Reverse Iterative Image Encryption Scheme Using 8-layer Cellular Automata

Xing Zhang[1], Hong Zhang[1], and Chungen Xu[2]
[1]School of Computer Sciences and Engineering, Nanjing University of Science & Technology, Nanjing, China
[2]School of Science, Nanjing University of Science & Technology, Nanjing, China
[E-mail: xingguang89@yeah.net; zhhong@njust.edu.cn; xuchung@njust.edu.cn]
*Corresponding author: Xing Zhang

Received January 11, 2016; revised May 30, 2016; accepted June 4, 2016; published July 31, 2016

Abstract

Considering that the layered cellular automata (LCA) are naturally fit for representing image data in various applications, a novel reverse iterative image encryption scheme based on LCA is proposed. Specifically, the plain image is set as the final configuration of an 8-layer CA, and some sequences derived from a random sequence are set as the pre-final configuration, which ensure that the same plain image will never be encrypted in the same way when encrypted many times. Then, this LCA is backward evolved by following some reversible two order rules, which are generated with the aid of a newly defined T-shaped neighborhood. The cipher image is obtained from the recovered initial configuration. Several analyses and experimental results show that the proposed scheme possesses a high security level and executive performance.

Keywords: Image encryption; layered cellular automata; T-shaped neighborhood; two order rule; reverse iteration;

This research was supported by two research grants from the Natural Science Foundations of Jiangsu Province of China [Grant No. BK20141405 and BK20131353].

http://dx.doi.org/10.3837/tiis.2016.07.029 **ISSN : 1976-7277**

1. Introduction

Because image data are widely used in multimedia applications, image security has become indispensable and crucial in communication and storage. The cryptographic technique is an essential component of any secure communications that ensure data confidentiality, authentication, integrity and non-repudiation. However, typical encryption schemes, such as AES, DES and RSA, are not suitable for image encryption because of the redundancy, bulk data capacity and the high correlation between pixels of image data. In recent years, many image encryption schemes have been put forward, using chaos theory, the permutation and diffusion technique, DCT, wavelet transform and so on. Some typical image encryption methods and schemes that include chaos-based cryptosystem, quantum encryption technique, bio-chaotic encryption schemes and some AES-based image encryption algorithms are studied and analyzed in [1]. In addition to these methods, applying cellular automata (CA) to design an image encryption scheme is also an attractive idea in theory, owing to its inherent features, such as parallelism, locality and homogeneity.

In past decades, numerous works based on CA have been reported in the literature [2-20]. In [4] and [5], a special type of one-dimensional (1D) CA with unity attractors, which can perform an encrypting function to transform pixel values, was utilized. Because an image is a two-dimensional (2D) pixel matrix rather than 1D numerical data, 2D CA will be more suitable for image encryption [3][7][9][19]. However, a fly in the ointment is that each pixel value should be converted into binary when processed by computer; thus, an image is not merely a simple 2D matrix and the binary representation of an image looks like a cube with 0's and 1's embedded within the cube rather than its sides. When crosscutting this cube, we will find that each cross section can be regarded as a 2D CA. This tells us that a layered CA (LCA), i.e., a stacked number of 2D CA, should be fitter for representing the gray image than a 2D CA. The LCA can be viewed as a highly parallel system and the encryption schemes based on LCA are more efficient than other traditional secret key cryptosystems [21]. LCA was studied in [21][22] for generating normal random numbers, and then [23] proposed a new cryptosystem based on the LCA and reversible CA (RCA); the proposed scheme was compared with AES regarding various parameters and it was found to be on par with it. Recently, a new public key encryption scheme using LCA, which was proven secure against chosen-plain attacks and is also more efficient than RSA-1024, was designed in [24].

In this paper, a new gray image encryption method using 8-layer CA with reversible two order rules is proposed. Two order transition rules, which were derived from elementary CA (ECA) and update the state of each cell according to two previous steps, were employed in [2][3][9]. First, we generate some two order rules from ECA with a newly defined neighborhood structure. Then, built upon an 8-layer CA, we present a reverse iterative image encryption scheme, in which the binary form of a plain image is set as the final configuration of this CA and some processed random sequences are set as the pre-final configuration, and then backward evolve this LCA to recover its initial configuration, which is set as the binary form of the cipher image. Moreover, random XOR operations between different layers, which are conducive to improving the diffusion and confusion effects of this encryption scheme, are also utilized. Finally, security analysis and a performance test are also performed, with the results showing that the proposed scheme has an excellent statistical property and execution performance.

The remainder of this paper is organized as follows. In Section 2, we introduce some definitions of CA, LCA, and the method of constructing two order rules. In Section 3, we present our image encryption scheme based on LCA with a T-shaped neighborhood. Then, we analyze its strengths and prove its security in Section 4. Finally, we draw our conclusion in Section 5.

2. Preliminaries

2.1 Cellular Automata

A cellular automata is a discrete and dynamic system that consists of several cells located on a grid regularly; each cell has a finite number of states and updates its state depending on the states of the cells in its neighborhood according to a local rule in a discrete time step.

Formally, let s_i^t denote the state of the *i*-th cell at t time step and s_i^{t+1} denote its state at time $t+1$; f is the local transition rule and r is the neighborhood radius. Then, the state update can be represented as follows:

$$s_i^{t+1} = f\left(s_{i-r}^t, \cdots, s_{i-1}^t, s_i^t, s_{i+1}^t, \cdots, s_{i+r}^t\right) \tag{1}$$

The states of all cells in a CA at t time step $\left(s_0^t, s_1^t, \cdots, s_i^t, \cdots\right)$ are called the configuration of this CA at this time.

Elementary CA is the simplest 1D CA, in which each cell only has two possible states (0 or 1) and three neighborhood dependencies. The state transition of each cell can be represented as follows:

$$s_i^{t+1} = f\left(s_{i-1}^t, s_i^t, s_{i+1}^t\right) \tag{2}$$

There are $2^3 = 8$ possible configurations for each cell and its immediate neighbors and hence there will be $2^8 = 256$ possible rules.

A CA is said to be reversible if and only if every current configuration of the CA has not only one successor but also one predecessor. Reversible CA (RCA) is especially suitable for a cryptosystem because the reversibility property of CA ensures that any encrypted message can be decrypted by running the same system in reverse. Research has shown that for one-dimensional CA, there are known algorithms for deciding whether a rule is reversible or irreversible, while for two or higher dimensional CA, whether the rule is reversible is undecidable [25][26].

2.2 Two order rules

Analysis [27] has shown that only a small number of rules in ECA have the property of being reversible. For example, among all 256 one-radius ECA transition rules, only six are reversible. Thus, selecting these reversible rules as the key of a cryptosystem will reduce the security of the system. Focusing on this question, this paper attempts to generate some reversible rules from the ECA rules.

