Research Article

Image Encryption Scheme Based on Balanced Two-Dimensional Cellular Automata

Xiaoyan Zhang,1,2 Chao Wang,3 Sheng Zhong,4 and Qian Yao3

1 School of Mathematical Science and Institute of Mathematics, Nanjing Normal University, Nanjing 210023, China
2 Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, 7500 AE Enschede, The Netherlands
3 College of Software, Nankai University, Tianjin 300071, China
4 State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210046, China

Correspondence should be addressed to Chao Wang; nkcs.wangchao@gmail.com

Received 14 April 2013; Revised 5 September 2013; Accepted 13 September 2013

Academic Editor: Jui-Sheng Lin

Copyright © 2013 Xiaoyan Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Cellular automaton (CA) is a simple model of computation which exhibit fascinatingly complex behavior. Due to the universality of CA model, it has been widely applied in both cryptography and image processing. The aim of this paper is to present a new image encryption scheme based on balanced two-dimensional cellular automata. In this scheme, a random image with the same size of the plain image to be encrypted is first generated by a pseudo-random number generator with a seed. Then, the random image is evolved alternately with two balanced two-dimensional CA rules. At last, the cipher image is obtained by operating bitwise XOR on the final evolution image and the plain image. This proposed scheme possesses some advantages such as very large key space, high randomness, complex cryptographic structure, and pretty fast encryption/decryption speed. Simulation results obtained from some classical images at the USC-SIPI database demonstrate the strong performance of the proposed image encryption scheme.

1. Introduction

Nowadays images are widely used in multimedia applications. These images often contain private or confidential information and sometimes they are associated with financial interests. Security thus becomes increasingly important in the communication and storage of these images. In order to guarantee security, we need process images with the help of cryptography. The purpose of cryptography is to hide the content of messages by encrypting them, so as to make them unrecognizable unless decrypted by someone who has been given a special decryption key. There are several conventional encryption methods which are often used to encrypt common messages or texts, but these traditional cryptographic algorithms do not fit image encryption because of the different characteristics between digital data and image data. The notion of “image encryption” is aiming toward the emerging cryptographic technologies and applications on images. Image encryption has applications in many fields such as internet communication, multimedia systems, medical imaging, telemedicine, and military communication. However, images have intrinsic characteristics like bulk data capacity and high redundancy, as well as real-time requirement. Encryption on images has its own special requirements.

Cellular automaton (CA) is a good candidate for image cryptosystems because of its inherent properties like parallelism, homogeneity, and unpredictability, as well as it is being easily implemented in both software and hardware systems. Ever since Wolfram studied the first secret key process based on cellular automata [1], many researchers had explored variant cryptography based on them [2–5]. In recent years, CA especially has been used widely for image cryptography, including image scrambling [6], watermarking [7], secret image sharing [8–11], image encryption [12–18], and image security system [19–21]. In addition, CA has been applied in the field of image processing [22], including image coding [23] and image segmentation [24]. Other methods also have been researched for image encryption [25–27].
In the first beginning, some researchers designed image encryption schemes with one-dimensional (1D) CA [17, 18]. Because the image is a two-dimensional array of pixels instead of a one-dimensional digital data, two-dimensional (2D) cellular automata will be more suitable to be applied for image encryption. So later, some people introduced two-dimensional CA in the application of image encryption. But many of them used Von Neumann neighborhood [12, 13, 19, 20], or incomplete Moore neighborhood [14], instead of Moore neighborhood, which influenced the capacity of key space. In this paper, a new image encryption method is proposed based on balanced two-dimensional cellular automata with complete Moore neighborhood. Firstly, we use 2D cellular automata with complete Moore neighborhood which is easy to get larger key space. Secondly, we use balanced CA rules which are helpful to generate cipher images with higher randomness. Thirdly, we choose several rows as the radius-2 neighborhood, which enhances the complexity of our encryption scheme. Furthermore, the encryption/decryption algorithms with their properties are tested and analyzed. Results show that this image encryption is lossless and has a good statistical key space analysis and other security analysis are shown in Section 5. Finally, some conclusions are presented in Section 6.

