International Journal of Modern Nonlinear Theory and Application, **2013, 2, 89-96**
http://dx.doi.org/10.4236/ijmnta.2013.21A011 Published Online March 2013 (http://www.scirp.org/journal/ijmnta)

A Self-Adaptive Parallel Encryption Algorithm Based on Discrete 2D-Logistic Map

Jing Wang[1], Guoping Jiang[2]
[1]College of Computer, Nanjing University of Posts and Telecommunications, Nanjing, China
[2]College of Automation, Nanjing University of Posts and Telecommunications, Nanjing, China
Email: jing9823@163.com, jianggp@njupt.edu.cn

Received December 4, 2012; revised January 6, 2013; accepted January 18, 2013

ABSTRACT

A self-adaptive parallel encryption algorithm based on discrete 2D-Logistic map is developed according to the position scrambling and diffusion of multi-direction in variable space of spatial chaos. The binary sequences $b_1b_2b_3\cdots b_n$ are obtained according to the user key, in which the binary sequence 0 and 1 denote distribution mode of processors and the number of binary sequence n denotes cycle number. Then the pseudorandom 2D matrix is generated by 2D-Logistic map, and adaptive segmentation is applied in image matrix and pseudorandom matrix according to the value and the number of binary sequence. The parallel operation is used among blocks to improve efficiency and meet real-time demand in transmission processes. However, the pixel permutation is applied in partitioned matrix through ergodic matrix generated by pseudo-random matrix-block to decrease the correlation of adjacent pixels. Then the pixel substitution is used for fully diffusing through cipher block chaining mode until n cycles. The proposed algorithm can meet the three requirements of parallel operation in image encryption and the real-time requirement in transmission processes. The security is proved by theoretical analysis and simulation results.

Keywords: 2D-Logistic Map; Pseudo-Random Matrix; Theoretical Analysis; Simulation Results

1. Introduction

With the rapid development of computer technologies and communication networks, especially the wide application of different Internet services, more and more digital information such as voice, image and critical data are transmitted on networks. However, much important data is stolen or destroyed in information transmission, it has been reported that there is one case of hacker invasion every 20 seconds in the world [1]. Chaos theory has been established since 1970s by many different research areas, such as physics, mathematics and chemistry, etc. During the last decades, the use of chaos in cryptography and secure communication has aroused great interest of researchers because of its sensitivity to initial condition and control parameters, ergodicity and mixing, which are analogous to the confusion and diffusion properties of a good cryptosystem [2-5]. As a response to this concern, a large number of chaotic cryptosystems have been proposed. An image scrambling encryption algorithm of pixel bit based on chaos map is proposed by Ye [6], in which each pixel is converted into binary numbers of 8 bits. However, Zhao pointed out that Ye's cryptosystem can not resist chosen-plaintext attack and known-plaintext attack [7]. A chaos-based image encryption algorithm is presented by Guan [8], which position scrambling is based on Arnold map and Chen's chaotic system is used in pixel mixing. Chen proposed a symmetric image encryption scheme based on 3D chaotic cat maps. In the algorithm, three-dimensional Arnold map is used in image scrambling, and the mixing method is similar to cipher block chaining (CBC for short). Though the forms of chaotic image encryption algorithms are many and different, these algorithms can be concluded to one framework [9-11]. Firstly, the image position is scrambled by chaotic map. Secondly, the pixel is mixed by similar-CBC mode. Finally, repeat the former two steps. However, the application of CBC makes algorithm steps become serial mode which cannot meet timeliness requirements in large data. The parallel mode can compensate the drawback of the problem mentioned above and can increase computing speed. A parallel encryption algorithm for color images based on Lorenz chaotic sequences is proposed by Wang [12], which encrypts three primary colors with parallel mode. In the algorithm, every primary color is encrypted by conventional encryption mode which is only suitable for color image and its application is limited. Zhou proposed a parallel image encryption algorithm based on discrete chaotic map [13], in which the original image is divided into several sub-images and

each sub-image is encrypted by advanced encryption standards algorithm (AES for short), but the short period and weak statistical characteristics of S-box is well known. A new image encryption method [14]: parallel sub-image encryption with hyper chaos is pro- posed by Mirzaei. In the algorithm, the original image is divided into four sub-images according to the up and down or so. Each sub-image is scrambled by logistic sequences, then the while image is permuted and CBC mode is applied in every sub-image, which has a high encryption speed, a large key space and weak difference characteristics. To overcome the drawbacks of the existing parallel encryption algorithm, we present a self-adaptive parallel encryption algorithm based on discrete 2D-Logistic map, which can meet real-time requirement with fast calculating speed. And the security is verified by differential analysis, correlation analysis, sensitivity test and information entropy analysis.

