http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (4): 835-840, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.835.840

A New Image Permutation Approach Using Combinational Chaotic Maps

[1]Feng Huang and [2]Guixiang Zhang
[1]College of Electrical and Information Engineering, Hunan Institute of Engineering,
Xiangtan, 411105, China
[2]Jianglu Machinery and Electronics Group Co., Ltd.,
Xiangtan, 411100, China

Abstract: Image encryption is widely needed to ensure image security. In order to rapidly charge the high correlation among pixels of image, the study proposed a new image permutation approach using combinational chaotic maps. The new chaotic map was realized by processing image stretch and fold. Firstly, a square image was divided into two isosceles triangles along the diagonal direction. Utilizing the difference of the pixel numbers of adjacent columns, each pixel in a column was inserted to the corresponding adjacent column. Then, the plain image could be stretched to a line. Finally, the line was folded over to the cipher image. Utilizing two different methods of fold, the study designed four maps for permutation. A new permutation approach was developed which used a six decimal numbers as the keys. The deciphering process was an invertible process using the same keys. The simulation results showed the speed of permutation is fast and high correlation among adjacent pixels was rapidly charged. And the encryption could satisfy the high security requirements or as a part of the other encryption.

Key words: Image encryption, image permutation, chaotic map, information security, baker map

INTRODUCTION

With the fast developments of computer network technology, more and more images are transmitted over the Internet. Sensitive image, such as military image, must be protected while transmitted over public channels. How to protect the security of images becomes a serious problem. The encryption is an important tool to protect security of images from unauthorized access. Some traditional encryption technologies such as Data Encryption Algorithm etc. can be used to protect the security of images. But for some intrinsic features of image, such as bulk data capacity and high correlation among adjacent pixels, it is known that conventional encryption technologies have obvious limitations on practical image encryption (Socek *et al*., 2007; Mazloom and Eftekhari-Moghadam, 2009).

Image permutation is a method or a process for protecting image from undesirable attacks by converting it into a new image not recognizable by its attackers. Permutation can against statistical cryptanalysis. At the same time, it can charge the correlation among adjacent pixels. Image permutation usually is taken as a very important part of image encryption. Some new technologies are used in image permutation; act as SCAN patterns (Maniccam and Bourbkis, 2004), chaotic map (Feng *et al*., 2006) etc. SCAN patterns generate very large number of scanning paths or space filling curves. The image encryption is performed by SCAN-based permutation of pixels and a substitution rule which together form an iterated product cipher. The SCAN patterns require the size of plain image must be even.

Chaos based image encryption provides a useful supplement to conventional methods (Kocarev, 2001; Zheng and Gao, 2011; Wang *et al*., 2011). Chaos has many characteristics which can be connected with the “confusion” and “diffusion” property in encryption, such as sensitive dependence on initial conditions and parameters, broadband power spectrum, randomness in the time domain, ergodicity, low-dimensional etc. In fact the idea of using chaos for encryption can be trace back to the classical Shannon’s paper (Shannon, 1949) in which the basic stretch-and-fold mechanism of chaos was proposed which could be used for encryption.

In particular, many chaotic image encryption schemes are based on permutation. There are some typical chaotic maps suitable for image permutation such as the Standard map, the Logistic map, the Cat map, the Baker map, the Tent map, the chaotic matrix etc. Among them, the Baker map, the Cat map and the Standard map (Patidar *et al*.,

Corresponding Author: Feng Huang, College of Electrical and Information Engineering, Hunan Institute of Engineering, Xiangtan, 411105, China Tel: 086-073158688934

2011) attracted much attention. Fridrich (1998) obtained a symmetric image encryption scheme. It is shown that the permutations induced by the Baker map behave as typical random permutations. The Baker map can be used in image permutation. But the version B of the Baker map doesn't have simple formula and the key are limited by size of image. Two symmetric image encryption schemes based on three-dimensional chaotic maps are proposed (Mao *et al.*, 2004; Chen *et al.*, 2004). They employ chaotic maps to shuffle the positions of image pixels which are the image permutation. In order to strengthen security of image encryption, a large chaotic permutation matrix was designed to achieve the high performance of pseudorandom permutation in by Yoon and Kim (2010).