2. 2D Cellular Automata

A cellular automaton is a discrete dynamical system that consists of an arrangement of basic components called cells together with a transition local rule \( f \) [1]. The cells have a finite number of states that are updated synchronously according to the specified local rule. Configuration \( C_j \) consists of all the states of cells at time \( t \). Since image is made up of pixels specified by two-dimensional plane coordinates, we take a two-dimensional CA as example in a concrete description. \( \mathbb{Z}^2 \) is the underlying space of the two dimensional CA. The cells are arranged in a square lattice structure. The intersections of squares form the cells of the automaton. We can use \((a, b)\) to specify every cell whose state is assumed to be 0 or 1 and whose neighborhood is the set of nine cells \((a \pm 1, b), (a, b \pm 1), (a \pm 1, b \pm 1), \) and \((a, b)\). The neighborhood is the so-called Moore neighborhood which is formed by the specified cell \((a, b)\) with its eight nearest neighbors. We denote the state of cell \((a, b)\) by \( s_{a,b} \). Then, the nine cells are arranged in \( f \) in such a way as

\[
f (s_{a-1,b-1}, s_{a-1,b}, s_{a-1,b+1}, s_{a,b-1}, s_{a,b}, s_{a,b+1}, s_{a+1,b-1}, s_{a+1,b}, s_{a+1,b+1}) .
\]

Hence, \( f \) can be expressed in equations as follows:

\[
f (000000000) = \varepsilon_0, f (000000001) = \varepsilon_1, \cdots f (\text{bin}(i)) = \varepsilon_i, \cdots, f (111111111) = \varepsilon_{511},
\]

where bin\((i)\) means the 8-bit binary representation of \( i \), \( \varepsilon_i \in \{0, 1\} \), and \( i \in \{0, 1, \ldots, 511\} \). \( f \) is called a radius-1 neighborhood local rule. For convenience, we represent the local rule \( f \) by \( \varepsilon_0\varepsilon_1 \cdots \varepsilon_{511} \), which is called the Wolfram number of \( f \). For example, \( f = 0\times A16DB83C \) \( ^{(1)} \) represents a 128-bit hexadecimal number, that is, 512-bit binary number, combined by 16 copies of the same hexadecimal number \( 0\times A16DB83C \) together. If there are 256 0s and 256 1s in the 512-bit binary representation of \( f \), we call \( f \) a balanced CA local rule. If we consider the CA rule and the plain text as an information source, the amount of information is the increased uncertainty of ciphertext generated by the information source, that is, entropy. It can be proved that the ciphertext generated by the balanced CA rules has a bigger entropy that that generated by the unbalanced CA rules. The balanced CA rules are better rules to generate ciphertext with maximal entropy. So in the proposed image security system, we use balanced CA rules to encrypt images.

We denote

\[
\{(y_1, y_2) : |y_1 - x_i| \leq j, 1 \leq i \leq 2, j \in \mathbb{Z}^+\}
\]

as the radius-\( j \) neighborhood of the cell \((x_1, x_2)\). Note that there are 25 cells in every cell’s radius-2 neighborhood, and

the number of independent variables of local rule achieves \( 2^{5^2} \). Then, the set of local rules would contain

\[
2^{25^2} = 2^{33554432}
\]

elements. Similar to radius-1 neighborhood local rule, we represent the radius-2 neighborhood local rule \( g \) by

\[
\varepsilon_0, \varepsilon_1, \cdots, \varepsilon_{33554431} .
\]

For example,

\[
g = 0\times E2A87F53C046BD191D5780AC3FB942E6 \]

represents a 8388608-bit hexadecimal number, that is, 33554432-bit binary number, combined by 262144 copies of the same hexadecimal number

\[
0\times E2A87F53C046BD191D5780AC3FB942E6
\]

together. If exactly half of these 33554432 binary numbers are 0, \( g \) is also a balanced CA local rule. A radius-\( j \) neighborhood local rule can be similarly represented if \( j \geq 3 \).