2. Self-Adaptive Parallel Encryption Algorithm Based on Discrete 2D-Logistic Map

2.1. Discrete 2D-Logistic Map

With the development of technology, more and more fields are related to discrete-time multivariable system which mathematical model is deviation equation. Those discrete-time multivariable systems are called space discrete dynamic system [15], that is, 2D-discrete dynamic system denoted as

$$x_{m+1,n} + \omega \cdot x_{m,n+1} = f\left(\mu, (1+\omega)\cdot x_{m,n}\right) \tag{1}$$

where $f()$ is a nonlinear function, also known as forcing function. $\Omega \in (-\infty,+\infty)$ is a real number. $\mu > 0$ is a constant. m, n and $x_{m,n}$ are states of space system. In fact, difference Equation (1) is the discrete form of one-dimensional convection Equation (2).

$$\frac{\partial v}{\partial x} + \omega \cdot \frac{\partial v}{\partial y} = f\left(\mu, (1+\omega)\cdot v\right) \tag{2}$$

Particularly, when $n = n_0$, $\omega = 0$, Equation (2) is denoted as

$$x_{m+1,n_0} + 0 \cdot x_{m,n_0+1} = f\left(\mu, (1+0)\cdot x_{m,n_0}\right) \tag{3}$$

Because n is constant n_0, Equation (3) can be expressed as follows:

$$x_{m+1} = f\left(\mu, x_m\right) \tag{4}$$

Therefore, 1D-discrete dynamic system Equation (4) is a special case of 2D-discrete dynamic system Equation (1). When $f\left(\mu, x_m\right) = \mu \cdot x_m \cdot (1 - x_m)$, Equation (4) is expressed as

$$x_{m+1} = f\left(\mu, x_m\right) = \mu \cdot x_m \cdot (1 - x_m) \tag{5}$$

This is classic 1D-Logistic map. When

$$f\left(\mu, (1+\omega)x_{m,n}\right) = 1 - \left(\mu(1+\omega)x_{m,n}\right)^2,$$

2D-discrete dynamic system is denoted as:

$$x_{m+1,n} + \omega x_{m,n+1} = 1 - \left(\mu(1+\omega)x_{m,n}\right)^2 \tag{6}$$

Study shows that the system Equation (6) is chaotic state with $2 > \mu >= 1.55$ and $\omega \in (-1,1)$, and has two iteration variables which is known as 2D-Logistic map.

There are two control parameters in space chaotic system. Space chaotic orbits are extremely sensitive to the change of control parameters. When $\mu = 1.96$, $\omega = -0.05$, the chaotic behavior of space chaotic system is described as **Figure 1**.

2.2. Demands of Parallel Encryption Algorithm

Parallel encryption algorithm refers to encrypting one image simultaneously by two or more processing elements (*PE* for short). Each *PE* has its separate memory space and computing resource. One superior parallel encryption mode must meet the following demands [13]:

1) Good diffusion effect

Small changes in plain text or key have a big influence on cipher text is called diffusion which is known as avalanche effect. Generally CBC mode is used to realize fully diffusing in existing encryption algorithms. However, the application of CBC makes algorithm steps become serial mode. Therefore, the encryption algorithm is not only parallel operation but also can achieve fully diffusing.

2) Load balance of computing quantity

Computing time depends on the *PE* with the longest running time. Therefore, the calculated quantity of every *PE* should be equal for a good parallel encryption algorithm.

3) Critical section management

Two or more *PE* may read and write one memory area in parallel operation, and this memory area is called

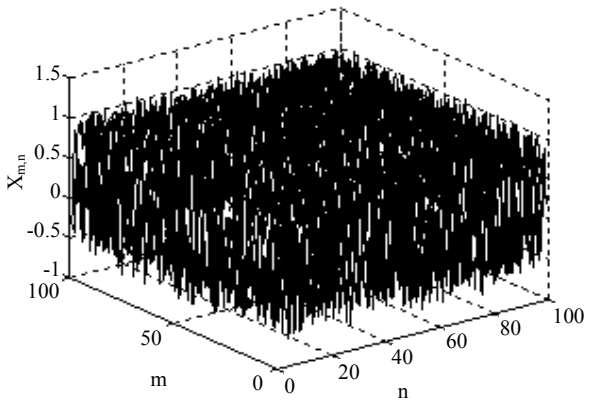

Figure 1. Chaotic behavior of space chaotic system.

critical section. Therefore, one good parallel algorithm should avoid this situation.

2.3. Inter-Block Encryption Algorithm

It is assumed that one original image of size $M \times M$ is encrypted by r processing elements $PE_1, PE_2, \cdots, PE_r$, and meets the divisibility of r dividing M. The encryption process is as follows:

Step 1. Key generation. The binary sequence $b_1 b_2 b_3 \cdots b_n$ is generated according to user key, where the binary values 0 and 1 represent two distribution forms of processing elements, and the number of binary sequence is cycle number.

Step 2. Image segmentation. The plain-text matrix is divided into r blocks. If $b_n = 0$, PE_i encrypts the pixels of matrix block which is from the $\left(\left((i-1)M + r\right)/r\right)$th row to the (iM/r)th row of plain-text matrix. If $b_n = 1$, PE_i encrypts the pixels of matrix block which is from the $\left(\left((i-1)M + r\right)/r\right)$th column to the (iM/r)th column of plain-text matrix, where $i = 1, 2, \cdots, r$.

Step 3. Each matrix block is scrambled and mixed by one processing element.

Step 4. Go to Step 1 until the end of the loop.

2.4. Encryption Algorithm in Block

Assumed that the original image is P, and the pixel matrix is denoted as

$$P = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,M} \\ \vdots & \vdots & \ddots & \vdots \\ p_{M,1} & p_{M,2} & \cdots & p_{M,M} \end{pmatrix} \tag{7}$$

where $i \in [1, M], j \in [1, M]$. A numerical matrix K of size $M \times M$ is produced according to 2D-discrete dynamical system, which is represented in Equation (8).

$$K = \begin{pmatrix} k_{1,1} & k_{1,2} & \cdots & k_{1,M} \\ \vdots & \vdots & \ddots & \vdots \\ k_{M,1} & k_{M,2} & \cdots & k_{M,M} \end{pmatrix} \tag{8}$$

where $i \in [1, M], j \in [1, M]$. In the following, the steps of the proposed algorithm are described in detail.