A new chaotic map, the line map was proposed by Feng *et al.* (2006). An image encryption scheme based on the two maps of the line map was developed. But there are two serious problems in the study: It has many week keys and duplicate keys and the encryption time may disclosure the key. It seriously affects the security of the encryption algorithm.

In order to avoid the above-mentioned problems, the study proposed a new image permutation approach using combinational chaotic maps. There are four chaotic maps different from such two maps. A new image permutation approach based on the four maps is developed which use a six decimal numbers as the key. The simulation results validate the permutation approach.

THE KEY GENERATION

On many occasions, people are required using six decimal numbers as a password, such as Internet, banks, games. Here the key in image permutation also is six decimal numbers. Suppose the key is $k_1k_2k_3k_4k_5k_6$, here $0 = k_i = 9$ (i = 1, 2, 3, 4, 5, 6). The process of the key generation is as following:

- Firstly, it uses the key to generate another key (key$_1$). The value of each part in key$_1$ is equal to the value of corresponding part in key which add 1. If the value is 6 in key, then, it equal to 7 in key$_1$. Act as the key is “671522”, then the key$_1$ is (6+1) (7+1) (1+1) (5+1) (2+1) (2+1). It is “782633”
- Secondly, it uses key and key$_1$ to generate another key (key$_2$). The value of each part in key$_2$ is equal to the value of corresponding part in the key add which in the key$_1$ divided by n. Here n is the first non-zero number add 1 in key. It takes the first two digits in each part of key. It takes the first two digits in each part of key. Act as the key is 671522 and the key$_1$ is 782633, then, key$_2$ is [(6+7)/7] [(7+8)/7] [(1+2)/7] [(5+6)/7] [(2+3)/7] [(2+3)/7]. Taking the first two significant digits, the actual key$_2$ is “182142157171”

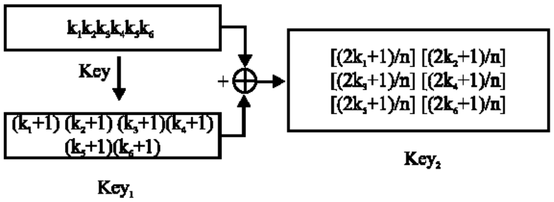

$k_1k_2k_3k_4k_5k_6$
Key
(k_1+1) (k_2+1) $(k_3+1)(k_4+1)$ $(k_5+1)(k_6+1)$
Key$_1$
+
$[(2k_1+1)/n]$ $[(2k_2+1)/n]$ $[(2k_3+1)/n]$ $[(2k_4+1)/n]$ $[(2k_5+1)/n]$ $[(2k_6+1)/n]$
Key$_2$

Fig. 1: The process of the key generation

Table 1: The relationship between the number in the key and the order of four maps

Number	Cipher image (key$_2$)
0	Firstly map A, then map B, map C, map D
1	Firstly map B, then map A, map C, map D
2	Firstly map B, then map C, map A, map D
3	Firstly map B, then map C, map D, map A
4	Firstly map A, then map C, map B, map D
5	Firstly map A, then map C, map D, map B
6	Firstly map C, then map A, map B, map D
7	Firstly map A, then map B, map D, map C
8	Firstly map D, then map A, map B, map C
9	Firstly map A, then map D, map B, map C

The process of the key generation can be seen in Fig. 1. It can use key$_2$ to encrypt image. Image permutation can be achieved by chaotic maps. Here suppose there are four chaotic maps in the study: map A, map B, map C and map D. Each number in the key means the order of four maps. It can see in Table 1. A simple example is given here. Assuming the key is “1821”. From Table 1 the number “1” means firstly map B, then map A, map C, map D. By analogy it can see the process of permutation is firstly map B, then map A, map C, map D, map D, then map A, map B, map C, map B, then map C, map A, map D, map B, then map A, map C, map D.

INTRODUCTION OF THE CHAOTIC MAPS

Suppose that a square image consists of N×N pixels with L gray levels. The key chaotic map utilizes an important characteristic of images, which is, each pixel in image can be inserted into the corresponding two adjacent pixels.