Suppose that the original configuration is \( C_0 \) at time \( t = 0 \), and the states of cells, say, in the radius-1 neighborhood...
of a cell whose state is $x_0$ are $x_0, x_1, \ldots, x_8$. If we operate
the rule $f$ on $C_0$, the state of the cell will change to be $f(x_0, x_1, x_2, \ldots, x_8)$ at time $t = 1$. All cells' states can be
obtained in the same way. We put the states together to obtain configuration $C_1$ which can be considered as the result of
the global transformation $\rho_1$ acting on $C_0$. We can write
$C_1 = f(C_0)$. Sometimes, we simply write $C_1 = f(C_0)$. Furthermore, $C_2 = f(C_1), C_3 = f(C_2), C_4 = f(C_3), \ldots$ can
be generated in succession.

3. The Proposed Image Encryption and Decryption Scheme

Because every image is finite while $\mathbb{Z}^2$ is infinite, we must deal
with this contradiction by applying two dimensional CA. If we consider
the image to be on a topological anchor ring, that is, the right border
of the image laps the left one and the top border superposes the bottom one, then every pixel has nine neighbors (including itself) in its neighborhood in the image.

Without loss of generality, we take black-and-white binary image $M$ with height $H$ and width $W$ to illustrate how to encrypt image by applying the proposed CA approach.

Let black represent 1 and let white represent 0 in a black-and-white binary image $M$. Each row of image $M$ can be considered as a 0-1 binary string of length $W$.

Our proposed image encryption/decryption scheme is a kind of stream cipher. In our image encryption scheme, the sender Alice and the receiver Bob agree on some necessary information in the beginning: a selected balanced radius-1 neighborhood local rule $f_0$, a selected balanced radius-2 neighborhood local rule $f_1$, a seed(seed) which is used to generate pseudo-random numbers, a subpermutation $\pi(H) = i_0 i_1 \ldots i_{\delta}$ of $\{1, 2, \ldots, H\}$, where $i_j \in \{1, 2, \ldots, H\}$, $1 \leq j \leq \delta, 1 \leq k \leq H$. In other words, Alice should transmit $f_0, f_1$, seed, and $\pi(H)$ to Bob through secret channel before she sends cipher images. Alice generates an original random binary image $R_M$ with the same size as the plain image $M$ by a pseudo-random number generator (PRNG) with seed.

Denote the $i$th row of $R_M$ by $L_i$, where $i \in \{1, 2, \ldots, H\}$. $L_i$ will be a binary string of length $W$. We concatenate the strings $L_{i_1}, L_{i_2}, \ldots, L_{i_\delta}$ one by one, then we get a binary string of length $kW$, denoted by $\bar{L}$, that is, $\bar{L} = L_{i_0}, L_{i_1}, \ldots, L_{i_{\delta}}$. Here $\bar{L}$ can be considered as a control sequence to determine whether $f_0$ or $f_1$ is being used in the encryption process. Suppose

$$\bar{L} = p_1, p_2, \ldots, p_{W \times \bar{L}} \mid p_{W+1}, p_{W+2}, \ldots, p_{2W} \mid \ldots \mid p_{(k-1)W+1}, p_{(k-1)W+2}, \ldots, p_{kW},$$

where $p_j$ is 0 or 1, $1 \leq j \leq kW$, and the short vertical lines are used to facilitate observation. The proposed image encryption/decryption scheme is described as follows.

3.1. The Image Encryption Algorithm

Step 1. Alice generates $R_M$ by a PRNG with seed. Let $i = 1$ and $C_0 = R_M$ be the configuration at time $t = 0$.

