2010 3rd International Conference on Computer and Electrical Engineering (ICCEE 2010)
IPCSIT vol. 53 (2012) © (2012) IACSIT Press, Singapore
DOI: 10.7763/IPCSIT.2012.V53.No.2.16

Algorithm of Image Encryption based on Permutation Information Entropy

CAO Guang-hui[1,2+], Hu Kai[2], Yang He[2] and E Xu[1]

[1] Electronic & Information Engineering College, Liaoning university of Technology

Jinzhou, China

[2] School of Computer Science and Technology, Beijing University of Aeronautics and Astronautics

Beijing, China

Abstract— Criterion for choosing chaos map to drive image bit permutation based on chaos permutation information entropy is proposed. An algorithm for image bit permutation is designed based on the fact that the output trajectory of chaotic system is very unpredictable. Image smallest granularity scrambling, namely, bit space maximum scrambling is implemented by applying the chaos that has been selected. Image substitution algorithm according to the principle of chaos map stretch and fold is designed to finish image encryption transformation to enhance the ability to resist attack. The algorithm's key space and resistance to differential attack, and the digital characteristics of cipher-image have been analyzed. Theoretical analyses and computer simulations both confirmed that the proposed algorithm is better than Ye's and Huang's algorithms in the ability of scrambling, decorrelation and anti-differential attack, can protect digital image effectively.

Keywords-chaos; permutation information entropy; sorting transformation

1. Introduction

There are two methods to effectively protect digital image information, one is digital watermark, the other is image encryption. Algorithms of image encryption fall into three categories: image pixel position permutation, image pixel value transformation, and the mixed transformation of position and value. Image position permutation includes matrix based transformation, for example: Arnold transformation [1], magic transformation [1], gray code and generalized gray code for permutation method [2], Permutation of IFS model based on fractal geometry [3] and permutation based on Hilbert curve [1], FASS curve and Tangram algorithm. Image position permutation belongs to image coordinate permutation. Image pixel value transformation includes changing the number of 0,1 bits of the original image pixel, such as the well known xor transformation, substitution transformation [4] and the permutation that didn't change the number of 0,1 bits of the original image, such as circular bit shift of pixel value[5].

In recent years, based on the study of chaos theory and cryptography, the similarities of their crucial concepts have been found, such as [6]: mixing property, and sensitivity to changes in initial conditions and parameters. Various algorithms about pixel bit permutation based on chaos drives have been come out, which can change pixel value and pixel position simultaneously. Classical examples have multi chaotic systems based pixel shuffle for image encryption [7], scrambling encryption algorithm of pixel bit based on chaos map [8], a chaotic image encryption scheme based on circular bit shift method [5]. Huang [7] proposed a method

+ Corresponding author.
E-mail address: caoguanghuineu@163.com

that turned image matrix into column, then converted each element in the column into binary value bit serial, divided 8 bits into four groups, fist permutation within the group, then between the groups. Permutation within the group, scrambling degree is not full. Ye [8] turned image into binary matrix firstly, then did image row permutation, followed by permutation of binary bit serial of each row. Bit permutation in the row, pixel bit scrambling degree isn't full. Fu [5] first scrambled the whole pixel matrix, then finished circular bit shift of each pixel. Adjacent bits in each pixel have correlation. These algorithms all finished image encryption based on bit level, however, every method didn't realize that each bit of any pixel can fall into any other bit position with the equal probability, scrambling degree didn't enough. An algorithm aimed at the above-mentioned shortcoming that can implement totally bit permutation and, at the same time, perform pixel value substitution is designed, not only have high scrambling degree, but can resist differential attack.

2. Basic Definition

(1) Chaos [9]: Continuous map f on a closed interval $I \subseteq R$, satisfying the following three requirements:

① For any natural number k, if $x_k \in I$, then $f^k(x_k) = x_k$, that say there exist period point with any order.