Step 1. The distribution form of processing elements is defined according to binary sequences $b_1 b_2 b_3 \cdots b_n$.

When $b_n = 0$, the numerical matrix K is divided into r blocks in row direction. The ith numerical matrix block K_i^0 is denoted as

$$\xrightarrow{\text{Equivalent to}} \begin{pmatrix} k_1 & k_2 & \cdots & k_M \\ \vdots & \vdots & \ddots & \vdots \\ k_{\frac{M^2}{r}-M+1} & k_{\frac{M^2}{r}-M+2} & \cdots & k_{\frac{M^2}{r}} \end{pmatrix}_{\frac{M}{r} \times M}$$

$$K_i^0 = \begin{pmatrix} k_{(i-1)\frac{M}{r}+1,1} & \cdots & k_{(i-1)\frac{M}{r}+1,M} \\ \vdots & \ddots & \vdots \\ k_{i\frac{M}{r},1} & \cdots & k_{i\frac{M}{r},M} \end{pmatrix}_{\frac{M}{r} \times M} \tag{9}$$

Then the numerical matrix block K_i^0 is transformed into ergodic matrix Q_i^0 in Equation (10).

$$Q_i^0 = \begin{pmatrix} q_1 & q_2 & \cdots & q_M \\ \vdots & \vdots & \ddots & \vdots \\ q_{\frac{M^2}{r}-M+1} & q_{\frac{M^2}{r}-M+2} & \cdots & q_{\frac{M^2}{r}} \end{pmatrix}_{\frac{M}{r} \times M} \tag{10}$$

where $k_{q_1} < k_{q_2} < \cdots < k_{q_{\frac{M^2}{r}-M+1}} < \cdots < k_{q_{\frac{M^2}{r}}}$.

When $b_n = 1$, the numerical matrix K is divided into r blocks in column direction. The ith numerical matrix block K_i^1 is denoted as

$$K_i^1 = \begin{pmatrix} k_{1,(i-1)\frac{M}{r}+1} & \cdots & k_{1,i\frac{M}{r}} \\ k_{2,(i-1)\frac{M}{r}+1} & \cdots & k_{2,i\frac{M}{r}} \\ \vdots & \ddots & \vdots \\ k_{M,(i-1)\frac{M}{r}+1} & \cdots & k_{M,i\frac{M}{r}} \end{pmatrix}_{M \times \frac{M}{r}}$$

$$\xrightarrow{\text{Equivalent to}} \begin{pmatrix} k_1 & \cdots & k_{\frac{M}{r}} \\ k_{\frac{M}{r}+1} & \cdots & k_{\frac{2M}{r}} \\ \vdots & \ddots & \vdots \\ k_{\frac{M^2}{r}-\frac{M}{r}+1} & \cdots & k_{\frac{M^2}{r}} \end{pmatrix}_{M \times \frac{M}{r}} \tag{11}$$

Then the numerical matrix block K_i^1 is converted into ergodic matrix Q_i^1 in Equation (12).

$$Q_i^1 = \begin{pmatrix} q_1 & \cdots & q_{\frac{M}{r}} \\ q_{\frac{M}{r}+1} & \cdots & q_{\frac{2M}{r}} \\ \vdots & \ddots & \vdots \\ q_{\frac{M^2}{r}-\frac{M}{r}+1} & \cdots & q_{\frac{M^2}{r}} \end{pmatrix}_{M \times \frac{M}{r}} \tag{12}$$

where $k_{q_1} < k_{q_2} < \cdots < k_{q_{\frac{M}{r}}} < \cdots < k_{q_{\frac{M^2}{r}-\frac{M}{r}+1}} < \cdots < k_{q_{\frac{M^2}{r}}}$.

Step 2. Position scrambling. We assume that the ith matrix block P_i is scrambled according to ergodic matrix in PE_i.

If $b_n = 0$, the ith matrix block P_i^0 is scrambled according to ergodic matrix Q_i^0, after scrambling the ma-

trix is denoted as $P_i^{0\prime}$ in Equation (13).

$$P_i^0 = \begin{pmatrix} p_{(i-1)\frac{M}{r}+1,1} & p_{(i-1)\frac{M}{r}+1,2} & \cdots & p_{(i-1)\frac{M}{r}+1,M} \\ \vdots & \vdots & \ddots & \vdots \\ p_{i\frac{M}{r},1} & p_{i\frac{M}{r},2} & \cdots & p_{i\frac{M}{r},M} \end{pmatrix}_{\frac{M}{r}\times M}$$

$$\xrightarrow{\text{Equivalent to}} \begin{pmatrix} p_1 & p_2 & \cdots & p_M \\ \vdots & \vdots & \ddots & \vdots \\ p_{\frac{M^2}{r}-M+1} & p_{\frac{M^2}{r}-M+2} & \cdots & p_{\frac{M^2}{r}} \end{pmatrix}_{\frac{M}{r}\times M}$$

$$P_i^{0\prime} = \begin{pmatrix} p_{q_1} & p_{q_2} & \cdots & p_{q_M} \\ \vdots & \vdots & \ddots & \vdots \\ p_{q_{\frac{M^2}{r}-M+1}} & p_{q_{\frac{M^2}{r}-M+2}} & \cdots & p_{q_{\frac{M^2}{r}}} \end{pmatrix}_{\frac{M}{r}\times M} \quad (13)$$

