A Two-Dimensional Cellular Automata Based Method for Multiple Image Encryption

Ping Ping
College of Computer and
Information Engineering
Hohai University
Nanjing, China
e-mail: pingpingnjust@163.com

Wang Zhi-jian
College of Computer and
Information Engineering
Hohai University
Nanjing, China
e-mail: zhjwang@hhu.edu.cn

Feng Xu
College of Computer and
Information Engineering
Hohai University
Nanjing, China
e-mail: xufeng@hhu.edu.cn

***Abstract*—This paper presents a novel CA-based multiple image encryption by using a kind of two-dimensional (2-D) reversible CA, and by using a circular chaining mode of operation. The proposed method allows images to be processed in a 2-D way and makes the statistical information of each plain image in the group hidden in all cipher images. The results of computer simulations and security analyses show the good performance of our proposed method.**

Keywords- Cryptography; Image encryption; Two-dimensional cellular automata

I. INTRODUCTION

In recent years, more and more digital images transmitted over communication media, such as computer network, mobile phones, cable TV, etc. Protection of images against of illegal copying and distribution, especially in the fields of military, economy, and iatrology, has become extremely important. Most conventional ciphers, such as DES, AES which consider plaintext as one-dimensional data stream are not suitable for image encryption. Therefore it is of great interest to find efficient image encryption algorithms.

Cellular automaton (CA) introduced by John Von Neumann has their inherent properties like simple regular structure, massive parallelism, local interconnection, random-like behavior, which make it a good candidate to design image cryptosystems. Many works based on CA have been reported in the literatures. In [1], 1D hybrid CA is used to produce the bit stream of a key and furthermore a Vernam cipher based image encryption process is proposed. In [2], Madjarova et al. suggest an image cipher using a special kind of irreversible 2D CA with toggle rules. Then, Chen et al. [3-5] have developed several image encryption schemes based on 2-D recursive CAs. Recently, a well-known 2-D CA called “Game of Life” is used to design image ciphers in [6] and [7].

On the other hand, the Multiple image encryption (MIE) algorithms[8-10] have received much attention in recent years. The advantage of MIE is that it can encrypt many images at a time. In this paper, we propose a novel multiple image encryption method by using a 2-D reversible CA and employing a circular chaining mode of operation together. The proposed method allows images to be processed in a 2-D way and makes the statistical information of each plain image in the group hidden in all cipher images. Experiment results show that the suggested method is robust and secure.

II. TWO-DIMENSIONAL REVERSIBLE CELLULAR AUTOMATON

Cellular automata are abstract dynamical systems in which state, space and time are discrete. A 2-D CA is defined as 2-D lattice of cells, each of which can take a finite number of discrete states, updated synchronously in discrete time steps, according to a local rule. In this paper, we concentrate on a two-dimentional reversible CA (2-D RCA) whose finite state set is {0, 1} and local rules are defined by:

$$s_i^{t+1} = f(s_{i,j+1}^t s_{i-1,j}^t s_{i,j}^t s_{i+1,j}^t s_{i,j-1}^t) \oplus s_i^{t-1}. \qquad (1)$$

where $s_{i,j}^t$ denotes the state of cell (*i*, *j*) at time step *t*, and *f* is a Boolean function that gives the new state of a cell in terms of the current states of all cells in its neighborhood. From Eq. (1), it follows that the local rule *f* can be expressed in the form of a lookup table by specifying the values in the 2-bit truth table with 2^5 entries. One example of the rule defined by Eq. (1) is shown in Table I. In the table, the outputs of column 2 and column 3 complement each other. The decimal version of all output bits of column 2 is referred to as the rule number.

The set of states of all CA cells at time step t is called the configuration $C^{(t)}$ .At each time step, a configuration is transformed into a new configuration by applying the local rule f to every cell of a CA. This transformation can also be defined by a global transition function: $F : C^{(t)} \rightarrow C^{(t+1)}$. For finite size CA, boundary condition is imposed. Values of the boundary cells are usually all zero, periodic or randomly chosen. In this paper, the periodic boundary condition is considered.