First, we will change the evolutionary direction of an ECA so that each cell backward updates its state in a discrete time step. That is, each cell gets its state at time *t*-1 from the states of itself and its neighbors in time *t* following a local rule f ; this procedure can be presented as:

$$s_i^{t\text{-}1} = f\left(s_{i-1}^t, s_i^t, s_{i+1}^t\right) \tag{3}$$

Second, further changing of the backward ECA is performed, which defines a T-shaped neighborhood structure. A backward CA with a T-shaped neighborhood implies that the state of a cell in the CA at time $t-1$ depends not only on the states of its neighborhood at time t but also on its state at time $t+1$; the evolution formula is

$$s_i^{t\text{-}1} = g\left(s_{i-1}^t, s_i^t, s_{i+1}^t, s_i^{t+1}\right) \tag{4}$$

where g is the two order rule because its input contains the states from two time steps. The reversibility of the two order rules has been demonstrated in [28].

Because each cell has two possible states, there are $2^4 = 16$ possible configurations of $\left(s_i^{t+1}, s_{i-1}^t, s_i^t, s_{i+1}^t\right)$ as the input of rule g, and correspondingly, there will be 2^{16} possible rules. Such rules can be named using the standard convention proposed by Wolfram; the rule name is the corresponding decimal value of the binary sequence, which is composed of the output of all possible configurations. Fig. 1 illustrates an example of such rules and is called rule T57630.

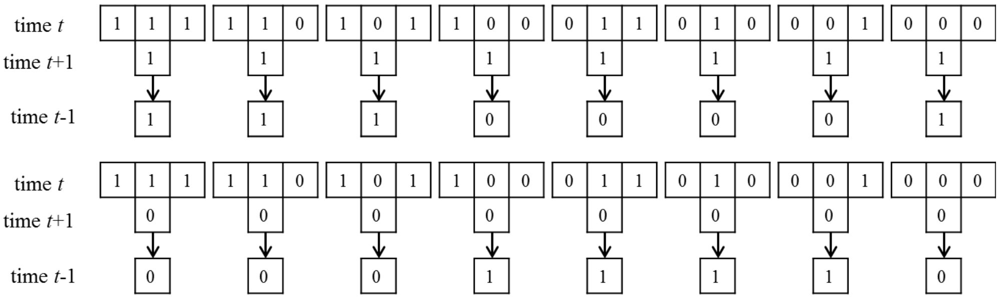
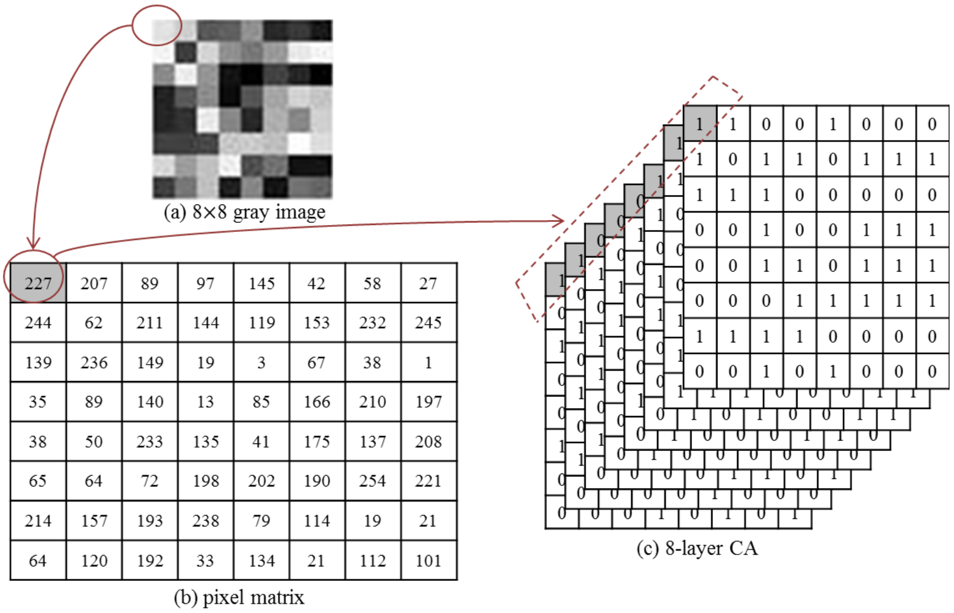


Fig. 1. Rule T57630

2.3 Layered Cellular Automata (LCA)

LCA can be viewed as a highly parallel system that consists of layers, with each layer being a 2D CA; moreover, each layer can also be seen as a composition of rows of 1D CA with the same size, and the number of layers can be changed according to the actual situation. The stacked structure gives not only LCA the inherent features of conventional CA but also a more complex and flexible neighborhood structure, which may lead to analysis of a new type of CA and is of much theoretical interest.


(a) 8×8 gray image

227	207	89	97	145	42	58	27
244	62	211	144	119	153	232	245
139	236	149	19	3	67	38	1
35	89	140	13	85	166	210	197
38	50	233	135	41	175	137	208
65	64	72	198	202	190	254	221
214	157	193	238	79	114	19	21
64	120	192	33	134	21	112	101

(b) pixel matrix

1	1	0	0	1	0	0	0
1	0	1	1	0	1	1	1
1	1	1	0	0	0	0	0
0	0	1	0	0	1	1	1
0	0	1	1	0	1	1	1
0	0	0	1	1	1	1	1
1	1	1	1	0	0	0	0
0	0	1	0	1	0	0	0

(c) 8-layer CA

Fig. 2. Correspondence between gray image and 8-layer CA

An image can be regarded as an integer matrix that is made up of the pixel value of each point. The basic idea of the proposed image encryption scheme is to change the pixel values. Each pixel value of the gray scale image ranges from 0 to 255 and can be converted into an 8-bit binary sequence, and we have found that the pixel matrix of an image can match up with an 8-layer CA completely if look out upon the binary pixel matrix (**Fig. 2**); thus, the LCA naturally applies to image processing.

3. Proposed Scheme

This section will present our proposed reverse iterative image encryption scheme. The binary sequences of the pixel values of the plain image are set as the final configuration of an 8-layer CA, in which each layer consists of rows of 1D CA with the same size and the size is defined by the size of the plain image. Then, the generated two order rule is applied to backward evolve this LCA many times to recover the initial configuration of the LCA; we will achieve the cipher image by converting this initial configuration into a pixel matrix. Moreover, according to the definition of the two order rule and backward iteration, a preceding configuration of the final configuration is needed and will be defined by means of some pseudo random numbers.

The proposed gray image encryption scheme is composed of three algorithms, namely, **KG**, **Enc** and **Dec**.

KG. Choose one two order rule f and generate a 128-bit pseudo random binary sequence (PRBS), denoted as r_0. Note that if the number of pixels of the image to be encrypted isn't 128, padding or splitting technologies can be applied. Then, choose a number $n \in N$ and randomly choose three numbers, denoted as n_1, n_2 and n_3, where $n_1 + n_2 + n_3 = n - 1$. Thus, the key of this scheme is (r_0, f, n_1, n_2, n_3).

Enc. There are five modules of this image encryption algorithm, namely, PRBS evolution, n_1 iterations, transposition, n_2 iterations and n_3 iterations, which are illustrated in **Fig. 3**.