If $b_n = 1$, the ith matrix block P_i^1 is scrambled according to ergodic matrix Q_i^1, and after scrambling the matrix is denoted as $P_i^{1\prime}$ in Equation (14).

$$P_i^1 = \begin{pmatrix} p_{1,(i-1)\frac{M}{r}+1} & \cdots & p_{1,i\frac{M}{r}} \\ p_{2,(i-1)\frac{M}{r}+1} & \cdots & p_{2,i\frac{M}{r}} \\ \vdots & \ddots & \vdots \\ p_{M,(i-1)\frac{M}{r}+1} & \cdots & p_{M,i\frac{M}{r}} \end{pmatrix}_{M\times\frac{M}{r}}$$

$$\xrightarrow{\text{Equivalent to}} \begin{pmatrix} p_1 & \cdots & p_{\frac{M}{r}} \\ p_{\frac{M}{r}+1} & \cdots & p_{\frac{2M}{r}} \\ \vdots & \ddots & \vdots \\ p_{\frac{M^2}{r}-\frac{M}{r}+1} & \cdots & p_{\frac{M^2}{r}} \end{pmatrix}_{M\times\frac{M}{r}}$$

$$P_i^{1\prime} = \begin{pmatrix} p_{q_1} & \cdots & p_{q_{\frac{M}{r}}} \\ p_{q_{\frac{M}{r}+1}} & \cdots & p_{q_{\frac{2M}{r}+1}} \\ \vdots & \ddots & \vdots \\ p_{q_{\frac{M^2}{r}-\frac{M}{r}+1}} & \cdots & p_{q_{\frac{M^2}{r}}} \end{pmatrix}_{\frac{M}{r}\times M} \quad (14)$$

Step 3. Image scrambling can destroy correlation between two adjacent pixels, but it cannot change pixel value thus the histogram is the same with original image. Therefore, the scrambled image should be further encrypted. For example, to encrypt the ith scrambled matrix block $P_i^{0\prime}$ with $b_n = 0$, and the result is denoted as C_i^0 in Equation (15).

$$C_i^0 = \begin{pmatrix} c_1 & c_2 & \cdots & c_M \\ \vdots & \vdots & \ddots & \vdots \\ c_{\frac{M^2}{r}-M+1} & c_{\frac{M^2}{r}-M+2} & \cdots & c_{\frac{M^2}{r}} \end{pmatrix}_{\frac{M}{r}\times M} \quad (15)$$

where $c_j = p_{q_j} \oplus c_{j-1}$ Go to step 1 until the end of the loop. The detailed encryption diagram is described in **Figure 2**.

When $n = 1$, the pixel mixing based on chaining structure can make a small change of one pixel have effect on other pixels in the same block. And when $n = 2$, the influence can spread into whole image, which meet the demands of parallel encryption algorithms. In addition, the data volume is exactly same for every processing element which can realize load balance. Finally, there is no problem of critical section.

3. Theoretical Analysis and Experimental Simulation

Experimental analysis of the proposed algorithm in this letter has been done. To estimate the performance of the proposed scheme, we carry out a series of experiments. The experimental simulation is all implemented using Matlab8.0 running on a personal computer with i7 – 3770 s 2.8 GHz∗4 processor. The parameter is selected as $\mu = 1.96$, $\omega = -0.05$.

3.1. Statistical Analysis

The distribution of cipher image is very important because it has to hide the redundancy of original image and should not leak any information about original image or any relation between encrypted image and original image. In statistical simulation, the parameters are selected as $b_1 = 0$, $b_1 = 1$ and $b_1 b_2 = 10$ respectively. The simulation results are described in **Figure 3**. As can be seen form **Figure 3**, if $n = 1$, the proposed encryption in the paper can hide the redundancy of plain image completely, *i.e.*, it can resist statistical attack effectively.

3.2. Differential Analysis

We know that two main cryptographic properties of a good cipher are confusion and diffusion. Confusion means to complicate the dependence of the statistics of cipher image on the statistics of plain image. Diffusion means to spread out the influence of a single plain image symbol over many cipher-image symbols so as to hide the statistical structure of plain image. Differential attack can be avoided effectively through confusion and diffusion. In order to illustrate the performance of resistance on differential attack directly, we change the pixel value