The new chaotic map is realized by processing image stretch-and-fold. Firstly, a square image was divided into two isosceles triangles along the diagonal, utilizing the difference of the pixel numbers of two adjacent columns of the triangles, each pixel in a column was inserted to the next adjacent column. Then, the plain image could be stretched to a line. Finally, the line was folded over to a new square image whose size was the same as the plain

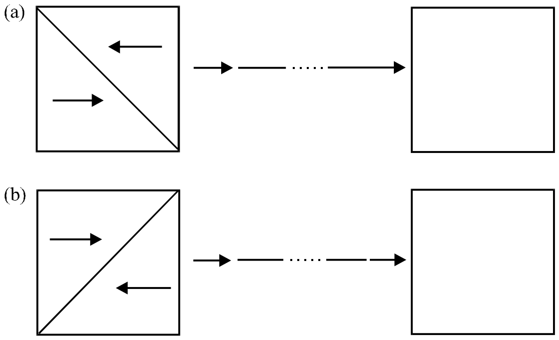
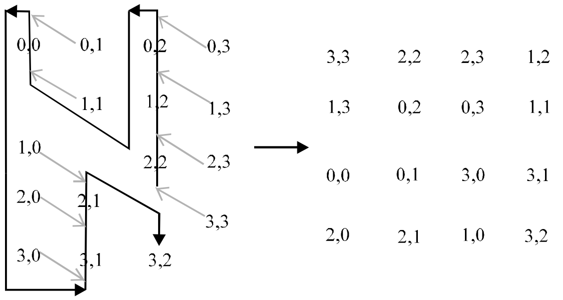


Fig. 2 (a-b): Principle of the chaotic map, (a) The left map, (b) The right map



3,3	2,2	2,3	1,2
1,3	0,2	0,3	1,1
0,0	0,1	3,0	3,1
2,0	2,1	1,0	3,2

Fig. 3: The process of the left map with a 4×4 image

image. The map shuffles the positions of image pixels. All the process is invertible which is seen in Fig. 2. The chaotic map is divided into two forms: The left map and the right map.

In order to explain the chaotic map more clearly, an example is given here. The process of the left map permutation is shown in Fig. 3. The image with 4×4 pixels, that is N = 4. In upper triangle, pixel (3, 3) can be inserted before pixel (2, 2). Pixel (2, 3) can be inserted between pixels (2, 2) and (1, 2). Pixel (1, 3) can be inserted between pixels (1, 2) and (0, 2) and so on. Then, the pixels join to a part of line: (3,3), (2,2), (2,3), (1,2), (1,3), (0,2).... in lower triangle, pixel (3,0) can be inserted before (3,1). Pixel (2,0) can be inserted between pixel (3,1) and (2,1) and so on. The pixels join to another part of line. Then, it connects two parts to a line. Lastly it is from a line to a square image different from the originally one. The right map is symmetric with the left map.

Supposing the dimension of a square image is N×N, where N is an integer. A (i, j) is the matrix of a square image, in which each element corresponds to a gray-level value of the pixel (i, j); L(i), i = 0, ..., N-1, j = 0, ..., N-1. N is a one dimensional vector mapped from A.

- **The left map:**

$$L[\frac{(N+j+2)(N-j-1)}{2}+2(j-i)]=A(i,j) \qquad (1)$$

while j≥i, N-j is odd number, i = 0, ..., N-1, j = 0, ..., N-1.

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

while j≥i, N-j is odd number, i = 0, ..., N-1, j = 0, ..., N-1.

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

while j≥i, N-j is odd number, i = 0, ..., N-1, j = 0, ..., N-1.

$$L[\frac{N^2+N+(2N-j)\times(j-1)}{2}+2(N-i)-1]=A(i,j) \qquad (4)$$

while j≥i, N-j is odd number, i = 0, ..., N-1, j = 0, ..., N-1.