1) PRBS evolution. We take the random sequence r_0 as the initial configuration of a one-dimensional (1D) CA and then randomly choose seven elementary transition rules to evolve this 1D CA once, with all seven corresponding new configurations denoted as $r_1, r_2, \cdots, r_7$.
2) n_1 iterations. In this module, we begin to encrypt a plain image with size $M \times H$ pixels.

- Step 1. Convert each pixel value that belongs to [0,255] into an 8-bit binary sequence and arrange them into an 8-layer CA, each layer having $M \times H$ bits. Let $P_0, P_1 \cdots, P_7$ denote the 8 layers, respectively.
- Step 2. Because the encryption algorithm is a reverse iteration for obtaining the cipher image, we set the binary sequence $P_i\,(0 \le i \le 7)$ as the final configuration of the *i*-th layer at n time step, denoted as P_i^n. Then, we apply the two order rules to generate the configuration at the $n-1$ time step P_i^{n-1} from the final configuration at n time step P_i^n and its pre-final configuration at the $n+1$ time step P_i^{n+1}. Now, we define $P_i^{n+1} = r_i \oplus P_{(i+1)\bmod 8}^n$, where $\oplus$ is the XOR operation.

- Step 3. We apply the transition rule f to each layer, along with the designated pre-final configurations for n_1 times, to get the new configurations. This procedure can be presented as follows:

$$\left(r_i \oplus P^{n}_{(i+1)\bmod 7}, P^{n}_i\right) \xrightarrow{f^{n_1}} \left(P^{n-n_1+1}_i, P^{n-n_1}_i\right) \tag{5}$$

where $P^{n-n_1}_i$ denotes the configuration of the i-th layer at the $n-n_1$ time step. We save all $P^{n-n_1}_i$ and their pre-configurations $P^{n-n_1+1}_i$ and then run the next encryption module.

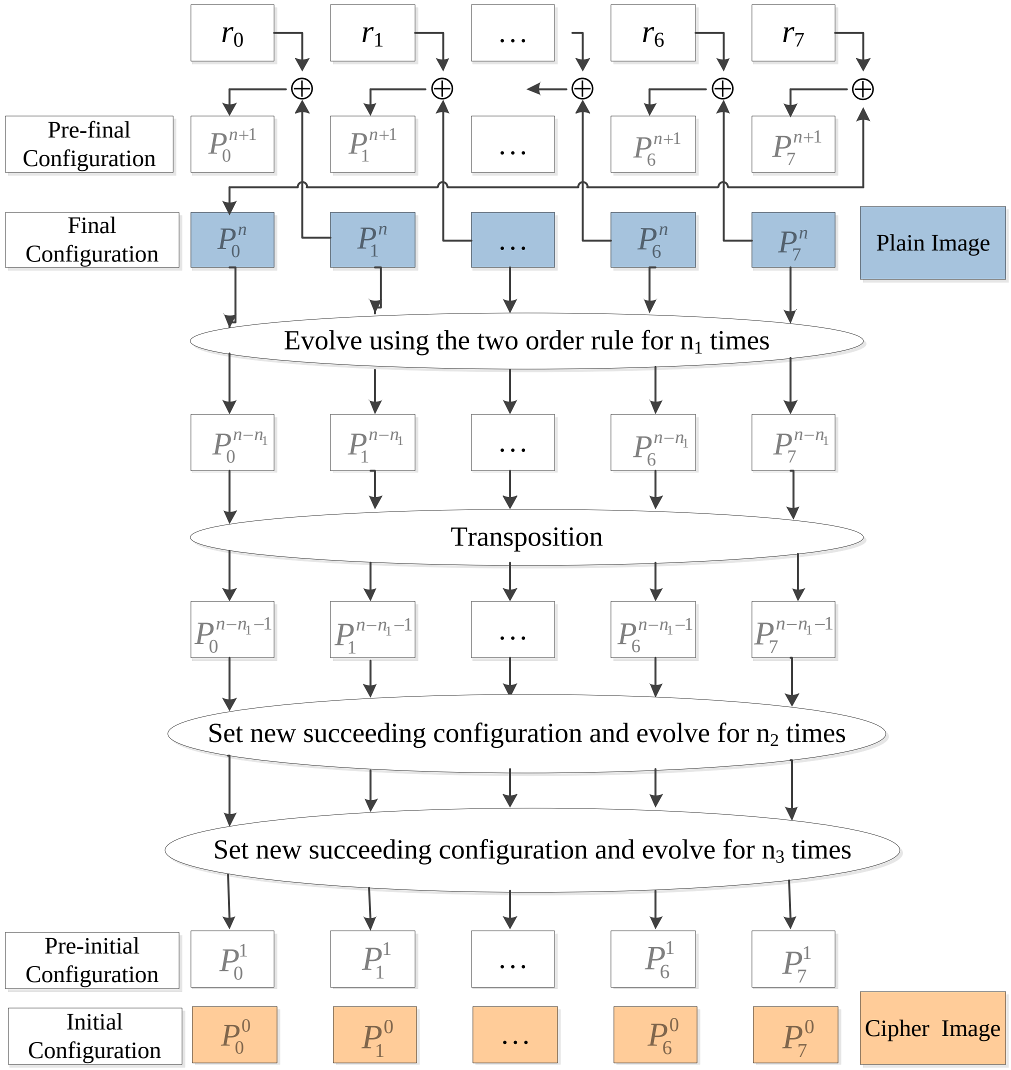


Fig. 3. Encryption procedure of a gray image

3) Transposition. Because each layer of the 4-layer CA consists of rows of 1D CA and the neighborhood of each cell is defined in the 1D CA, if we change the state of any cell, only the cells in the same row will be affected, i.e., the influence is limited. To obtain better a diffusion and confusion effect, we apply the transposition operation, which transposes rows and columns in each layer. In this module, we give a transposition of the

configuration $P_i^{n-n_1}$ achieved in the last encryption module, and the corresponding new configuration at time $n-n_1-1$ is denoted as $P_i^{n-n_1-1}$.

4) n_2 iteration. In this module, we continue to backward evolve the layered CA from the configuration $P_i^{n-n_1-1}$. Here, we compute $P_i^{n-n_1}\oplus P_{(i+n_1)\bmod 8}^{n-n_1-1}$ as the preceding configuration of $P_i^{n-n_1-1}$ and then apply the rule f to them for n_2 iterations, which can be represented as follows:

$$\left(P_i^{n-n_1}\oplus P_{(i+n_1)\bmod 8}^{n-n_1-1}, P_i^{n-n_1-1}\right)\xrightarrow{f^{n_2}}\left(P_i^{n-n_1-n_2}, P_i^{n-n_1-n_2-1}\right) \tag{6}$$

where $P_i^{n-n_1-n_2-1}$ is the new configuration after n_2 iterations. **Fig. 4** shows the concrete evolution of this module. We save $P_i^{n-n_1-n_2-1}$ and its preceding configuration $P_i^{n-n_1-n_2}$ and then proceed to the next module.