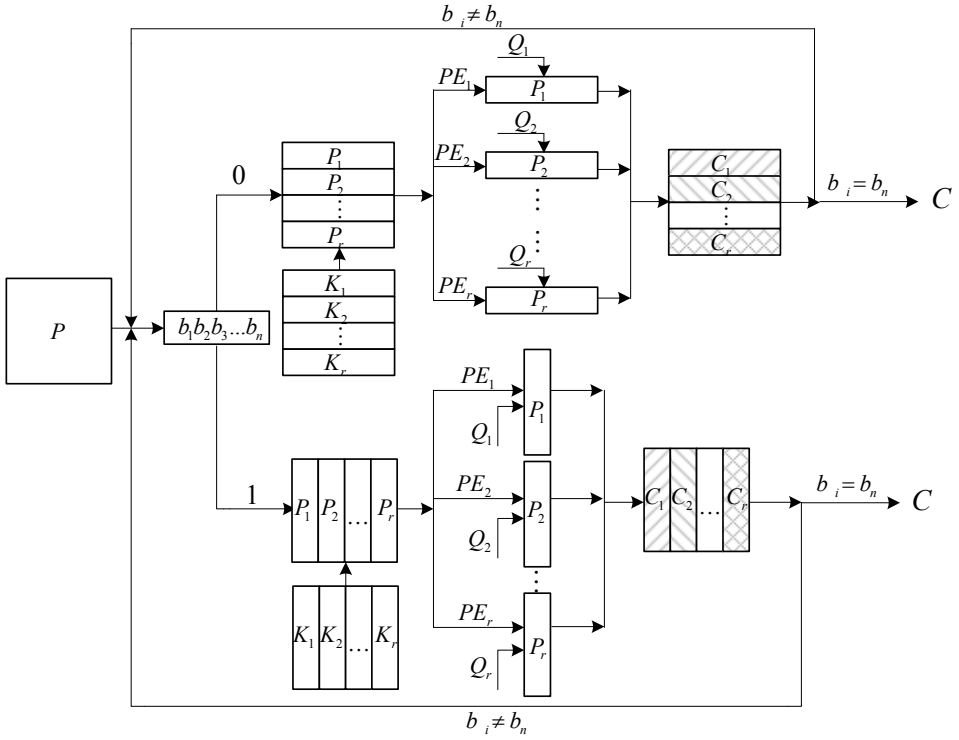

Figure 2. Encryption diagram.

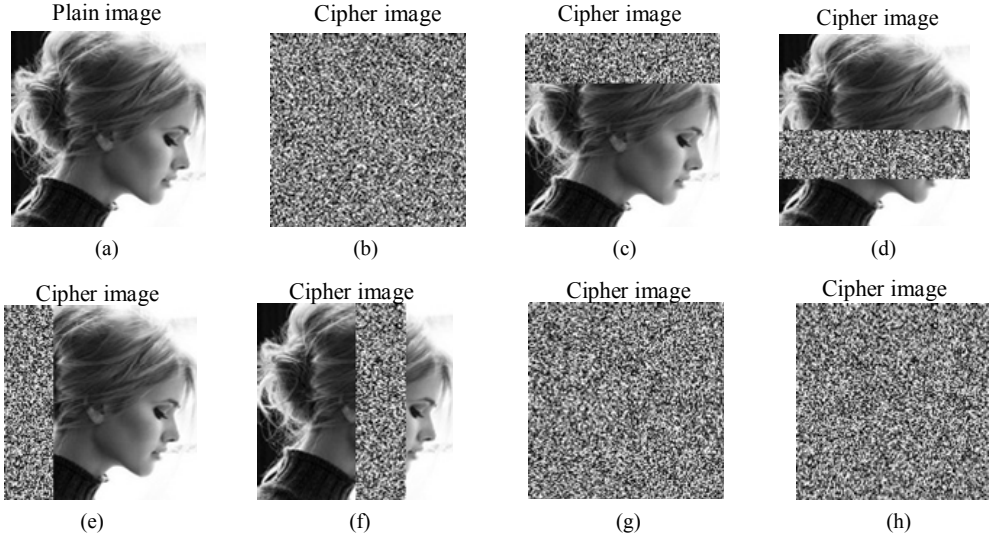

Figure 3. Plain image and cipher image. (a) Plain image. (b) Cipher image with $b_1b_2 = 10$. (c) Cipher image with $b_1 = 0$ for PE_1. (d) Cipher image with $b_1 = 0$ for PE_3. (e) Cipher image with $b_1 = 1$ for PE_1. (f) Cipher image with $b_1 = 1$ for PE_3. (g) Cipher image with $b_1 = 0$ for PE_1, PE_2, PE_3 and PE_4. (h) Cipher image with $b_1 = 1$ for PE_1, PE_2, PE_3 and PE_4.

of **Figure 3(a)** from 113 to 116 in the position (69,98) to illuminate the confusion and diffusion of Mirzaei's scheme and our proposed scheme. Some experiments are performed to study confusion and diffusion properties in **Figure 4**, which there is only a tiny difference in plain-image sequences, and then the difference values of cipher-image sequences are obtained correspondingly. For Mirzaei's algorithm in **Figure 4(a)**, we observe that if there is only a tiny difference in plain image, and there is only some difference in the second block of cipher image correspondingly. However, for our proposed scheme (n = 1) in **Figure 4(b)**, a tiny difference in plain image results in all the other cipher image change in the third block. And the parameter n = 2 in **Figure 4(c)**, all the other cipher text will be influenced accordingly. Therefore, the diffusion and confusion properties of the proposed sche-

me are confirmed with $n = 2$.

