Y. B. Mao, R. G. Chen, G. S. Lian,* "A novel fast image encryption scheme based on 3D chaotic baker maps", Int. J. Bifurcat. Chaos, **vol. 14**, no. 10, 2004, pp. 3613-3624.
[11] *R. G. Chen, Y. B. Mao, C. K. Chui,* "A symmetric image encryption scheme based on 3D chaotic cat maps", Chaos Solitons Fractals, **vol. 21**, no. 3, 2004, pp. 749-761.
[12] *L. Xu, X. Gou, Z. Li, J. Li,* "A novel chaotic image encryption algorithm using block scrambling and dynamic index-based diffusion", Opt Lasers Eng, **vol. 91**, 2017, pp. 41-52
[13] *T. G. Gao, Z. Q. Chen,* "A new image encryption algorithm based on hyper-chaos", Phys Lett A, **vol. 372**, 2008, pp. 394-400
[14] *X. Y. Wang, Q. Wang,* "A fast image encryption algorithm based on only blocks in cipher text", Chin Phys B, **vol. 23**, no. 3, 2014, pp. 165-172
[15] *X. L. Chai, Z. H. Gan, M. H. Zhang,* "A fast chaos-based image encryption scheme with a novel plain image-related swapping block permutation and block diffusion", Multimed Tools Appl, **vol. 76**, 2017, pp. 15561-15585.
[16] *S. Guo, Y. Liu, L. Gon*g, et *al,* "Bit-level image cryptosystem combining 2D hyper-chaos

with a modified non-adjacent spatiotemporal chaos", Multimedia Tools & Applications, **vol. 77**, no. 3, 2018, pp. 1-22.

[17] *F. Huang, G. X. Zhang*, "A New Image Permutation Approach Using Combinational Chaotic Maps", Information Technology Journal, **vol.** 12, no. 4, 2013, pp. 835-840.

[18] *C. D. Joshua, L. Jian, M. G. Kamachi, et al,* "DWT based encryption technique for medical images", In: 2016 13th International computer conference on wavelet active media technology and information processing (ICCWAMTIP), 2016, pp. 252-255.

[19] *N. B. Slimane, K. Bouallegue, M. Machhout,* "Nested chaotic image encryption scheme using two-diffusion process and the Secure Hash Algorithm SHA-1", In: International conference on control engineering & information technology, 2016, pp. 1-5.

[20] *J. S. Khan, J. Ahmad, S. O. Hwang,* "An efficient imageencryption scheme based on: Henon map, skew tent mapand s-box", In: 2015 6th International conference on modeling, simulation and applied optimization (ICMSAO), 2015, pp.1-6.

[21] *M. Essaid, I. Akharraz, A. Saaidi, et al,* "A new image encryption scheme based on confusion-diffusion using an enhanced skew tent map", Procedia Comput Sci, **vol. 127**, 2018, pp.539-548.

[22] *Y. Q. Luo, J. Yu, W. R. Lai, et al,* "A novel chaotic image encryption algorithm based on improved baker map and logistic map", Multimedia Tools and Applications, **vol. 78**, no. 15, 2019, pp.22023-22043.

[23] *E. Yavuz, R. Yazıcı, M. C. Kasapbası MC, et al,* "A chaos-based image encryption algorithm with simple logical functions", Comput Electr Eng, **vol. 54**, 2016, pp.471-483.

[24] *Y. Zhang,* "The uified image encryption algorithm based on chaos and cubic S-Box", Inf Sci. **vol. 450**, 2018, pp.361-377.

[25] *F. Huang, X. J. Ouyang, X. P. Liu,* "An improved image encryption algorithm based on a new two-dimensional map", Advanced Materials Research, **vol. 216**, 2011, pp.293-296.