② For uncountable set $N \subset I$, not include periodic point in set N, for any $x, y \in N$, then $\overline{\lim_{x\to\infty}} \left| f^n(x) - f^n(y) \right| > 0$, $\underline{\lim_{n\to\infty}} \left| f^n(x) - f^n(y) \right| = 0$

③ For any periodic points $x, y \in N$, if there exists

$$\overline{\lim_{x\to\infty}} \left| f^n(x) - f^n(y) \right| > 0$$

f is called chaos mapping on interval I.

(2) Permutation entropy

Let $(x_n)_{n=1}^T, T \in N$, be a sequence which comes from real or simulated data. For m≥2, Let s_m be the group of permutation of length m, for which the cardinality is m!. Let $x_m(r) = (x_r, x_{r+1}, ..., x_{r+m-1}), 1 \le r < T-m+1,$ be a sliding window taken from the sequence $(x_n)_{n=1}^T$. One can say that the window $x_m(r)$ is of π type, $\pi \in s_m,$ if and only if $\pi = (i_1, i_2, ..., i_m)$ is the only element of s_m satisfying the two following conditions:

$$x_{r+i_1} \le x_{r+i_2} \le ... \le x_{r+i_m} \quad (1)$$

$$i_{s-1} < i_s \quad if \quad x_{r+i_{s-1}} = x_{r+i_s} \quad (2)$$

② Permutation entropy [10] :

$$H_s(x_n, m) = -\sum_{\pi \in s_m} p(\pi) \log p(\pi).$$

In which

$$p(\pi) = \frac{\#\{x_m(j),\ j = 1, 2, ..., k;\ x_m(j)\ is\ of\ \pi - type\}}{k},$$

$k < T-m+1$ is the probability of π type.

③ Permutation entropy per symbol [11] :

$$h(n) = H_s(x_n, m)/(m-1)$$

3. Image encryption algorithm based on chaos permutation information entropy

An algorithm of image encryption is proposed, the first step finished image pixel bit permutation which can realize pixel bit position permutation and change pixel value simultaneously. Image bit permutation is driven by nonlinear chaos dynamics. The most important but often ignored problem is:

Determining how to choose chaotic map among so many chaotic maps to accomplish image bit permutation.

3.1. Choosing chaotic map

There are a lot of chaotic maps, such as one-dimensional *L*ogistic map, two-dimensional Hénon map, three-dimensional Lorenz map and Chua map and so on. How to choose a good map which will be used to drive image bit permutation among so many chaotic maps is a very important challenge. According to document [11], chaos permutation information entropy criterion is proposed.

From the definition of permutation information entropy, one can know that the bigger the permutation information entropy is, the higher the strength for the image bit permutation is, furthermore, the more security.

chaotic permutation information entropy for most common chaos maps has been calculated as follows: (see Table Ⅰ)

In table Ⅰ, the first row denotes window width, the first column denotes chaos type. As can be seen from the experiment data, the permutation information entropy of Lorenz chaos map is higher than Logistic map, Henon map and Chua map, also the permutation information entropy of component x of Lorenz is larger than its y, z components. Thus, this paper adopted component x of Lorenz to drive the image bit permutation.

3.2. Chaos encryption algorithm

1) Image bit permutation

The detailed image bit permutation algorithm is described as follows:

Let I be the digital image with the size M*N.

Step 1. Chose suitable chaos dynamics equation according to chaos permutation information entropy, and gave a initial value x_1 or initial vector V_1 , and let K=1.

Step 2. Iterated the chaos dynamics equation which has been selected m=M*N*8 times, produced chaos sequence $\{x_1,x_2,\ldots,x_m\}$, or vector sequence $\{V_1,V_2,V_3,\ldots,V_m\}$. For vector sequence, chose the row which has the maximum permutation entropy calculated by permutation entropy formula among all the rows to form the chaos sequence $\{x_1,x_2,\ldots,x_m\}$.

Step 3. Sorting the chaos sequence $\{x_1,x_2,\ldots,x_m\}$ from small to large or based on some other rules, and got the wanted sequence $\{y_1,y_2,\ldots,y_m\}$.