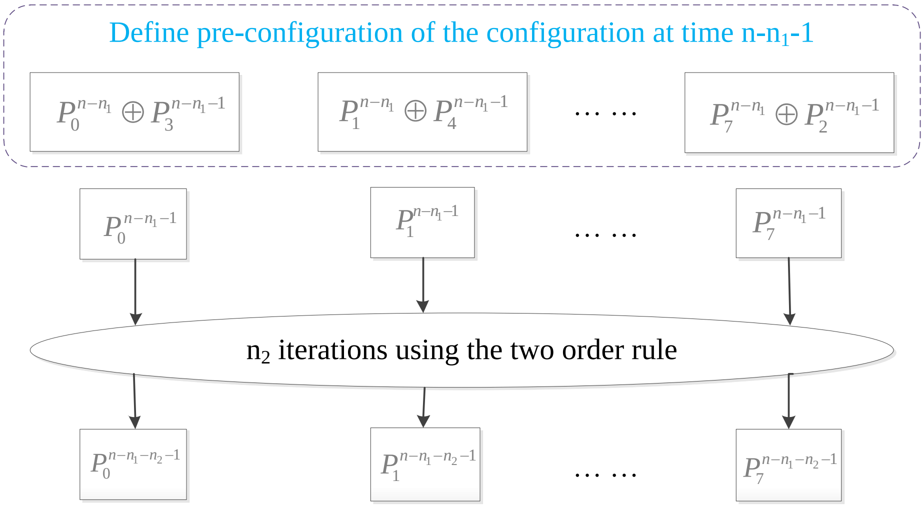

Define pre-configuration of the configuration at time n-n$_1$-1
$P_0^{n-n_1}\oplus P_3^{n-n_1-1}$
$P_1^{n-n_1}\oplus P_4^{n-n_1-1}$
… …
$P_7^{n-n_1}\oplus P_2^{n-n_1-1}$
$P_0^{n-n_1-1}$
$P_1^{n-n_1-1}$
… …
$P_7^{n-n_1-1}$
n$_2$ iterations using the two order rule
$P_0^{n-n_1-n_2-1}$
$P_1^{n-n_1-n_2-1}$
… …
$P_7^{n-n_1-n_2-1}$

Fig. 4. Concrete n$_2$ iterations with n$_1$=3

5) n_3 iterations. This is the last module of the encryption algorithm. In a similar way, we set $P_i^{n-n_1-n_2}\oplus P_{(i+n_2)\bmod 8}^{n-n_1-n_2-1}$ as the preceding configuration of the $P_i^{n-n_1-n_2-1}$ achieved in the last module. Because $n=n_1+n_2+n_3+1$, the configuration $P_i^{n-n_1-n_2-n_3-1}$, namely, P_i^0, generated after n_3 backward evolutions is the initial configuration of the 8-layer CA. This procedure can be presented as follows:

$$\left(P_i^{n-n_1-n_2}\oplus P_{(i+n_2)\bmod 8}^{n-n_1-n_2-1}, P_i^{n-n_1-n_2-1}\right)\xrightarrow{f^{n_3}}\left(P_i^1, P_i^0\right) \tag{7}$$

We can obtain the cipher image by merging all of the recovered initial configurations $P_i^0\left(0\le i\le 7\right)$ into one and converting it into a pixel matrix. The pre-configurations of the initial configurations P_i^1 are also kept as final data and will be used to perform the decryption algorithm.

Dec. The decryption algorithm is simply the inverse process of the encryption.

Each value in the pixel matrix of the cipher image is converted into a binary sequence and arranged into an 8-layer CA, which is regarded as the initial configuration of this CA. The decryption is a forward evolution of this layered CA by using the two order rule f; thus, a preceding configuration is needed, which is the work of the final data achieved in the encryption procedure. For the initial and pre-initial configurations, we successively run the n_3 iterations, n_2 iterations, transposition and n_1 iterations, and then the final configurations after $n-1$ iterations are combined and converted into the pixel matrix of the plain image. Moreover, the recovered configuration after n iterations is the preceding configuration defined during encryption that was generated by the PRBS and plain image.

4. Security Analysis and Performance Test

The characteristics of the image data, which dictate that the performance evaluation of the image encryption should be different from that of other data encryption algorithms, should consider not only the key space, sensitivity of the key and algorithm efficiency but also the information entropy, real-time, anti-interference, compressibility in the transmission, and so on. In a good encryption algorithm, the plain image information should be randomly distributed in the cipher image such that it's difficult for the attacker to extract significant information from the cipher image.

In this section, several experiments are performed to evaluate the performance of the proposed encryption algorithm.