Step 2. Alice gets $T_i$ after the sequential local rules $f_{P_0}, f_{P_1}, \ldots, f_{P_W}$, $f_{P_{W+1}}, \ldots, f_{P_{2W}}$, acting on $C_{i-1}$ successively, that is, $T_i = f_{P_0}(f_{P_{W+1}}(\cdots f_{P_{W+1}}(f_{P_{W+1}}(C_{i-1}))))$. Then let $C_i = T_i \oplus R_M$; that is, operate XOR on $T_i$ and $R_M$.

Step 3. Let $i = i + 1$. If $i \leq k$, go to Step 2; else, go to Step 4.

3.2. The Image Decryption Algorithm.

Step 1. Bob gets $C_0 = R_M$ by the PRNG with seed that he has obtained through the secret channel from Alice. And then he gets $L = L_1, L_2, \ldots, L_\delta$ from $\pi(H)$ and $R_M$; that is, he knows

$$L = p_1, p_2, \ldots, p_{W \times \bar{L}} \mid p_{W+1}, p_{W+2}, \ldots, p_{2W} \mid \ldots \mid p_{(k-1)W+1}, p_{(k-1)W+2}, \ldots, p_{kW}.$$ Let $i = 1$.

Step 2. Bob gets $T_i$ after the sequential local rules $f_{P_0}, f_{P_1}, \ldots, f_{P_W}$, $f_{P_{W+1}}, \ldots, f_{P_{2W}}$, acting on $C_{i-1}$ successively, and then he gets $C_i = T_i \oplus R_M$.

Step 3. Let $i = i + 1$. If $i \leq k$, go to Step 2; else, go to Step 4.

Step 4. Bob easily recovers the original secret black-and-white binary image $M$ through the operation of XOR on $C_k$ and $E$ because $C_k \oplus E = C_k \oplus (C_k \oplus M) = M$.

4. Simulation Experiments

As an example, we take the classic gray-scale image “Lena” with size $W \times H$, as the plain image M to show our algorithm. Here, $W = H = 128$. The gray scale of each pixel in the image is between 0 and 255. In order to use our algorithm, we represent each pixel's gray scale in the binary number system. For example, let $m_{ij}$ be the gray scale of the pixel in the $i$th row and $j$th column and $m_{ij} = b_1 b_2 \ldots b_8$, where $b_1 = 0$ or 1, $1 \leq i \leq 8$. Then, we regard “Lena” image as composition of 8 layers. Each layer is a black-and-white image of size $W \times H$.

The pixel located in the $i$th row and $j$th column of the $k$th layer is black if $b_k = 1$ and white if $b_k = 0$. In other words, the $k$th layer consists of all the pixels with value 0 or 1 that equals to the $k$th bit of the binary representation of each pixel of “Lena” image.

Suppose Alice and Bob both agree on the balanced radius-1 neighborhood local rule $f_0$ and balanced radius-2 neighborhood local rule $f_1$ being

$$f_0 = 0x(A16DB83C)^{16},$$

$$f_1 = 0xE2A87F53C046DB19)^{524288},$$ respectively. Of course they can choose other appropriate balanced CA rules as they want. Alice generates a random gray-scale image $R_M$ with the same size of “Lena” image by the PRNG with seed. $R_M$ is also
5. Security Analysis

A good encryption scheme should be robust against all kinds of attacks that are already known, and key space should be large enough to make brute-force attack infeasible. In this section, some security analyses such as key space analysis, statistical analysis, and confusion analysis, have been performed on the proposed image encryption scheme to show that the proposed scheme is secured against the most common attacks.

5.1. Key Space Analysis. Key space should be large enough to be robust against the unpredicted attacks. In our encryption scheme, we only need at most \((H + 4M + 64)\) byte memory to store the sequential local rules \(f_{p_1}, f_{p_2}, \ldots , f_{p_{16}}\), which can be realized easily in practice. If the eavesdropper Charlie wants to break this scheme by brute force, he must contend with searching through all subpermutations of \(\{1, 2, \ldots , H\}\), all \(\left( \frac{512}{256} \right)\) possible balanced radius-1 neighborhood local rules and all \(\left( \frac{33554432}{16777216} \right)\) possible balanced radius-2 neighborhood local rules. By counting arguments, we can know that the number of all subpermutations of \(\{1, 2, \ldots , H\}\) is

\[
\sum_{k=1}^{H} \binom{H}{k} k! = \sum_{k=1}^{H} \frac{H!}{(H-k)!} = H! \sum_{m=0}^{H-1} \frac{1}{m!} \approx H!e, \quad (11)
\]

where \(e\) is the base of natural logarithm.