In the following, we illustrate the performance of resistance on differential attack numerically. The number of pixels change rate (NPCR for short) and the unified average change intensity (UACI for short) are the significant measure of resistance on differential attack. NPCR and UACI denote the number of pixels change and the unified average change intensity in cipher image respectively, while we change one pixel value in the plain image randomly. Assume that there are two cipher images ($E_{X\times Y}$, $E'_{X\times Y}$) whose plain images only have one different pixel. The pixels in position (i, j) of these two cipher images are denoted as $E(i, j)$ and $E'(i, j)$, respectively. Then one matrix D is defined with the same size of $X \times Y$, and we assume that if $E(i, j) = E'(i, j)$, then $D_{i,j} = 0$, otherwise, $D_{i,j} = 1$. Therefore, NPCR and UACI can be represented as

$$\mathrm{NPCR} = \frac{\sum_{i=0}^{X-1}\sum_{j=0}^{Y-1} D_{i,j}}{X \times Y} \times 100\% \quad (16)$$

$$\mathrm{UACI} = \frac{1}{X \times Y}\left(\sum_{i=0}^{X-1}\sum_{j=0}^{Y-1} \frac{E(i, j) - E'(i, j)}{255}\right) \times 100\% \quad (17)$$

We change the pixel value of **Figure 3(a)** from 81 to 85 in the position (169,236), and then based on Equation (16) and (17) with $n = 2$, NPCR and UACI can be calculated to be 99.63% and 31.26%, respectively. Therefore, one pixel change in plain image can lead to completely change of cipher image. We can draw conclusion that the proposed encryption algorithm can resist differential attack effectively.

3.3. Key Sensitivity Test

An encryption scheme has to be key-sensitive, meaning that a tiny change in keys will cause decryption failure completely. The key sensitivity test is performed in detail according to the following steps:

- **Figure 3(a)** is encrypted with the test key $\mu = 1.96$, $\omega = -0.05$, respectively. And its corresponding decrypted image with the correct key is presented in **Figure 5(a)**.
- Then, the corresponding decrypted image with $\mu = 1.96001$ is presented in **Figure 5(b).**
- Again, when one pixel in PE_3 is changed, the corresponding decrypted image with the correct key is presented in **Figure 5(c)**, while b_1 is chosen as "1".

Finally, when one pixel in PE_3 is changed, the corresponding decrypted image with the correct key is presented in **Figure 5(d)**, while b_1b_2 are chosen as "10", respectively.

Based on the simulation mentioned above, we can draw a conclusion that if there is a tiny change in the key explained before, we can obtain a completely different image from the original image. And if there is a tiny change in the plain image, the decrypted image is completely different form the original image, while b_1b_2 are chosen as "10" respectively. Those simulation results show high key-sensitivity and plain-text sensitivity of the

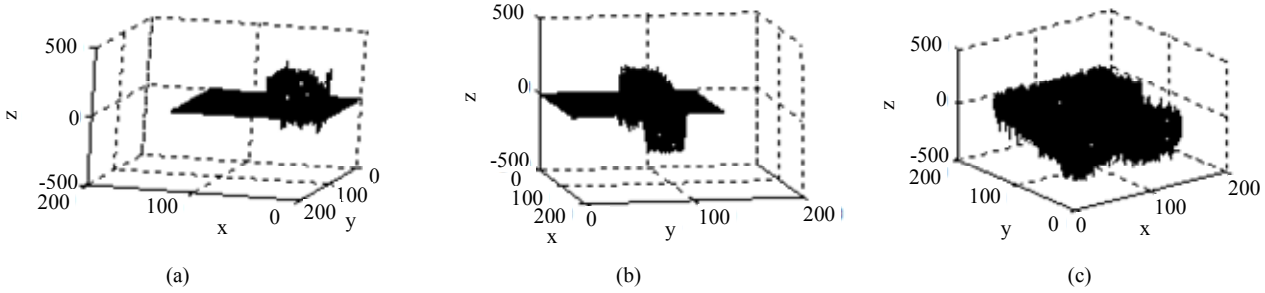

Figure 4. Difference in cipher text changing the pixel value of Figure 3(a) from 113 to 116 in the position (69,98). (a) Mirzaei's. (b) Ours: n = 1. (c) Ours: n >= 2.

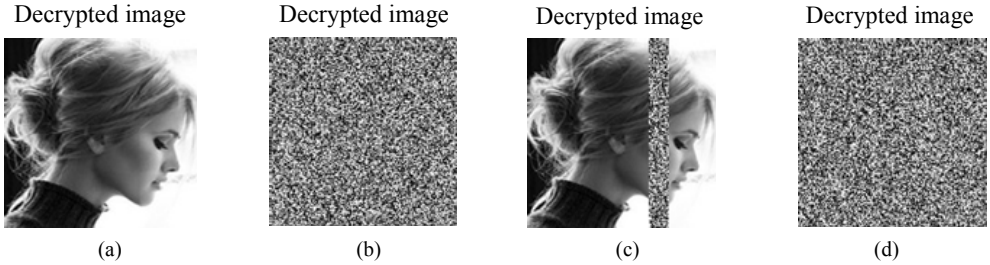

Figure 5. Decrypted image. (a) μ = 1.96. (b) μ = 1.96001. (c) One pixel in PE_3 is changed with b_1 = 1. (d) One pixel in PE_3 is changed with b_1b_2 = 10.

proposed algorithm.

3.4. Analysis of Information Entropy

Entropy $H(x)$ was introduced by Shannon in 1949 firstly and can be obtained through the following formula:

$$H(x) = -\sum_{i}^{n} p(x_i) \log_2 p(x_i) \tag{18}$$

where n is the number of gray scale levels in an image, $p(x_i)$ is the occurrence probability of gray scale in the image and meets the following formula:

$$\begin{cases} 0 \le p(x_i) \le 1, (i = 1, 2, \cdots, n) \\ \sum_{i}^{n} p(x_i) = 1 \end{cases} \tag{19}$$

The entropy value will be 8 for images that are produced totally randomly. The closer the entropy of an algorithm is 8, the less predictable, and thus the more secure the scheme. The cipher-image entropy values have been measured using six plain images in USC-SIPI data base and the results are shown in **Table 1**.