Step 4. Calculated the set of scrambling address codes $\{t_1,t_2,\ldots,t_m\}$, where $t_i \in \{1,2,\ldots,m\}$. t_i is the new subscript of x_i in the sorted sequence $\{y_1,y_2,\ldots,y_m\}$.

Step 5. Turned the image into one-dimensional array, then converted each element in the array into bit series, connected all the bits in sequence to form bit sequence which length is M*N*8, then scrambled all the bits with scrambling address code set. Finally, turned the bit series into one-dimensional array, this process accomplished the image bit permutation.

TABLE I. COMMON CHAOS PERMUTATION INFORMATION ENTROPY

	2	3	4	5	6	7	8	9
Logistic	0.9757	1.1127	1.1094	1.0661	1.0272	0.9904	0.9622	0.9524
Hénon	0.9948	1.0966	0.9611	0.9870	0.9524	0.9324	0.8969	0.8702
Lorenz (x)	0.9991	1.2883	1.5150	1.6718	1.7495	1.7019	1.5403	1.3690
Lorenz (y)	0.9990	1.2883	1.5113	1.6627	1.7432	1.6985	1.5431	1.3689
Lorenz (z)	0.9932	1.2714	1.4422	1.5508	1.6072	1.5997	1.4987	1.3585
Chua(x)	1.0000	1.1656	1.2050	1.1814	1.1456	1.0985	1.0513	1.0038
Chua(y)	0.9999	1.1090	1.1187	1.1653	1.1444	1.1424	1.1011	1.0757
Chua(z)	1.0000	1.0915	1.1287	1.1070	1.0723	1.0321	0.9928	0.9568

2) Image substitution algorithm

Although the above mentioned smallest granularity bit permutation algorithm can change pixel value and position and can produce good encryption entropy, only permutation isn't security enough according to document [12]. And it can't resist differential attack. The above mentioned steps followed by pixel substitution process to enhance the ability of resistance differential attacks. Basic idea is: imitated the operating mechanism of chaos stretch and fold, first put the pixel that want to substitute into a larger computing space, made the present pixel operate with the front encrypted pixel as many as possible to enlarge the orbit recursive periodic, at last, fold back to the pixel space. The proposed image substitution algorithm has been performed in the following steps:

Step 1. Given the seed element, and selected the suitable chaos.

Step 2. Let E_i,E_{i+1} denote the i^{th} and $i+1^{th}$ encrypted elements, P_{i+2} denote the sequence number i+2 element which is expected to be encrypted, turned E_iE_{i+1} into binary serial and contacted them, made them as the input value of hash function, if the hash function used the MD5 algorithm, the output length is 32 hexadecimal string, then converted them into decimal and as the input parameter of chaos equation, iterated the chaos equation and turned it into uint64 type, operated this value with the pixel which wanted to encrypt, modular the result into pixel space.

Initialvalue= F(Hash(dec2bin(E_i, E_{i+1})));
W=Chaosfunction(initialvalue);
E_{i+2} =Mod(bitxor(uint64(p_{i+2}),w),256).

Principal function description:

Hash(): hash function.
Chaosfunction(): chaotic map which is used to produce chaos sequence.
F(): turned the 32 hexadecimal string which is the output of hash function into decimal number.
Step 3. Repeat the above-mentioned steps until the last element.
Step 4. Turn the one-dimensional array into two-dimensional matrix.
Further description for the procedure:

① For the first element to be encrypted, w value is given by user, for E0, that is hexadecimal string which has the length 8 given by user.

② Method for converting array elements into binary sequence.

$p^t(i,j)$ denotes the t^{th} digit that turned the array element $p(i,j)$ into binary sequence. One can turn the gray image into bit matrix by formula (3).

$$p^t(i,j)=\begin{cases}1 \ \ if \ \ (p(i,j)/2^t)\bmod 2=1\\ 0 \ \ others\end{cases} \tag{3}$$