Hence, the volume of security key of our encryption scheme is approximately

\[
H!e \times \left( \frac{512}{256} \right) \times \left( \frac{33554432}{16777216} \right) > H!e \times 10^{101039}. \quad (12)
\]

The volume of security key of the scheme that we proposed is much larger than 256, which is the volume of security key introduced in [14]. And it is also larger than \(10^{536}, 10^{18241}, \) and \(10^{4775}\) which are the lower bounds of the volume of security key introduced in [12], [19], and [20], respectively. In addition, it is larger than the volume of security key introduced in [16–18]. This makes the algorithm robust against unpredicted attacks.

5.2. Statistical Analysis. An encryption algorithm should be robust against statistical attacks. For testing it, statistical analysis such as image histogram and correlation coefficient of images has been conducted. In order to get the better statistical performance, we use the balanced CA rules instead of common CA rules in our proposed image encryption scheme.

5.2.1. Histogram Analysis. Image histogram is used to show the number of pixels per gray level. In general, more uniformed histogram results in less statistical attacks. The encrypted results are shown in Figure 3. Frame (a) is the original image and frame (c) is its histogram. Frame (b) is the encrypted image and frame (d) is the corresponding histogram of it.

It is clearly shown in Figure 3 that the histogram of the encrypted image is almost uniform regardless of the original image and is significantly different from that of the original image, so statistical attacks will become very difficult.

5.2.2. Correlation Coefficients. In a digital image pixels are not independent and their relevance is great. This may indicate that a large area of the image has similar gray-scale value. For example, in a common television digital image,
the correlation coefficient of adjacent pixels may reach 0.9; that is, the relevance (which means information redundancy) is very big. The smaller the relevance, the better is the encryption effect and the higher is the security.

We show the fact that the relevance between plain images and cipher images is very small in two ways.

First we repeat the encryption scheme 16 times using the plain-image "Lena." Each time a different random image $R_M$ is generated for the encryption scheme. We calculate the correlation coefficients between the plain-image $M$ and the ciphered-image $E$. Here, we use the formulae in [21]:

$$\rho = \frac{\text{cov}(x, y)}{\sqrt{D(x)} \sqrt{D(y)}}$$

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

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

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

where $x$ are the gray-scale values of all the pixels in the images $M$, $y$ are the gray-scale values of all the pixels in the images $E$, and $N$ is total number of pixels of $M$ or $E$. $\rho$ can vary between $-1$ and $1$, so $0 \leq |\rho| < 1$. If $|\rho|$ is near 1, we will conclude that $x$ and $y$ are correlated and if $|\rho|$ is near 0, we will conclude that there is a trivial correlation between $x$ and $y$. All the correlation coefficients are calculated and listed in Table 1 as below.

Second, we test horizontal, vertical, and diagonal correlations for the plain-image and cipher-image after the encryption process. A number of 4096 random pairs of horizontally, vertically, and diagonally adjacent pixels have been tested. Numerical results are shown in Table 2.

Correlation coefficients in the table are between $-1$ and 1. It is a fact that the closer the coefficients are to zero, the smaller the correlation between pixels is. The results show that the correlation coefficients of the encrypted image are close to zero, so correlation between pixels of the encrypted image is very small. This indicates little possibility of prediction and high security.