3.5. Analysis of Correlation of Two Adjacent Pixels

Each pixel in the digital image is not independent of other pixels, but has significant correlation. One of purpose for an encryption algorithm is to decrease correlation between adjacent pixels and realize zero co-correlation properties. To show correlation between adjacent pixels in encrypted images, we take **Figure 4(a)** for example to analyze correlation between two vertically adjacent pixels, two horizontally adjacent pixels, and two diagonally adjacent pixels, respectively.

Table 1. The entropy of original gray scale image and its corresponding encrypted ones by the proposed algorithm.

H(x)	7.1.06	7.1.07	7.1.08	7.1.09	7.1.10	7.2.01
Plain image	6.695	5.991	5.053	6.189	5.908	5.641
Cipher image	7.996	7.993	7.999	7.997	7.998	7.998

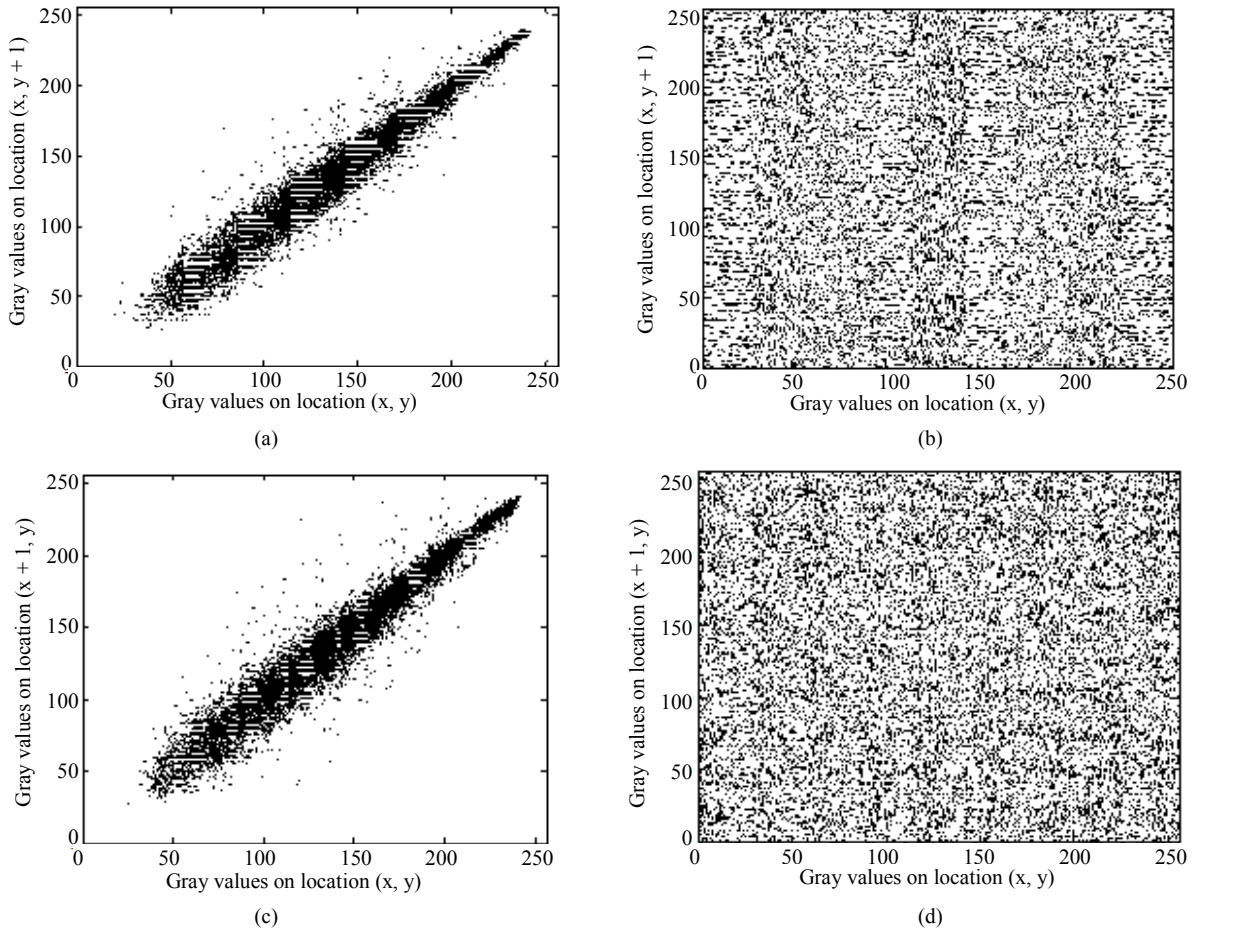

Figure 6. Correlation of two adjacent pixels. (a) The distribution of vertically adjacent pixels in Figure 3(a). (b) The distribution of vertically adjacent pixels in Figure 3(b). (c) The distribution of horizontally adjacent pixels in Figure 3(a). (d) The distribution of horizontally adjacent pixels in Figure 3(b).

Table 2. Comparison of running time and theirs performance.