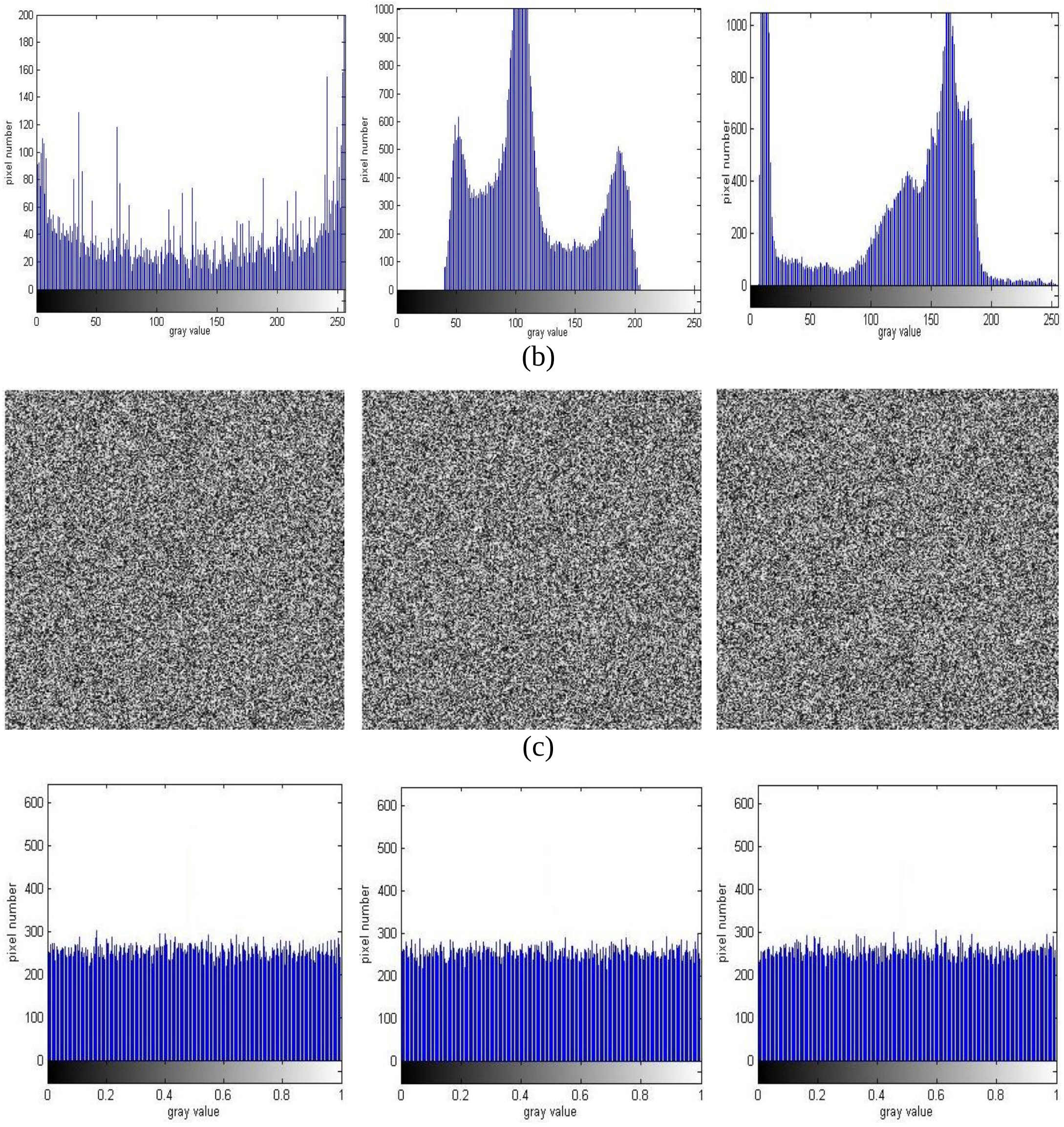
4.1 Key Space

The key space is the set of all possible keys that can be used in the encryption. The larger the key space is, the lower the possibility to guess the encryption key; hence, the encryption algorithm can accordingly be made secure against exhaustive attack. In our proposed encryption algorithm, the key is composed of a two order transition rule, a 128-bit pseudo random binary sequence and three integers. Thus, the key space is larger than $2^{2^4} \times 2^{128} = 2^{144}$, which is sufficiently large to withstand exhaustive attack. Moreover, using a different random sequence in each time encryption makes the same plain image have different cipher images when encrypted many times.

4.2 Histogram

An image histogram acts as a graphical representation of the distribution of pixel values in an image. Generally, the plain image pixel values are irregularly distributed, but the encrypted image must be similar to random ones and show a pseudo random distribution.

(a)



(b)

(c)

(d)

Fig. 5. Histograms of images, (a) three plain images: testpat1, rice and cameraman, (b) histograms of three plain images, (c) three cipher images, (d) histograms of three cipher images.

Fig. 5 illustrates the histograms of three different plain images and the corresponding cipher images, which are generated by the proposed encryption. It is obvious that the histograms of the cipher images are fairly uniform and are different from those of the plain images. Therefore, deriving information about the plain image from the cipher image is impossible without the key.

4.3 Correlation of Two Adjacent Pixels

Low correlation between image pixels indicates that the image has good diffusion and confusion properties. The correlation coefficient is defined as follows:

$$r_{xy}=\frac{\operatorname{cov}(x,y)}{\sqrt{D(x)}\bullet\sqrt{D(y)}} \tag{8}$$

$$D(x)=\frac{1}{n}\sum_{i=1}^{n}\left(x_i-E(x)\right)^2 \tag{9}$$

$$E(x)=\frac{1}{n}\sum_{i=1}^{n}x_i \tag{10}$$

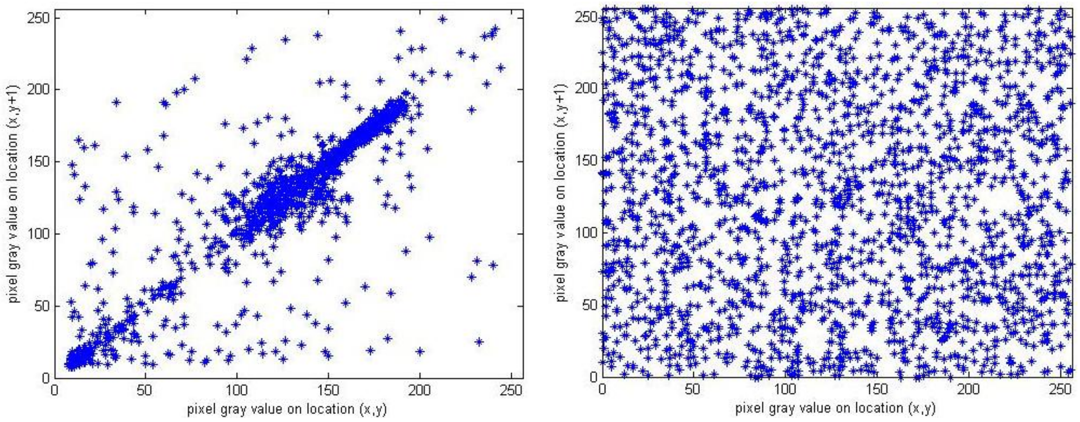
$$\operatorname{cov}(x,y)=\frac{1}{n}\sum_{i=1}^{n}\left(x_i-E(x)\right)\left(y_i-E(y)\right) \tag{11}$$

where x and y represent gray values of two adjacent pixels; r_{xy} is the correlation coefficient.

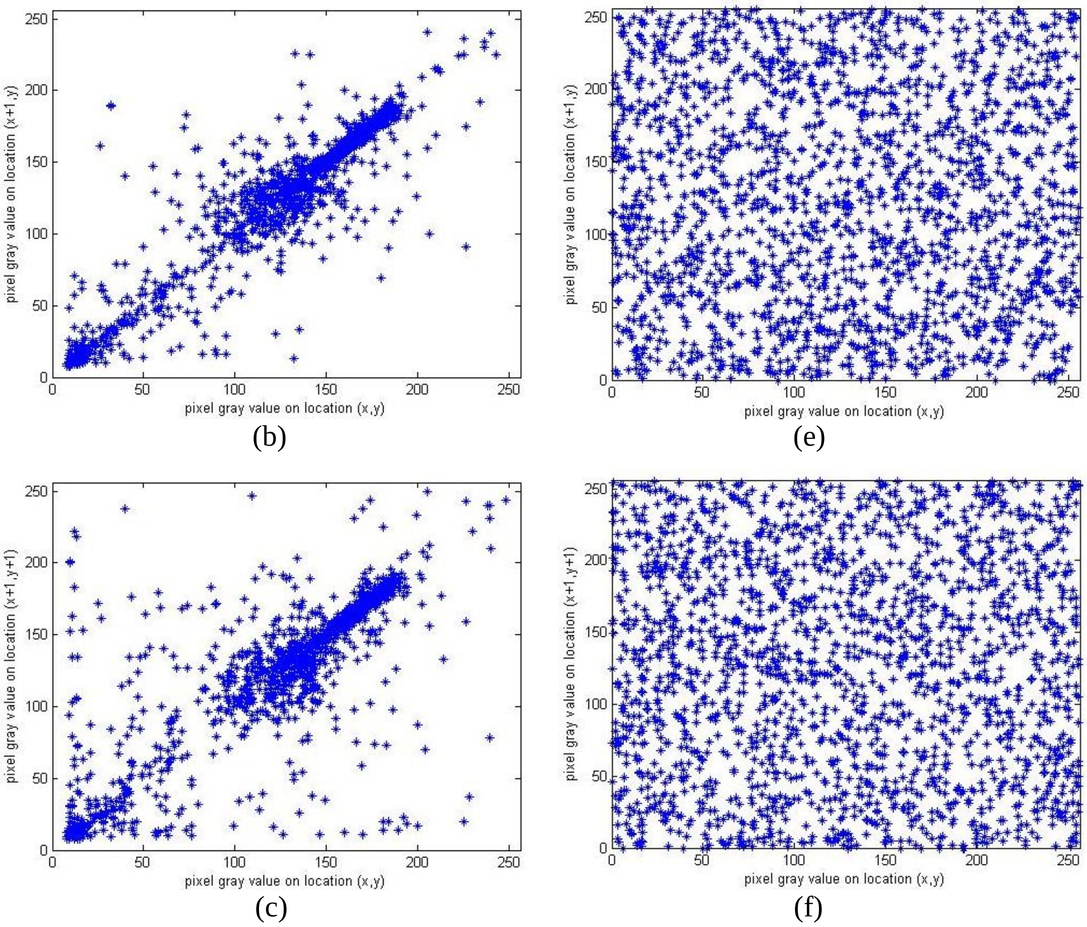
To test the correlation between horizontally, vertically and diagonally adjacent pixels in the image, taking three images and the cipher images for instance, we randomly choose 10000 pairs of adjacent pixels and calculate the corresponding correlation coefficient, which is shown in **Table 1**. **Fig. 6** illustrates the correlation distributions of the pixels extracted from three directions for image cameraman. From **Table 1** and **Fig. 6**, we can see that our proposed algorithm reduces the correlation between adjacent pixels effectively and hence has good diffusion and confusion properties.