In order to see it more clearly, we show the correlation of two horizontally, vertically, and diagonally adjacent pixels in the original image and the encrypted image in Figure 4. Note that to be more explicit, only 600, instead of 4096, points are drawn in each frame.
Figure 4: Correlation of two horizontally, vertically, and diagonally adjacent pixels: (a), (c), and (e) in the original image (b), (d), and (f) in the encrypted image.
5.3. Confusion Analysis. In Shannon’s original definitions, confusion refers to making the relationship between the ciphertext and the private key as complex and involved as possible. An excellent encryption algorithm should be sensitive to small changes in key. In other words, a slight change in the key can cause very different results. For example, in the experiment simulation, we use \( f_0 \) and \( f_1 \) to encrypt the plain-images. Then we make a very slight modification, for example, changing the first bit of the Wolfram number of \( f_0 \) to its complement to get \( f_0’ \). In order to prove that the very small change can lead to a large difference, we perform two simulations as follows.

(1) Encrypt three classical images at the USC-SIPI database [28] with key CA rules \( f_0 \) and \( f_1 \), \( f_0’ \) and \( f_1’ \), respectively, while the supermutation \( \pi(H) \) is the same. Here “Lena,” “House,” and “Pepper” are used as examples. The correlation coefficients between each pair of encrypted images are calculated in Table 3, which shows that a small change of key can lead to very different results.

In order to give a clearer quantitative illustration, we introduce two indexes NPCR and UACI for further analysis.

(1) NPCR Analysis. In this analysis, a 128 × 128 image “Lena” is chosen and encrypted by using \( f_0 \) and \( f_1 \) to get \( E_1 \). Then, it is encrypted once again by using \( f_0’ \) and \( f_1’ \) to get \( E_2 \). Label the gray-scale values of the pixels at grid \((i, j)\) in \( E_1 \) and \( E_2 \) by \( E_1(i, j) \) and \( E_2(i, j) \), respectively. Define the “difference array,” \( D \), which is a binary array, with the same size as images \( E_1 \) and \( E_2 \). And \( D(i, j) \) is determined by \( E_1(i, j) \) and \( E_2(i, j) \), namely,

\[
D(i, j) = \begin{cases} 
1 & \text{if } E_1(i, j) = E_2(i, j), \\
0 & \text{otherwise}. 
\end{cases}
\]

(14)

The number of pixels change rate (NPCR) is defined as

\[
\text{NPCR} = \frac{\sum_{i,j} D(i, j)}{W \times H} \times 100\%,
\]

(15)

where \( W \) and \( H \) are the width and height of \( E_1 \) or \( E_2 \).

For two independent random images, the expected value of NPCR is \( E(\text{NPCR}) = (1 - 2^{-8}) \times 100\% = 99.6094\% \).

(II) UACI Analysis. The unified average changing intensity (UACI) is defined as

\[
\text{UACI} = \frac{1}{W \times H} \left[ \sum_{i,j} \left| E_1(i, j) - E_2(i, j) \right| \right] \times 100\%,
\]

(16)

where \( W \) and \( H \) are the width and height of \( E_1 \) or \( E_2 \).

The UACI measures the average intensity of the differences between the two cipher images. For two random images, the expected value of UACI is

\[
E(\text{UACI}) = \frac{\sum_{i=1}^{2^{L-1}} i(i+1)}{2^L(2^L - 1)} \times 100\% = \frac{1 + 2^{-L}}{3} \times 100\%.
\]

(17)

where \( L \) is the number of bits used to represent one pixel of the image. For two 8-bit per pixel gray-scale random images, \( E(\text{NPCR}) = (1 - 2^{-8}) \times 100\% = 99.6094\% \).

In our study, we encrypt the plain-image “Lena” with key CA rules \( f_0 \) and \( f_1 \). Then, we encrypt the plain-image “Lena” once again with key CA rules \( f_0’ \) and \( f_1’ \), \( f_0’ \) and \( f_1’ \) only differ in the first bit. Finally, the encrypted images are compared. We measure NPCR and UACI of the two cipher images with only one bit difference in key rules and the results are given in Table 4.