On the contrary, using formula (4), one also can turn the $p^t(i,j)$ into $p(i,j)$:

$$p(i,j)=\sum_{t=0}^{7}2^t\times p^t(i,j) \tag{4}$$

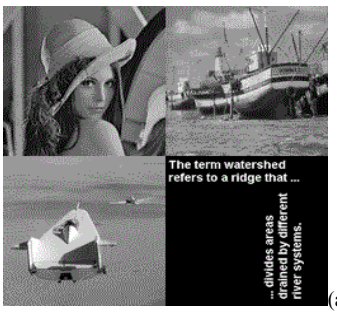
3) Decryption algorithm

Step 1. Given the same initial value, turned the encrypt image into one-dimensional array, decrypted elements from the last one to the first one, which can accomplish the reverse operating of image substitution,

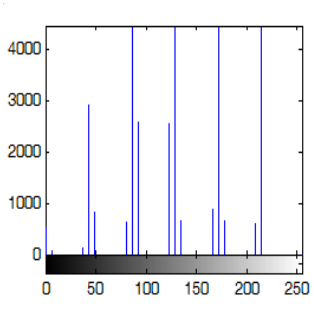
Step 2. Similar to image bit permutation algorithm, the difference is substituting the i^{th} column bit value with the t_i column, where i is from 1 to m.

4. Experimental results

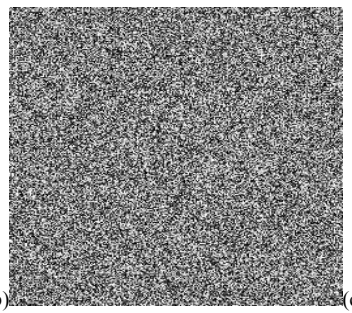
Simulations have been performed on a PC matlab platform to verify the validity of the proposed encryption technique. By calculating chaos permutation information entropy among Logistic, Henon, Lorenz and Chua, permutation information entropy regarding the foregoing mentioned chaotic map (see Table Ⅰ) can be gotten. From the experiment results, one can know that component x of Lorenz dynamic system has the maximum permutation entropy. Based on this fact, component x of Lorenz dynamic system has been selected as the source dynamic to drive image bit permutation.



(a)



(b)


(c)

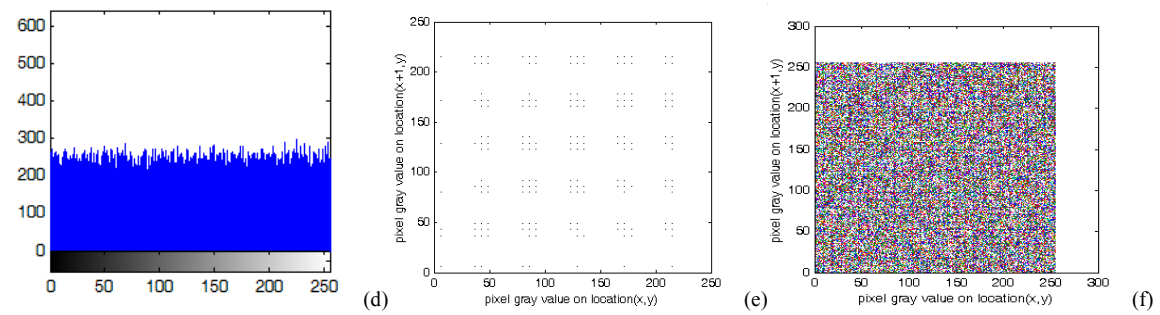


Fig.1. (a) plain-image, Lena, (b) histogram of the plain-image, (c) cipher-image used the proposed algorithm (d) histogram of cipher image (e) horizontal direction correlation of plain-image (f) horizontal direction correlation of cipher-image.

As far as choosing image is concerned, image (Fig. 1. (a)) is a synthesis of four images, including Lena picture, Boat picture, Text picture, Liftingbody picture, which almost concludes all attributes of image. However, there is not smooth transition at the junction among four pictures, this property don't have for natural image, increases the difficulty for image encryption.

Experimental results gotten by the proposed algorithm can be seen in Figure 1. Figure 1(b) and (d) indicate the histogram of plaintext and cipher-image respectively. Figure 1(c) is the cipher-image. Figure 1(e) and (f) show the horizontal direction correlation of the plaintext and cipher-image respectively.