TABLE I. EXAMPLE OF A RULE TABLE FOR 2-D REVERSIBLE RCA

$s_{i,j+1}^t s_{i-1,j}^t s_{i,j}^t s_{i+1,j}^t s_{i,j-1}^t$	$s_{i,j}^{t+1}$	
	$s_{i,j}^{t-1} = 0$	$s_{i,j}^{t-1} = 1$
00000	1	0
00001	0	1
00010	1	0
00011	1	0
00100	0	1
…	…	…
11111	0	1

III. Multiple Image Encryption Algorithm

2-D RCA can be regarded as an information processing unit, as shown in Fig. 1. Here, two initial configurations of the 2-D RCA $C^{(0)}$, $C^{(1)}$ are treated as the inputs, while the outputs are two final configurations $C^{(t)}$, $C^{(t+1)}$ which are obtained after several forward iteration of 2-D RCA by applying a rule f. We refer to this processing unit as an elementary block of our cryptosystem.

With the elementary block, our 2-D RCA based encryption structure for one round is depicted in Fig. 2. $I_1, I_2, ..., I_P$ are a group of plain images with size $M \times 2N$ pixels and P is the number of images in the group (Note that images in a group must be the same size, if not, padding or splitting technologies can be applied). $C_1, C_2, ..., C_P$ are P encrypted images for one round. The subkeys $f_1, f_2, ..., f_P$ for different sub-round encryption are generated by a PRNG with a secret seed K.

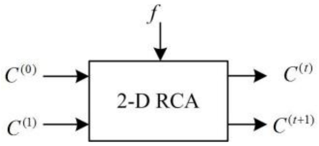
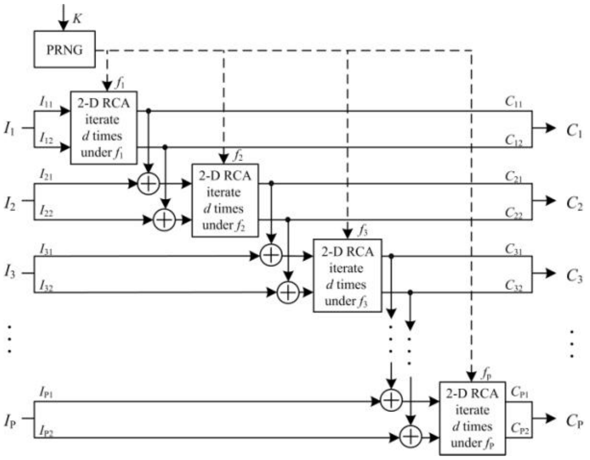


Figure 1. Regarding the 2-D RCA as an information processing unit



K
PRNG
f1
f2
f3
fP
2-D RCA iterate d times under f1
2-D RCA iterate d times under f2
2-D RCA iterate d times under f3
2-D RCA iterate d times under fP
I1
I11
I12
I2
I21
I22
I3
I31
I32
IP
IP1
IP2
C11
C12
C1
C21
C22
C2
C31
C32
C3
CP1
CP2
CP

Figure 2. Block diagram of the first round encryption for multiple images.

A. Encryption

The encryption process involves two-round encryption for P images. First, image I_i with size $M \times 2N$ pixels is divided into two blocks with size $M \times N$ pixels. Then, two blocks are coded as $M \times (w \times N)$ -bit 2-D RCA configurations I_{i1}, I_{i2} respectively, w is the number of bits per pixel. Next, the encryption is given as follows:

(1) The first round

Step 1: Take I_{11}, I_{12} as two initial configurations $C^{(0)}, C^{(1)}$. By applying rule f_1, the 2-D RCA iterates d times and generates two final configurations $C^{(d)}, C^{(d+1)}$. Then, let $C_{11} = C^{(d)}, C_{12} = C^{(d+1)}$.

Step 2: Take $I_{21} \oplus C_{11}$, $I_{22} \oplus C_{12}$ as two initial configurations $C^{(0)}, C^{(1)}$. By applying rule f_2, the 2-D RCA iterates d times and generates two final configurations $C^{(d)}, C^{(d+1)}$. Then, let $C_{21} = C^{(d)}, C_{22} = C^{(d+1)}$.