- **The right map:** The right map is shown in Fig. 2b; first, a mirror process of the image is made. The algorithm of the mirror image is described with the following formula:

$$A'(i,j)=A(i,N-1-j) \qquad (5)$$

where, A′ is the matrix of the mirror image of a square image A:

$$L[\frac{(N+j+2)(N-j-1)}{2}+2(j-i)]=A(i,N-1-j) \qquad (6)$$

while j≥i, N-j is odd number, i = 0, ..., N-1, j = 0, ..., N-1.

$$L[\frac{(N+j+3)(N-j-2)}{2}+2(j-i)+1]=A(i,N-1-j) \qquad (7)$$

while j≥i, N-j is odd number, i = 0, ..., N-1, j = 0, ..., N-1.

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

while j≥i, N-j is odd number, i = 0, ..., N-1, j = 0, ..., N-1.

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

while j≥i, N-j is odd number, i = 0, ..., N-1, j = 0, ..., N-1.

- **Folding algorithm:** The line of N×N pixels L is further mapped to a same size N×N square image, B. Here there are two methods to do this. It can be seen in Fig. 4

Method one is shown in Fig. 4a. The map from line L to image B is described with the following formula:

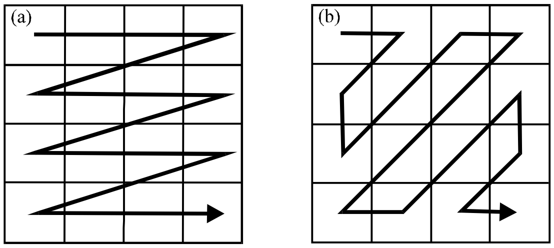


Fig. 4 (a-b): The process of folding algorithms

$$B(i, j) = L(i \cdot N + j) \tag{10}$$

while j ≥ i, N-j is odd number, i = 0, ..., N-1, j = 0, ..., N-1.

Method two is shown in Fig. 4b. The map from line L to image B is described with the following formula:

$$B\,(i, j) = L(\sum_{k=0}^{i+j} k + i) \tag{11}$$

while j ≥ i, N-j is odd number, i = 0, ..., N-1, j = 0, ..., N-1.

$$B(i, j) = L(\sum_{k=0}^{i+j} k + i - (i + j - N + 1)^2) \tag{12}$$

while j = i, N-j is odd number, i = 0, ..., N-1, j = 0, ..., N-1.

- **Four maps:** Combination of the two maps and two folding methods, it designs four maps to encrypt image: Map A uses the left map and method one, map B uses the left map and method two, map C uses the right map and method one and map D uses right map and method two

A NEW IMAGE PERMUTATION APPROACH

A new image permutation approach is carried out based on the four maps. The plain image and cipher image are shown in Fig. 5. It has 256×256 pixels with 256 grey levels. The plain image is encrypted using the maps by the key_1 "1111" and key_2 "1821421571 71". It can be seen that the plain image has been encrypted. The plain image and the cipher image are equal for every pixel; the decrypted image is recovered completely. It shows the image encryption using the chaotic map has no message loss. The time of encryption by key_1 is 0.1319 sec and the time of encryption by key_2 is 0.8478 sec; the time of decryption by key_1 is 0.0929 sec and the time of encryption by key_2 is 0.7844 sec. (the CPU of PC is Intel's core i5 2.5 GHz, the ram is 4G and the operating system is Windows 7).

Correlation of two adjacent pixels in a cipher image, cov (x, y) = E (x-E(x)) (y-E(y)):