Algorithms	Running time(s)	Performance
Zhou's	13.7527	Short-period and bad statistical properties
Mirzaei's	10.4251	Bad differential properties
Ours	9.3826 (n = 2)	Better statistical and differential properties with n >= 2

3.6. Running Time Test

Three parallel algorithms are simulated in running time test. They are Zhou's algorithm [13], Mirzaei's algorithm [14] and the proposed algorithm. The encrypted image is gray 21.512 in USC-SIPI image database and the simulation results are described in **Table 2**.

4. Conclusion

In this paper, the security property of a class of image encryption algorithms based on chaotic maps bas been discussed. And some findings on the security problem of the existing parallel encryption algorithms have been reported. Based on discrete 2D-Logistic map, a self-adaptive parallel encryption algorithm has been developed according to the position scrambling and diffusion of multi-direction in variable space of spatial chaos. The results of numerical analysis show that the proposed algorithm can meet the three requirements of parallel operation in image encryption and the real-time requirement in transmission processes. The security has been proved by theoretical analysis and simulation results.

5. Acknowledgements

The authors are greatly indebted to anonymous reviewers for their valuable comments and suggestions. This research is supported in part by the National Natural Science Foundation of China under Grant No. 60874091, the Six Projects Sponsoring Talent Summits of Jiangsu Province under Grant No. SJ209006, the Foundation for Doctoral Program of High Education of China under Grant No. 20103223110003, the MOE Research in the Humanities and Social Sciences Planning Fund of China (Grant No. 12YJAZH120), and the Postgraduate Scientific Innovation Project for Universities of Jiangsu Province under Grant No. CXZZ11-0401.

REFERENCES

[1] X. Ma, C. Fu and W. M. Lei, "Novel Chaos-Based Image Encryption Scheme with an Improved Permutation Process," *International Journal of Advancements in Computing Technology*, Vol. 3, No. 5, 2011, pp. 223-233.

[2] Z. Z. Wang and Q. Y. Han, "Finite-Time Chaos Synchronization of Unified Chaotic System with Uncertain Parameters," *Communications in Nonlinear Science and Numerical Simulation*, Vol. 14, No. 5, 2009, pp. 2239-2247.

[3] H. Wang, Z. Z. Han, Q. Y. Xie, *et al*., "Finite-Time Chaos Control of Unified Chaotic Systems with Uncertain Parameters," *Communications in Nonlinear Science and Numerical Simulation*, Vol. 55, No. 4, 2009, pp. 323-328.

[4] H. P. Lu and X. W. Wang, "A New Spatiotemporally Chaotic Cryptosystem and Its Security and Performance Analyses," *Chaos*: *An Interdisciplinary Journal of Nonlinear Science*, Vol. 14, No. 3, 2004, pp. 617-629.

[5] Y. Yang and X. F. Liao, "Cryptanalysis and Improvement on a Block Cryptosystem Based on Iteration a Chaotic Map," *Physics Letters A*, Vol. 363, No. 4, 2007, pp. 277-281.

[6] G. D. Ye, "Image Scrambling Encryption Algorithm of Pixel Bit Based on Chaos Map," *Pattern Recognition Letters*, Vol. 31, No. 5, 2010, pp. 347-354.

[7] L. Zhao and A. Adhikari, "On the Security Analysis of an Image Scrambling Encryption of Pixel Bit and Its Improved Scheme Based on Self-Correlation Encryption," *Communications in Nonlinear Science and Numerical Simulation*, Vol. 17, No. 8, 2012, pp. 3303-3327.

[8] Z. Guan, F. Huang and W. Guan, "Chaos-Based Image Encryption Algorithm," *Physics Letters A*, Vol. 346, No. 1, 2005, pp. 153-157.

[9] G. R. Chen, "A Symmetric Image Encryption Scheme Based on 3D Chaotic Cat Maps," *Chaos*, *Solitons and Fractals*, Vol. 21, No. 7, 2004, pp. 749-761.

[10] P. Li, Z. Li and W. A. Halang, "A Multiple Pseudorandom-Bit Generator Based on a Spatiotemporal Chaotic Map," *Physics Letters A*, Vol. 349, No. 6, 2006, pp. 467-473.

[11] A. Kanso and M. Ghebleh, "A Novel Image Encryption Algorithm Based on a 3D Chaotic Map," *Communications in Nonlinear Science and Numerical Simulation*, Vol. 17, No. 7, 2012, pp. 2943-2959.

[12] Y. Wang and C. Y. Han, "A Parallel Encryption Algorithm for Color Images Based on Lorenz Chaotic Sequences," *Conference Proceedings on* 6*th World Congress on Intelligent Control and Automation*, Dalian, 21-23 June 2006, pp. 9744-9747.

[13] Q. Zhou, K. W. Wong and X. F. Liao, "Parallel Image Encryption Algorithm Based on Discretized Chaotic Map," *Chaos Soliton & Fractals*, Vol. 29, No. 11, 2008, pp. 1081-1092.

[14] O. Mirzaei, M. Yaghoobi and H. Irani, "A New Image Encryption Method: Parallel Sub-Image Encryption with Hyper Chaos," *Nonlinear Dynamics*, Vol. 67, No. 1, 2012, pp. 557-566.

[15] F. Y. Sun and S. T. Liu, "Cryptographic Pseudo-Random Sequence from the Spatial Chaotic Map," *Chaos Solitons & Fractals*, Vol. 41, No. 5, 2009, pp. 2216-2219.