Keep on doing…until

Step P: Take $I_{P1} \oplus C_{(P-1)1}$, $I_{P2} \oplus C_{(P-1)2}$ as two initial configurations $C^{(0)}, C^{(1)}$. By applying rule f_P, the 2-D RCA iterates d times and generates two final configurations $C^{(d)}, C^{(d+1)}$. Then, let $C_{P1} = C^{(d)}, C_{P2} = C^{(d+1)}$.

(2) The second round

Let $I_{11} = C_{P1}$, $I_{12} = C_{P2}$, $I_{21} = C_{11}$, $I_{22} = C_{12}$, $I_{31} = C_{21}$, $I_{32} = C_{22}, \ldots, I_{P1} = C_{(P-1)1}, I_{P2} = C_{(P-1)2}$, and then repeat the first round encryption, where rule $f_1{}', f_2{}', ..., f_P{}'$ are applied. Finally, we obtain the cipher images $C_1{}', C_2{}', ..., C_P{}'$ by merging two blocks $C_{i1}{}', C_{i2}{}'$ into one and converting it into a pixel matrix.

B. Decryption

The decryption algorithm is similar to the encryption one and is the reverse process of encryption. The cipher images are decrypted one after another in a sequence of $C_P, C_{P-1}, ..., C_1$ with the same subkeys in reverse order. The proposed method can also be used for encrypting a single image by dividing them into several parts with the same size.

C. Key generation

In our scheme, the ciphering key is a set of 2-D RCA rules $\{f_1, f_2, ... f_P, f_1{}', f_2{}', ..., f_P{}'\}$, which are applied successively and expected to be completely random. But it may prove very difficult to manipulate such long key in practice implementation. Consequently, a pseudo-random number generation (PRNG) is used in our scheme to obtain the ciphering key. The PRNG initialized with the secret seed K generates random bit streams for "rule" bits. Likewise, on the decryption end identical PRNG initialized with the same secret seed K generates the same key streams. The advantage of such key generation scheme is that the relatively short secret seed is extended to the sufficient long key streams, and both the communicators only need to share a short key, which is more convenient. The following experiments assume that there has existed a secure PRNG, as the design of PRNG is out of the scope of this paper.

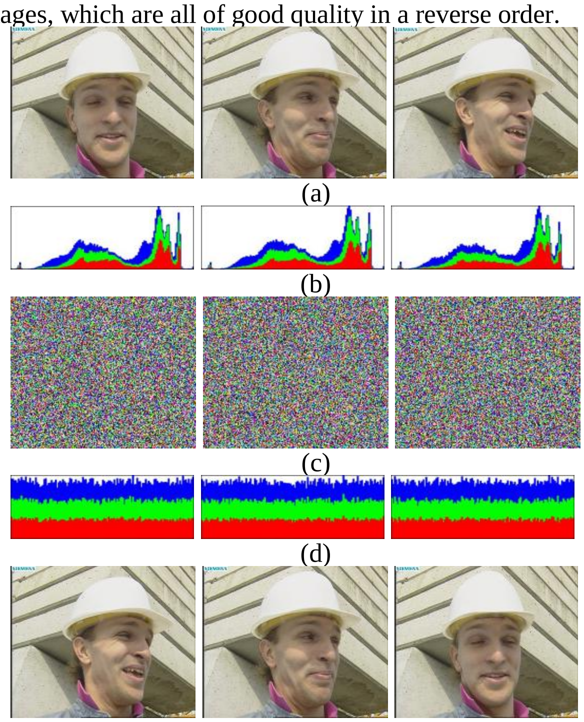
D. Selection of iteration number

The iteration number d has to be determined before encryption in our method. Consider two initial configurations $C^{(0}$; $C^{(1)}$ with $M \times N$ bits ($M \le N$), the necessary iteration number d_n is estimated by the equation:

$$d_n = \begin{cases} N - \lfloor (N-M+2)/2 \rfloor & \text{if } N \text{ is odd,} \\ N - \lfloor (N-M+1)/2 \rfloor & \text{if } N \text{ is even.} \end{cases} \qquad (2)$$

IV. COMPUTER SIMULATIONS

Extensive computer simulations have been done with Mathematica 8.0 to validate the proposed method. Fig.3(a) shows three 24-bit color images I_1, I_2, I_3 with 144×176 pixels. The iteration number d is 1000. By virtue of the encryption algorithm described in section 3, three plain images are well covered by the key in three cipher images C_1, C_2, C_3, as shown in Fig.3(c). From the histograms, one can see that the histograms of the cipher images Fig.3(d) are fairly uniform and are significantly different from that of the plain images Fig.3(b). Once the cipher images and the same key are available on the receiving side, the three plain images are well recovered. Fig.3(e) shows three decrypted images, which are all of good quality in a reverse order.