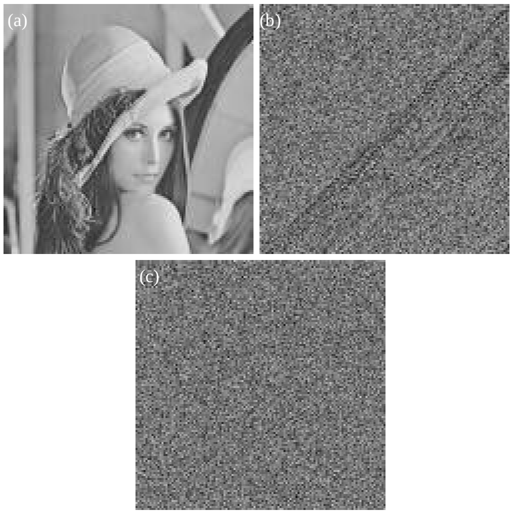


Fig. 5 (a-c): (a) Plain image and Cipher image of (b) key_1 and (c) key_2

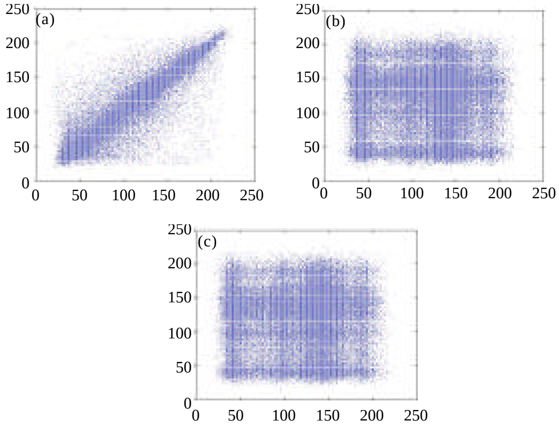


Fig. 6 (a-c): Correlations of adjacent pixels (a) Plain image and Cipher image of (b) key_1 and (c) key_2

Table 2: Correlation coefficients of two adjacent pixels

	Plain image	Cipher image (key_1)	Cipher image (key_2)
Horizontal	0.9442	0.0004	0.0007
Vertical	0.9711	0.0043	0.0021
Diagonal	0.9187	0.0315	0.0070

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}$$

where, x and *y* are gray-scale values of two adjacent pixels in the image.

Figure 6 shows the correlations of two horizon-tally adjacent pixels in plain image and cipher image (key_1 and key_2): The correlation coefficients are 0.9442, 0.0004 and 0.0007. Similar results for diagonal and vertical directions were obtained and are shown in Table 2:

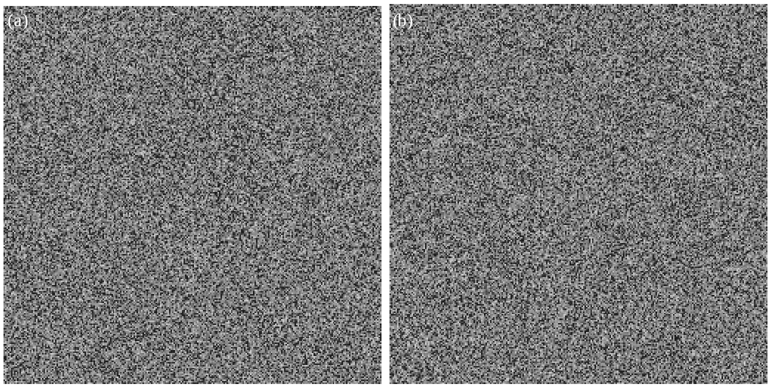


Fig. 7 (a-b): The sensitivity of keys (a) Decrypted cipher image of key$_1$ and (b) Decrypted cipher image of key$_2$

Table 3: 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
Lena	0.41	0.41	0.46	0.50	0.54	0.57	0.60	0.62	0.64	0.66	0.68	0.69	0.70	0.71	0.73	0.73	0.74	0.75	0.76	0.77
key$_1$	0.15	0.15	0.16	0.17	0.18	0.19	0.20	0.20	0.21	0.22	0.22	0.23	0.24	0.24	0.25	0.26	0.26	0.27	0.27	0.28
key$_2$	0.14	0.14	0.15	0.16	0.16	0.17	0.18	0.18	0.19	0.19	0.20	0.21	0.21	0.22	0.22	0.23	0.24	0.24	0.25	0.25