Table 1. Correlation coefficients for horizontal, vertical and diagonal directions

Image	Plain image			Cipher image		
	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
testpat1	0.9146	0.9175	0.8979	0.0020	-0.0234	-0.0274
rice	0.9329	0.9137	0.8827	-0.0083	0.0023	0.0008
cameraman	0.9520	0.9377	0.9374	-0.0058	0.0191	0.0099



(a) (d)



(b) (e)

(c) (f)

Fig. 6. Correlation between horizontally, vertically and diagonally adjacent pixels: (a), (b) and (c) in the plain image, (d), (e) and (f) in the cipher image

4.4 Image Information Entropy

According to Shannon's theory, information entropy is a measure of unpredictability of information content. In an image encryption algorithm, entropy is one of the measurements of the gray value distribution and can be calculated by using the following formula:

$$H(I) = -\sum_i p(x_i)\log_2 p(x_i) \tag{12}$$

where I is an image, x_i denotes the i-th gray values, $p(x_i)$ is the probability of x_i in the image I, and $\sum_i p(x_i) = 1$.

Because the gray images are coded as 8-bit, the optimal entropy value is 8 and the entropy of a well encrypted image should be very close to this bound. Table 2 shows a comparison of the entropy values of three different plain images and their cipher images, which were encrypted by our proposed algorithm and six other existing encryption methods based on 1D CA (proposed in Refs. [2, 4, 14]) and 2D CA (proposed in Ref. [3, 7, 9]). It is clear that images encrypted by our proposed algorithm have near to the optimal entropies better than those of the other existing algorithms and thus have good random properties that prevent any statistical cryptanalysis attack because no significant information can be derived from the cipher image without the key.

Table 2. Comparison of entropies of plain/ cipher images for proposed and existing algorithms

Image	Plain image	Cipher image						
		Proposed	1D cellular automata			2D cellular automata		
			Ref. [2]	Ref. [4]	Ref. [14]	Ref. [3]	Ref. [7]	Ref. [9]
testpat1	2.4536	7.9971	7.9978	7.9357	7.9706	7.9979	7.9811	7.9886
rice	7.0115	7.9992	7.9987	7.9733	7.9811	7.9973	7.9893	7.9885
cameraman	7.0097	7.9989	7.9969	7.9781	7.9795	7.9981	7.9886	7.9889

4.5 Sensitivity Analysis

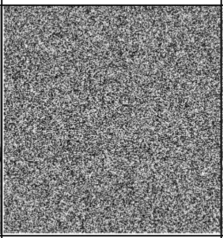
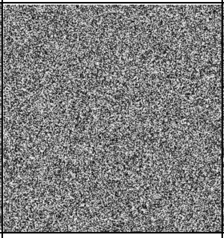
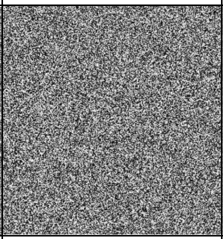
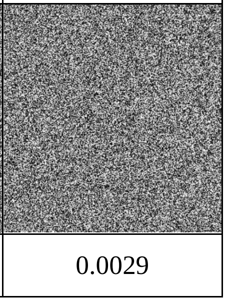
4.5.1 Key Sensitivity

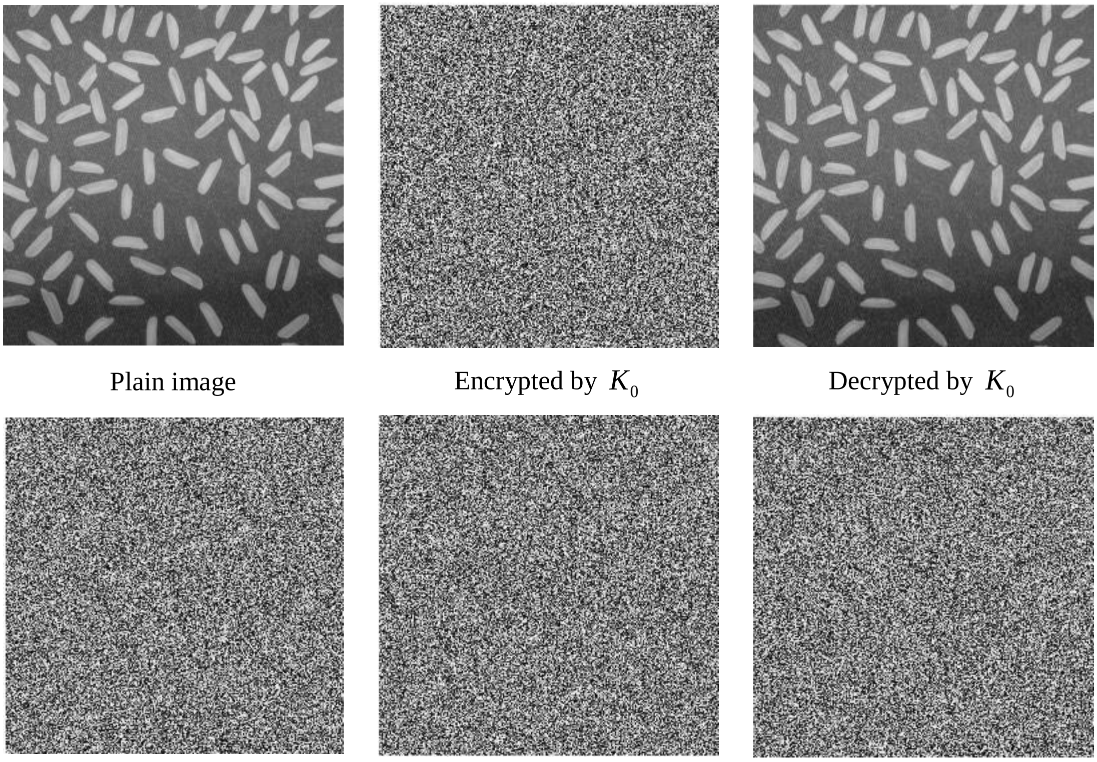
Key sensitivity indicates the change rate of the pixel values in the cipher image with only one bit modification in the key. In our proposed scheme, the key consists of a 128-bit random sequence r_0, two order rule f and three random numbers n_1, n_2 and n_3.