(a)

(b)

(c)

(d)

(d)

Fig. 3 Experimental results for images encryption and decryption. (a) Three plain images I_1, I_2, I_3. (b) Histograms of three plain images. (c) Three cipher images C_1, C_2, C_3. (d) Histograms of three cipher images. (e) Three decrypted images

V. SECURITY AND PERFORMANCE ANALYSIS

A. Key Space Analysis

The key space refers to the set of all possible keys that can be used in the cipher system. In our proposed encryption algorithm, the relatively short key (secret seed) shared by the transmitter and receiver is expanded to form the sufficient long keys (2-D RCA rules) by a secure PRNG. This would make different rules are used in each sub-round encryption. So, the key space κ is given by

$$\kappa = (2^{2^5})^{2P}, \qquad (3)$$

where P is the number of images. If $P = 2$, then $\kappa = 2^{128}$ which is sufficient large to prohibit exhaustive search of the key space. Additionally, the secret seed should be equipped with appropriate length to make brute-force attack completely infeasible and be changed often by the user.

B. Correlation of Two Adjacent Pixels

In general, adjacent pixels of most natural images are highly correlated. An effective encryption scheme should produce the ciphered images with sufficiently low correlation of adjacent pixels. To test the correlation between horizontally, vertically, and diagonally adjacent pixels in the image, the following procedure was carried out. First, we select N pairs of two adjacent pixels from an image. Then, we calculate the correlation coefficient by using the following formulas[11]:

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i, \quad (4)$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))^2, \ (5)$$

$$\operatorname{cov}(x,y) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))(y_i - E(y)), \qquad (6)$$

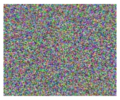
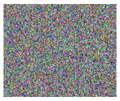
$$\gamma_{xy} = \frac{\operatorname{cov}(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}, \qquad (7)$$

where x and y represent gray values of two adjacent pixels in the image.

In Table 2, the correlation coefficients are shown for the plain images I_1, I_2, I_3 in Fig3. (a) and the ciphered images C_1, C_2, C_3 in Fig3. (c). It is clear that the cipher images obtained from the proposed method have very small correlation coefficients and hence the proposed method has a good ability of diffusion and confusion.

TABLE II. THE CORRELATION COEFFICIENTS OF ADJACENT PIXELS (N=20000)

Image		correlation coefficient		
		Horizontal	Vertical	Diagonal
Plain image I_1	R	0.910935	0.927921	0.904955
	G	0.927069	0.946646	0.922962
	B	0.934856	0.948418	0.925147
Plain image I_2	R	0.907943	0.927049	0.898946
	G	0.925826	0.945683	0.920498
	B	0.935225	0.949610	0.925479
Plain image I_3	R	0.904697	0.921751	0.889037
	G	0.922419	0.939153	0.910794
	B	0.932695	0.944234	0.920755

Cipher image C_1	R	-0.009542	-0.010005	-0.016255
	G	-0.005200	0.009350	-0.000488
	B	-0.006233	0.011485	-0.003374
Cipher image C_2	R	-0.007593	-0.010428	0.002292
	G	-0.002094	-0.002409	0.004744
	B	0.005451	-0.010809	-0.003936
Cipher image C_3	R	0.003868	0.001975	-0.008834
	G	0.003689	0.000459	0.008206
	B	-0.007540	-0.012087	0.013055

C. *Differantial Analysis*

Generally, the attacker can make a slight change of the plain image, and then observes the change of the result. Thus, he may find out a meaningful relationship between the plain image and the ciphered image. If one minor change in the plain image can cause a significant change in the ciphered image, with respect to diffusion and confusion, then this differential attack would become very inefficient and practically useless.