As you see in Table 4, the NPCR and UACI values of the scheme are very near to the ideal values 99.6094% and 33.4635%, respectively. Based on these findings, we conclude that our method is safe and our system is secured.

(2) After encrypting the plain-image with key rules \( f_0 \) and \( f_1 \) as mentioned above, we make a very slight modification, for example, changing the first bit of the Wolfram number of \( f_0 \) to its complement to get \( f_0’ \). Then, if we use \( f_0’ \) and \( f_1 \) to decrypt the ciphered image, the decryption should not succeed. Table 5 confirms our findings. In it, frame (a) is the plain image. Frame (b) is the cipher-image encrypted with rules \( f_0 \) and \( f_1 \). Frames (c) and (d) show the decrypted images from frame (b) with key rules \( f_0’ \) and \( f_1’ \), respectively. It is clear that the image encrypted by \( f_0’ \) and \( f_1’ \) cannot be correctly decrypted by using \( f_0’ \) and \( f_1’ \). Since there is only one bit difference between the two keys, the results show the high key sensitivity of the proposed CA encryption scheme.

5.4. Information Entropy. Entropy is one of the important features for randomness. The information entropy is defined as follows:

\[
H(s) = - \sum_{i=0}^{L} p(m_i) \log_2 p(m_i) \text{ bits},
\]

(18)
Table 3: Encrypted images with different keys with very small modification.

<table>
<thead>
<tr>
<th>Original image</th>
<th>Encrypted image with $f_0$ and $f_1$</th>
<th>Encrypted image with $f'_0$ and $f_1$</th>
<th>Correlation coefficients</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Original Image" /></td>
<td><img src="image2.png" alt="Encrypted Image" /></td>
<td><img src="image3.png" alt="Encrypted Image" /></td>
<td>0.0112</td>
</tr>
<tr>
<td><img src="image4.png" alt="Original Image" /></td>
<td><img src="image5.png" alt="Encrypted Image" /></td>
<td><img src="image6.png" alt="Encrypted Image" /></td>
<td>-0.0032</td>
</tr>
<tr>
<td><img src="image7.png" alt="Original Image" /></td>
<td><img src="image8.png" alt="Encrypted Image" /></td>
<td><img src="image9.png" alt="Encrypted Image" /></td>
<td>-0.0038</td>
</tr>
</tbody>
</table>

Table 4: Comparison of NPCR and UACI in encrypting different plain images.

<table>
<thead>
<tr>
<th>Original images</th>
<th>NPCR of encrypted images</th>
<th>UACI of encrypted images</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Original Image" /></td>
<td>99.7736%</td>
<td>33.4916%</td>
</tr>
<tr>
<td><img src="image4.png" alt="Original Image" /></td>
<td>99.6950%</td>
<td>33.6278%</td>
</tr>
<tr>
<td><img src="image7.png" alt="Original Image" /></td>
<td>99.6532%</td>
<td>33.5719%</td>
</tr>
</tbody>
</table>

Table 5: Encrypt Lena image with key rules $f_0$ and $f_1$ and then decrypt it with $f'_0$ and $f_1$.

<table>
<thead>
<tr>
<th>Original Lena image</th>
<th>Encrypted image with $f_0$ and $f_1$</th>
<th>Decrypted image with $f_0$ and $f_1$</th>
<th>Decrypted image with $f'_0$ and $f_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image10.png" alt="Original Lena" /></td>
<td><img src="image11.png" alt="Encrypted Lena" /></td>
<td><img src="image12.png" alt="Decrypted Lena" /></td>
<td><img src="image13.png" alt="Decrypted Lena" /></td>
</tr>
</tbody>
</table>

Frame (a) | Frame (b) | Frame (c) | Frame (d)
where \( n_i \) is the \( i \)th gray value for an \( L \) level gray image. The closer an entropy rate is to eight, the less is the possibility of predictability and the higher is the security level. As specified in Table 6, entropy of the encrypted images is very close to 8, and it indicates that this algorithm is robust against entropy attack.