5. security analysis

5.1. Analysis of key space

A security and reliable encryption algorithm should hold enough large key space to make brute-force attack infeasible. For encryption algorithm based on chaos map, parameters usually were used as key, which means the more parameters one algorithm has, the more keys it holds. Large parameters space can guarantee large key space. The proposed encryption algorithm included two steps: the first step is image bit permutation, key is chaos initial value and parameter, Lorenz chaos initial value: x0=10e-15, y0=10e-15, z0=10e-14. The second step is image bit substitution, the key is initial seed, E0 for hash function input, chaos parameter. In such large key space, there is no significant for exhaust attack.

5.2. Statistical analysis

Statistical analysis includes research of histogram and relevance of the adjacent elements.

4) Original image and cipher image digital characters

All kinds of digital characters of encrypted image can be seen from Table Ⅱ. From the experiment results, one can see that cipher information entropy, cipher mean, variance, and histogram for the proposed algorithm are better than those of Huang's and Ye's algorithms, these fact demonstrated that the proposed algorithm has better resistance attack ability.

TABLE II. IMAGE DIGITAL CHARACTER

		Proposed algorithm	Ye algorithm[8]	Huang algorithm[7]
Plaintext	Entropy	2.8573	2.8573	2.8573
Cipher	entropy	7.9976	7.9862	7.4714
Cipher	mean	127.8855	132.1122	119.3389
Cipher	variance	74.0174	74.2745	62.4324
Histogram	variance	14.7172	36.1985	254.9077

5) Correlation of adjacent pixels

The following procedures are carried out to test the correlation between two adjacent pixels. First, randomly selected 10,000 pairs of two horizontal (vertical, diagonal) adjacent pixels from an image and then calculated the correlation coefficient r_{xy}, of each pair using the following equations:

$$\mathrm{cov}(x,y) = E(x - E(x))(y - E(y)) \quad r_{xy} = \mathrm{cov}(x,y) / \sqrt{D(x)}\sqrt{D(y)}$$

$$r = mean\,(r_{xy})$$

where x and y are grey-level values of the two adjacent pixels in the image,

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i\ , \quad D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2\ ,$$

$$\mathrm{cov}(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y))\ .$$

Repeated the same process 1000 times and got the average value. Table Ⅲ lists the correlation coefficients of the studied images, and compared the proposed algorithm with Huang and Ye algorithm.

TABLE III. AVERAGE CORRELATION COEFFICIENT OF ADJACENT PIXELS

Different images	Horizontal direction	Vertical direction	Diagonal direction
Original image	0.8569	0.8665	0.8387
Proposed algorithm	-0.0147	0.0037	0.0333
Ye's algorithm[8]	-0.0353	0.0201	0.0375
Huang's algorithm[7]	0.0084	-0.0104	0.0349

From table Ⅲ, one can know that the adjacent correlation coefficient of original image is approach to 1. Comparing the proposed algorithm with Ye's and Huang's Algorithms, correlation of proposed algorithm is lower than that of Ye's algorithm and the same as the Huang 's algorithm. Therefore, the proposed algorithm possesses higher security against statistical attacks.

5.3. Differential attack

Two common quantitative measures, NPCR and UACI, are used to test the influence of changing a single pixel in the original image on the whole image encrypted by the proposed algorithm. NPCR stands for the number of pixels change rate while one-pixel of plain image is change. The unified average changing intensity (UACI) measures the average intensity of differences between the plain image and ciphered image. They are defined as follow [13]:

$$\mathrm{NPCR} = \frac{\sum_{i,j} D(i,j)}{W \times H} \times 100\% \qquad \mathrm{UACI} = \frac{1}{W \times H}\left[\sum_{i,j}\frac{|c_1(i,j) - c_2(i,j)|}{255}\right] \times 100\%$$

where c1 and c2 are two cipher-images whose corresponding plain-images have only one-pixel difference, the grey-scale values of the pixels at position (i , j) of c1 and c2 are denoted as c1(i, j) and c2(i, j), respectively; *W* and *H* are width and height of the cipher-image, respectively; D(i, j) is determined by c1(i, j) and c2(i, j), namely, if c1(i, j)=c2(i, j), d(i, j)=1; otherwise, d(i, j)=0.