To test the key sensitivity, the following experiments are performed: for a given plain image, encrypting using $K_0\left(r_0, f, n_1=3, n_2=4, n_3=3\right)$ is carried out to obtain a reference cipher image. Then, encryption of the plain image with three modified keys $K_1\left(r_0', f, n_1, n_2, n_3\right)$, $K_2\left(r_0, f', n_1, n_2, n_3\right)$ and $K_3\left(r_0, f, n_1=4, n_2=3, n_3=3\right)$ is also performed, where the difference between r_0 and r_0', f' and f is only one bit.

To compare the results, the average correlation coefficient between some pixels for each pair selected from the cipher images is calculated, as shown in **Table 3**. The results show that the encryption is very sensitive to slight key modifications. Another way to show key sensitivity is to compare the deciphered image using a wrong key with the correct deciphered key. The results of this experiment performed on image rice are illustrated in **Fig. 7,** and it's obvious that decryption is also sensitive to key modifications.

Table 3. Key sensitivity with respect to encryption

Plain image	Cipher image with K_0	Cipher image with K_1	Cipher image with K_2	Cipher image with K_3
				
Correlation coefficient		0.0055	-0.0020	0.0029



Plain image Encrypted by K_0 Decrypted by K_0

Decrypted by K_1 Decrypted by K_2 Decrypted by K_3

Fig. 7. Key sensitivity with respect to decryption

4.5.2 Differential Analysis

Differential attack is an important method in cryptanalysis, the attacker makes a slight change of the plain image and then observes the corresponding cipher image to determine the relationship between the plain and cipher image. Only if a minor change in the plain image causes a significant change in the cipher image can we ensure that the encryption algorithm is secure against differential attack.

Two criteria can be used to measure the variation in the cipher image when a slight change occurs in the plain image: the number of pixels change rate (NPCR) and unified average changing intensity (UACI); they are calculated using the following formulas:

$$\mathrm{NPCR} = \frac{\sum_{i,j} D(i,j)}{M \times N} \times 100\% \tag{13}$$

$$UACI = \frac{1}{M \times N} \sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{255} \times 100\% \tag{14}$$

$$D(i,j) = \begin{cases} 0, & C_1(x,y) = C_1(x,y) \\ 1, & \text{otherwize} \end{cases} \tag{15}$$

where M and N are the width and height of the image, respectively, and C_1 and C_2 are the cipher images, with only one different pixel between their plain images. For gray images with 8 bits per pixel, the expected values of NPCR and UACI are 99.6094% and 33.4635%, respectively. **Table 4** and **Table 5** illustrate different NPCR and UACI values, respectively, obtained for cipher images of three gray images. The results are also compared with those obtained by six existing 1D CA-based and 2D CA-based algorithms. It is obvious that all NPCR and UACI values of our proposed algorithm are the closest to the expected values and hence the proposed algorithm can resist differential attack effectively.

Table 4. Comparison of NPCR values of images encrypted by different algorithms

Image	Proposed	1D cellular automata			2D cellular automata		
		Ref. [2]	Ref. [4]	Ref. [14]	Ref. [3]	Ref. [7]	Ref. [9]
testpat1	99.61%	99.58%	99.82%	99.10%	99.79%	99.77%	99.60%
rice	99.61%	99.61%	99.87%	99.46%	99.76%	99.70%	99.62%
cameraman	99.58%	99.64%	99.79%	99.55%	99.50%	99.65%	99.59%

Table 5. Comparison of UACI values of images encrypted by different algorithms

Image	Proposed	1D cellular automata			2D cellular automata		
		Ref. [2]	Ref. [4]	Ref. [14]	Ref. [3]	Ref. [7]	Ref. [9]
testpat1	33.48%	33.32%	33.52%	33.16%	33.35%	33.49%	33.43%
rice	33.52%	33.40%	33.49%	33.61%	33.44%	33.63%	33.43%
cameraman	33.43%	33.44%	33.70%	33.57%	33.40%	33.57%	33.49%

4.6 Performance Analysis

Because cellular automata are natively parallelizable, a CA-based encryption scheme is very efficient in both hardware and software implementations. The results of several experiments above indicate that the proposed algorithm based on LCA, compared to the algorithms based on 1D or 2D CA, has better encryption effect; whether it is more efficient would require more experiments. Analysis of the encryption speed of the proposed scheme is given by comparing with six other 1D CA-based and 2D CA-based algorithms proposed in Refs. [2, 4, 14] and Refs. [3, 7, 9], respectively. We randomly choose one 256×256 gray image and encrypt it using these schemes on an Intel Core 2 Duo 2.0 GHZ in the Matlab 2010a platform; the encryption times are listed in **Table 6**. It is observed that the time taken by our proposed scheme is less than that by the other algorithms, which obviously demonstrates the efficiency of the proposed scheme.

Table 6. Encryption time comparison of different algorithms

Algorithm	Proposed	1D cellular automata			2D cellular automata		
		Ref. [2]	Ref. [4]	Ref. [14]	Ref. [3]	Ref. [7]	Ref. [9]
Encryption time (ms)	299	336	427	392	367	352	411

5. Conclusion

In this paper, a new reverse iterative image encryption scheme using LCA is proposed. The plain image is set as the final configuration of an 8-layer CA, which is backward evolved to return to its initial configuration by using a reversible two order rule. According to the definition of the two order rule, we employ a random sequence to define the pre-final configuration of this LCA; moreover, iterations of the modules and the crossover operations between different layers are determined by using random numbers to ensure that the same plain image will have distinctly different cipher images when encrypted many times. A comprehensive security analysis and performance test of the proposed scheme are given, and the results show that the proposed scheme possesses good confusion and diffusion properties and has excellent performance against statistical attacks; moreover, this scheme is more efficient when compared to other schemes.