- **Fixed point ratio:** Where key$_1$ is "1111" BD = 0.72%, where key$_2$ is "182142157171" BD = 0.72%. Those mean the positions of the 99.3% plain image pixels are charged
- **Change of the gray:** Where key$_1$ is "1111" GAVE = 52.5869, where key$_2$ is "182142157171" GAVE = □52.3459. Those mean the average values of the pixels are charged by 20%
- **Sensitivity of keys:** Assume that an image is encrypted using the map with the key$_{2□}$ "182142157171", just as seen in Fig. 5. Now, the least significant bit of the key is changed and the test is done for image decryption. The key$_2$ "182142157171" is changed to key$_3$ "182142157172" and key$_4$ "182142157170", both of which are used to decrypt the cipher image by the original key$_2$ respectively. The two decrypted images by two different keys are shown in Fig. 7. It can be seen that the image can't be decrypted by the two keys, which are different from the correct key only in the least one bit. Therefore, the security of the image permutation is much effective
- **r-m self-relevance:** Where r = 1, the r-m self-relevance can seen in Table 3, here key$_1$ is "1111", key$_2$ is "182142157171". It can be proved the self-relevance of cipher image significantly reduced compared with the plain image. The value of self-relevance is even smaller than the value when m = 1. Those mean the effect of permutation is very good.

CONCLUSION

Image permutation can charge the correlation among adjacent pixels. It converts the plain image into a new image not recognizable by its attackers. Chaotic mapping scrambling is a very common way. But there are serious problems: it has many week keys and duplicate keys and the encryption time may disclosure the key. It seriously affects the security of the encryption. In order to avoid the problems, the study proposed four maps for image permutation. The simulation results validated the proposed image permutation approach. The advantages of the approach can be described as follows: (1) the approach is safe and convenient, (2) the image encryption and decryption based on the map have no message loss, (3) the key space can be enough big to satisfy different security requirements (if hardware is enough) and (4) the size of the ciphered image is equal to that of the plain image.

ACKNOWLEDGMENTS

Authors gratefully acknowledge the Projects Supported by Scientific Research Fund of Hunan Provincial Education Department of China (12A032) for supporting this research. Project supported by Hunan Provincial Natural Science Foundation of China (12JJ9018) supports the research. Project supported by Hunan Provincial Science and Technology plan project of China (2011GK3141) supports the research.

REFERENCES

Chen, G., Y. Mao and C.K. Chui, 2004. A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos Solitons Fractals, 21: 749-761.

Feng, Y., L.J. Li and F. Huang, 2006. A symmetric image encryption approach based on line maps. Proceedings of the 1st International Symposium on Systems and Control in Aerospace and Astronautics, January 19-21, 2006, Harbin, pp: 1367-1367.

Fridrich, J., 1998. Symmetric ciphers based on two-dimensional chaotic maps. Int. J. Bifurcat. Chaos, 8: 1259-1284.

Kocarev, L., 2001. Chaos-based cryptography: A brief overview. IEEE Circuits Syst. Magaz., 1: 6-21.

Maniccam, S.S. and N.G. Bourbkis, 2004. Image and video encryption using SCAN patterns. Pattern Recogn., 37: 725-737.

Mao, Y., G. Chen and S. Lian, 2004. A novel fast image encryption scheme based on 3D chaotic baker maps. Int. J. Bifurcat. Chaos, 14: 3613-3624.

Mazloom, S., A.M. Eftekhari-Moghadam, 2009. Color image encryption based on coupled nonlinear chaotic map. Chaos, Solitons Fractals, 42: 1745-1754.

Patidar, V., N.K. Pareek, G. Purohit and K.K. Sud, 2011. A robust and secure chaotic standard map based pseudorandom permutation-substitution scheme for image encryption. Opt. Commun., 284: 4331-4339.

Shannon, C.E., 1949. Communication theory of secrecy systems. Bell Syst. Tech. J., 1: 656-715.

Socek, D., S. Magliveras, D. Culibrk, O. Marques, H. Kalva and B. Furht, 2007. Digital video encryption algorithms based on correlation-preserving permutations. J. Inform. Secur. 10.1155/2007/52965

Wang, Y., K. Wong, X. Liao and G. Chen, 2011. A new chaos-based fast image encryption algorithm. Applied Soft Comput., 11: 514-522.

Yoon, J.W. and H. Kim, 2010. An image encryption scheme with a pseudorandom permutation based on chaotic maps. Commun. Nonlinear Sci. Numer. Simul., 15: 3998-4006.

Zheng, J.M. and W.Z. Gao, 2011. Color image encryption algorithm based on chaotic map. Comp. Eng. Design., 32: 2934-2937.