To test the influence of one image pixel change on the whole image encrypted by the proposed method, two common measures were used: NPCR (number of pixels change rate) and UACI (unified average changing intensity)[11]. They are defined as follows:

$$NPCR_{R,G,B} = \frac{\sum_{i,j} D_{R,G,B}(i,j)}{W \times H} \times 100\%, \qquad (8)$$

$$UACI_{R,G,B} = \frac{1}{W \times H} \sum_{i,j} \frac{|C_{R,G,B}(i,j) - C'_{R,G,B}(i,j)|}{2^{L_{R,G,B}} - 1} \times 100\%, \ (9)$$

where W and H are the width and height of the ciphered image. $C_{R,G,B}(i,j)$ and $C'_{R,G,B}(i,j)$ are the values of the corresponding color component red(R), green(G), blue(B) in two ciphered images, respectively. $L_{R,G,B}$ is the number of bits used to represent the red, green or blue channels of the image. The 2-D matrix $D_{R,G,B}(i,j)$ is determined by the following rule: if $C_{R,G,B}(i,j) = C'_{R,G,B}(i,j)$, then $D_{R,G,B}(i,j) = 0$; otherwise, $D_{R,G,B}(i,j) = 1$.

Tests are carried out with three plain images by using the above-mentioned formulas. The average values of NPCR and UACI thus obtained for all three images are shown in Table 3. It is clear that the values of NPCR and UACI are very close to the expected values, so the proposed method is good at resisting differential attack.

TABLE III. TABLE 4 SENSITIVITY TO PLAIN IMAGE

	Average NPCR (%)		
Plain image I_1	R :0.9961934	G :0.9961778	B:0.995926
Plain image I_2	R :0.9960646	G :0.9962916	B:0.996193
Plain image I_3	R :0.99605288	G :0.9961178	B:0.996176
	Average UACI (%)		
Plain image I_1	R :0.334105	G :0.334747	B:0.334353
Plain image I_2	R :0.334846	G :0.333994	B:0.334358
Plain image I_3	R :0.335518	G :0.334814	B:0.334373

ACKNOWLEDGMENT

This research was supported by the National Natural Science Foundation of China under Grant number 61272543, and by the Natural Science Foundation of Jiangsu Province under Grant number BK20130852, BK2012807.

REFERENCES

[1] A. M. del Rey, “A novel cryptosystem for binary images,” Studies in Informatics and Control, vol.47, 2004, pp.5–14.

[2] M. Madjarova, M. Kakuta, T. Obi, M. Yamaguchi, N. Ohyama, “Optoelectronic block-cipher based on iteration of the 2-d toggle cellular automata: algorithm,” Optical Review, Vol.6, 1999, pp.110–7.

[3] R. J. Chen, J. L. Lai, “Data encryption using non-uniform 2-d von neumann cellular automata”, in: Proceeding of IEEE 9th International Workshop on Cellular Neural Networks and Their Applications, 2005, May, pp.77–80

[4] R. J. Chen, J. L. Lai, “Image security system using recursive cellular automata substitution”, Pattern Recognition, Vol.40, 2007, pp.1621–1631

[5] R. J. Chen, S. J. Horng, “Novel scan-ca-based image security system using scan and 2-d von neumann cellular automata,” Signal Processing:Image Communication, Vol.25, 2010, pp.413–426.

[6] X. Wang, C. Jin, “Image encryption using game of life permutation and PWLCM chaotic system”, Optics Communications, Vol.285, 2012, pp.412–417.

[7] J. Machicao, A. G. Marco, O. M. Bruno, “Chaotic encryption method based on life-like cellular automata”, Expert Systems with Applications, Vol.39, 2012, pp.12626–12635.

[8] Q. H. Lin, F. L. Yin, T. M. Mei, H. L. Liang H, “A blind source separation-based method for multiple images encryption,” Image Vision Comput, Vol.26, 2008, pp.788-798.

[9] T. H. Chen, K. C. Li, “Multi-image encryption by circular random grids,” Information Sciences, Vol.189, 2012, pp.255-265.

[10] D. Z. Kong, X. J. Shen, “Multiple-image encryption based on optical wavelet transform and multichannel fractional Fourier transform,” Optics & Laser Technology, Vol.57, April, 2014, pp.343-349.

[11] H. Liu, X. Wang, “Color image encryption using spatial bit-level permutation and high-dimension chaotic system,” Optics Communications, Vol.284, 2011, pp.3895-3903.