References

[1] Priya S S S, KarthigaiKumar P and Mangai N M S et al., "Survey on Efficient, Low-power, AES Image Encryption and Bio-cryptography Schemes," *Smart Computer Review*, vol. 2, no. 6, pp. 379-390, December 2012. Article (CrossRef Link)

[2] Mohamed and Faraoun Kamel, "A parallel block-based encryption schema for digital images using reversible cellular automata," *Engineering Science and Technology, an International Journal*, vol. 17, no. 2, pp. 85-94, 2014. Article (CrossRef Link)

[3] Wang, Xingyuan, and Dapeng Luan, "A novel image encryption algorithm using chaos and reversible cellular automata," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 11, pp. 3075-3085, 2014. Article (CrossRef Link)

[4] Abdo, A. A., et al., "A cryptosystem based on elementary cellular automata," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 1, pp. 136-147, 2013. Article (CrossRef Link)

[5] Jin and Jun, "An image encryption based on elementary cellular automata," *Optics and Lasers in Engineering*, vol. 50, no. 12, pp. 1836-1843, 2012. Article (CrossRef Link)

[6] Wang, Xingyuan, and Dahai Xu, "A novel image encryption scheme using chaos and Langton's Ant cellular automaton," *Nonlinear Dynamics*, vol. 79, no. 4, pp. 2449-2456, 2015. Article (CrossRef Link)

[7] Zhang, Xiaoyan, et al., "Image encryption scheme based on balanced two-dimensional cellular automata," *Mathematical Problems in Engineering*. 2013. Article (CrossRef Link)

[8] Ping, Ping, Feng Xu, and Zhi-Jian Wang, "Image encryption based on non-affine and balanced cellular automata," *Signal Processing*, vol. 105, pp. 419-429, December 2014. Article (CrossRef Link)

[9] Ping, Ping, Wang Zhi-jian, and Feng Xu, "A Two-Dimensional Cellular Automata Based Method for Multiple Image Encryption," in *Proc. of International Conference on Computer Science and Service System (CSSS 2014)*, pp. 525-528, 2014. Article (CrossRef Link)

[10] Chen, Rong-Jian, and Jui-Lin Lai, "Image security system using recursive cellular automata substitution," *Pattern Recognition*, vol. 40, no. 5, pp. 1621-1631, 2007. Article (CrossRef Link)

[11] Zhang, Shuiping, and Huijune Luo, "The research of image encryption algorithm based on chaos cellular automata," *Journal of Multimedia*, vol. 7, no. 1, pp. 66-73, 2012. Article (CrossRef Link)

[12] del Rey, A. Martín, G. Rodríguez Sánchez, and A. de la Villa Cuenca, "Encrypting digital images using cellular automata," *Hybrid Artificial Intelligent Systems*. Springer Berlin Heidelberg, pp. 78-88, 2012. Article (CrossRef Link)

[13] Tafti, Ahmad Pahlavan, and Safoura Janosepah, "Digital images encryption in frequency domain based on DCT and one dimensional cellular automata," *Informatics Engineering and Information Science*. Springer Berlin Heidelberg, pp. 421-427, 2011. Article (CrossRef Link)

[14] Bakhshandeh, Atieh, and Ziba Eslami, "An authenticated image encryption scheme based on chaotic maps and memory cellular automata," *Optics and Lasers in Engineering*, vol. 51, no. 6, pp. 665-673, 2013. Article (CrossRef Link)

[15] Wu, Xiaotian, and Wei Sun, "Secret image sharing scheme with authentication and remedy abilities based on cellular automata and discrete wavelet transform," *Journal of Systems and Software*, vol. 86, no. 4, pp. 1068-1088, 2013. Article (CrossRef Link)

[16] Zamani, Samaneh, et al., "A novel image encryption scheme based on hyper chaotic systems and fuzzy cellular automata," in *Proc. of Electrical Engineering (ICEE), 2014 22nd Iranian Conference on*. IEEE, 2014. Article (CrossRef Link)

[17] Chen, Rong-Jian, and Shi-Jinn Horng, "Novel SCAN-CA-based image security system using SCAN and 2-D von Neumann cellular automata," *Signal Processing: Image Communication*, vol. 25, no. 6, pp. 413-426, 2010. Article (CrossRef Link)

[18] Nandi, Subrata, et al., "1-D Group Cellular Automata based Image Encryption Technique," in *Proc. of Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014 International Conference on*. IEEE, 2014. Article (CrossRef Link)

[19] Ping, Ping, Feng Xu, and Zhi-Jian Wang, "Color image encryption based on two-dimensional cellular automata," *International Journal of Modern Physics C*, vol. 24, no. 10, 2013. Article (CrossRef Link)

[20] Machicao, Jeaneth, Anderson G. Marco, and Odemir Martinez Bruno, "Chaotic encryption method based on life-like cellular automata," *Expert Systems with Applications*, vol. 39, no. 16, pp. 12626 -12635, 2012. Article (CrossRef Link)

[21] Ayanzadeh, Ramin, et al., "Multi-layer cellular automata for generating normal random numbers," in *Proc. of Electrical Engineering (ICEE), 2010 18th Iranian Conference on*. IEEE, 2010. Article (CrossRef Link)

[22] Jaberi, Alireza, Ramin Ayanzadeh and Azam S. Zavar Mousavi, "Two-layer cellular automata based cryptography," *Trends in applied sciences research*, vol. 7, no. 1, pp. 68-77, 2012. Article (CrossRef Link)

[23] Rao, Chinta Somswara, et al., "Implementation of object oriented encryption system using layered cellular automata," *International Journal of Engineering Science and Technology (IJEST)*, vol. 3, no. 7, 2011. Article (CrossRef Link)

[24] Zhang, Xing, et al., "A New Public Key Encryption Scheme based on Layered Cellular Automata," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 8, no. 10, pp. 3572-3590, 2014. Article (CrossRef Link)

[25] Kari, Jarkko, "Reversibility of 2D cellular automata is undecidable," *Cellular Automata: Theory and Experiment*, vol. 45, no. 1-3, pp. 379-385, September 1990. Article (CrossRef Link)

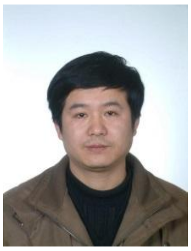
[26] Kari, Jarkko, "Reversibility and surjectivity problems of cellular automata," *Journal of Computer and System Sciences*, vol. 48, no. 1, pp. 149-182, 1994. Article (CrossRef Link)

[27] Das, Debasis, "A survey on cellular automata and its applications," *Global trends in computing and communication systems*. Springer Berlin Heidelberg, pp. 753-762, 2012. Article (CrossRef Link)

[28] Xuewen, Xia, et al., "Data encryption based on multi-granularity reversible cellular automata," in *Proc. of Computational Intelligence and Security, 2009. CIS'09. International Conference on*. Vol. 2. IEEE, pp. 192-196, 2009. Article (CrossRef Link)

Xing Zhang received the B.S.degree from Xuchang University, China, in 2010. From 2010 to now, she is working her Ph.D. degree in Computer Application from Nanjing University of Science and Technology (NUST), Jiangsu, China. During the period from November 2013 to May 2014, she was also a visiting Ph.D. student at the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Her research interests include information security and cryptography, and the encryption scheme based on cellular automata.

Hong Zhang is a professor in the Department of Computer Science, Nanjing University of Science and Technology. His current interests are in the areas of theory and technology of information security, data mining and network fault diagnosis.



Chungen Xu received the M.S. degree from East China Normal University, Shanghai, China, in 1996 and the Ph.D degree from Nanjing University of Science and Technology in 2003. He is a professor in the Department of Applied Mathematics, School of Sciences, Nanjing University of Science and Technology. His current interests are in the areas of computer and network security, cryptography and coding.