### 6. Conclusion

In this paper, a new image encryption/decryption scheme is proposed based on balanced two-dimensional CA. If Alice wants to transmit a secret image \( M \) with height \( H \) and width \( W \) to Bob, they need agree on a subpermutation \( \pi(H) \), a balanced radius-1 neighborhood local rule \( f_0 \), a balanced radius-2 neighborhood local rule \( f_1 \), and a seed. Then, Alice generates a random image \( R_M \) by a PNRG with the seed and evolutes it \( kW \) times by alternatively using rules \( f_0 \) and \( f_1 \) according to the sub-permutation \( \pi(H) \) and \( R_M \). After each evolution, a bitwise XOR will be operated on \( R_M \) and the evoluted result to get \( C_i \). At last, Alice sends to Bob \( E = C_k @ M \) as the cipher-image. The decryption process is similar.

The scheme proposed in this paper possesses the characteristics of convenient realization, very large number of security keys, pretty fast encryption speed, and low cost. Theoretical analysis and simulated experiment show that the scheme has the confusion property and excellent performance against statistical attacks.

Cellular automata are simple models of computation which exhibit fascinatingly complex behavior. They have captured the attention of several generations of researchers, leading to an extensive body of work. For example, John Conway’s Game of Life is actually a two-dimensional CA with special local rule. Due to the universality of CA model, it can be more widely used in cryptography and image processing, and that will be our future work.

### Acknowledgments

The authors would like to thank the anonymous reviewers for their careful reading of the paper and valuable suggestions and comments that have helped a lot to improve the quality of the paper. This work was supported by RPGE, NSFC-61021062, and the Natural Science Foundation of the Jiangsu Higher Education Institutions of China (13KJB10018).

### References


### Table 6: Entropy of encrypted image.

<table>
<thead>
<tr>
<th>Original image</th>
<th>Entropy of the encrypted image</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Original image" /></td>
<td>7.9886</td>
</tr>
<tr>
<td><img src="image2.png" alt="Original image" /></td>
<td>7.9889</td>
</tr>
<tr>
<td><img src="image3.png" alt="Original image" /></td>
<td>7.9885</td>
</tr>
</tbody>
</table>
in Proceedings of the IEEE 10th International Symposium on

of images with 2-dimensional cellular automata," in Proceedings
of the 8th International Conference on Information Systems

image encryption system by 2D memorized cellular automata
and chaos mapping," in Proceedings of the 6th International
Conference on Digital Content, Multimedia Technology and Its

[16] J. Jin, "An image encryption based on elementary cellular

[17] L. Yu, Y. X. Li, and X. W. Xia, "Image encryption algorithm
based on self-adaptive symmetrical-coupled toggle cellular
automata," in Proceedings of the 1st International Congress on

[18] F. Maleki, A. Mohades, S. M. Hashemi, and M. E. Shiri, "An
image encryption system by cellular automata with memory," in
Proceedings of the 3rd International Conference on Availability,

cellular automata substitution," Pattern Recognition, vol. 40, no. 5,

security system using SCAN and 2-D von Neumann cellular

using recursive Cellular automata substitution and its paral-
lelization," in Proceedings of the CSI International Symposium on
Computer Science and Software Engineering (CSSS '11), pp. 77–
86, June 2011.


[23] L. Cappellari, S. Milani, C. Cruz-Reyes, and G. Calvagno, "Reso-
1468, 2011.

segmentation by cellular automaton on GPU," International
Journal of Computer Assisted Radiology and Surgery, vol. 5, no. 3,


[26] A. V. Diaconu and K. Loukhaoukha, "An improved secure image
encryption algorithm based on Rubiks cube principle and
digital chaotic cipher," Mathematical Problems in Engineering,

[27] Z. L. Zhu, W. Zhang, K. W. Wong, and H. Yu, "A chaos-
based symmetric image encryption scheme using a bit-level
permutation," Information Sciences, vol. 181, no. 6, pp. 1171–1186,
2011.

database/.