A plain-image is first encrypted. Then, a pixel in that image is randomly selected and modified. The modified image is encrypted by using the same key so as to generate a new cipher-image. Finally the NPCR and UACI values are calculated. Their values are listed in Table Ⅳ.

TABLE IV. TWO COMMON MEASURES

	NPCR	*UACI*
Proposed algorithm	0.3493	0.1173
Ye's algorithm[8]	9.1553e-005	6.2232e-006
Huang's algorithm[7]	6.1035e-005	1.6037e-005

As can be seen from the table Ⅳ, the proposed algorithm is superior to Ye's and Huang's algorithms in NPCR, UACI . Therefore, the proposed algorithm have a better ability to resist differential attack than Ye's and Huang's algorithms.

6. Conclusion

Criterion for choosing the chaotic map among many chaos maps is proposed for image bit permutation algorithm. The proposed algorithm applied the selected chaos to accomplish image binary bit serial permutation based on unpredictability of chaos orbit, and realized the maximum degree permutation. Then, image bit substitution algorithm according to chaos map principle which means stretching and folding is designed to perform image encryption. This study analyzed the algorithm's key space, ability of resistance to differential attack and digital characteristics of cipher-image. Experiments demonstrated that the proposed algorithm has higher security than Ye's and Huang's algorithms and can protect digital image effectively.

7. Acknowledgment

The work described in this paper was supported by Liaoning doctoral funds under Grant No.20091034 and Liaoning higher education funds under Grant No. 2008T090.

8. References

[1] Ding Wei, Qi Dongxu. Digital image transformation and information hiding and disguising technology[J]. Chinese journal of computers, 1998, 21(9): 839-943.

[2] Zou Jiancheng, Li Guofu, Qi Dongxu. Generalized Gray code and its application in the scrambling technology of digital images[J]. Applied mathematics A, Journal of Chinese Universities, 2002, 17(3): 363-373.

[3] Qi Dongxu. Matrix transformation and its applications to image hiding[J]. Journal of north china university of technology, 1999, 11(1): 24-28.

[4] Baptista, M. S. Cryptography with chaos[J], Physics Letters A, 1998, Volume 240(1): 50-54.

[5] Chong Fu, Zhiliang Zhu. A chaotic image encryption scheme based on circular bit shift method[c]// The 9th International Conference for Young Computer Scientists. Zhang Jia Jie, Hunan, China, 2008: 3057 – 3061.

[6] Kocarev L, Jakimoski G, Stojanovski T, Parlitz, U. From chaotic maps to encryption schemes[c]// Proceedings of the 1998 IEEE International Symposium on Circuits and Systems 31 May-3 Jun 1998.

[7] C. K. Huang, H. H. Nien. Multi chaotic systems based pixel shuffle for image encryption[J]. Optics Communications, 2009, 282(11): 2123-2127.

[8] Guodong Ye. Scrambling encryption algorithm of pixel bit based on chaos map[J]. Pattern Recognition Letters, 2010, 31(5): 347-354.

[9] Yu Wanbo. Calculation experiment and analysis of chaos[M]. BeiJing: academic press,2008.

[10] Jose S. Cánovas, Antonio Guillamón. Permutations and time series analysis[J]. chaos, 2009, 19(4): 043103-1-043103-12.

[11] Bandt, C, Pompe, B. Permutation entropy-a natural complexity measure for time series[J], Phys. Rev. Lett. 2002, 88(174102): 1-4.

[12] Rhouma Rhouma, Safya Belghith. Cryptanalysis of a new image encryption algorithm based on hyper-chaos[J] Physics Letters A, 2008, 372 (38) :5973–5978.

[13] Chen G R, Mao Y B, Chui C K. A symmetric image encryption scheme based on 3D chaotic cat maps[J]. Chaos Solution and Fractals, 2004, 21(3): 749-761.