Digital image encryption with chaotic map lattices*

Sun Fu-Yan(孙福艳)† and Lü Zong-Wang(吕宗旺)

College of Information Science and Engineering, Henan University of Technology, Zhengzhou 450001, *China*

(Received 5 September 2010; revised manuscript received 29 November 2010)

This paper proposes a secure approach for encryption and decryption of digital images with chaotic map lattices. In the proposed encryption process, eight different types of operations are used to encrypt the pixels of an image and one of them will be used for particular pixels decided by the outcome of the chaotic map lattices. To make the cipher more robust against any attacks, the secret key is modified after encrypting each block of sixteen pixels of the image. The experimental results and security analysis show that the proposed image encryption scheme achieves high security and efficiency.

Keywords: image encryption, image decryption, chaotic map lattice, cryptography, spatiotemporal chaotic system

PACS: 05.45.–a, 05.45.Gg **DOI:** 10.1088/1674-1056/20/4/040506

1. Introduction

In the last decade, security communication by applying chaotic maps has attracted a great deal of attention. Especially with the rapid distribution and application of digital images, chaos-based image encryption has become a significant topic.

A cryptosystem is a useful tool for information security. As we know, a chaotic system has many interesting features, such as the sensitivity to the initial condition and control parameter, ergodicity and mixing property, which have tight relationships with the requirements of cryptography.[1−3] In recent years, many researchers have proposed and analysed a lot of chaotic cryptosystems,[4−7] and they have proven that chaos-based ciphers have some exceptionally good properties in many concerned aspects regarding security, complexity, speed, computing power, etc. However, many of them are not suitable to encrypt images directly, due to some intrinsic features of image such as bulk data capacity and high correlation among pixel. So far, for realizing secure image transmission over the Internet and through wireless networks, a number of chaos-based image encryption schemes have been proposed,[8−15] which are briefly discussed here. In Ref. [5], an image encryption scheme which utilizes the SCAN language to encrypt and compress an image simultaneously was proposed. In Ref. [4], a chaotic Kolmogorov-flow-based image encryption algorithm was designed. In this scheme, the whole image was taken as a single block and permuted through a key-controlled chaos system based on Kolmogorov flow, in order to confuse the pixels, a shift register pseudo random generator was also applied. Further, Chen *et al.*[11] have proposed a symmetric image encryption in which a two-dimensional chaotic cat map is generalized to three-dimensional (3D) to shuffle the positions of the image pixels and another chaotic map is used to confuse the relationship between the encrypted and its original image. Spatiotemporal chaotic system is a spatially extended system, which can exhibit spatiotemporal chaos, namely, nonlinear dynamics in both space and time, it may fully manifest all the advantages of chaotic cryptography.[16−23] Recently, coupled map lattices (CML)-based spatiotemporal chaotic system was proposed for cryptography of a self-synchronizing stream cipher.[18,21] It has been verified that spatiotemporal chaotic systems maintain much longer periodicity in digitalization and gain good performance in cryptography,[20] and then the communication with the CMLs is more secure than the communication with a single map.

In this paper, a red-green-blue (RGB) colour image encryption scheme is proposed, in which a CML consisting of logistic maps is used to perform encryption and decryption. In the proposed encryption process, eight different types of operations are used to encrypt the pixels of an image and which operation will

*Project supported by the National Natural Science Foundation of China (Grant Nos. 61001099 and 10971120) and the Foundation for the Author of National Excellent Doctoral Dissertation of China (Grant No. 200444).

†Corresponding author. E-mail: fuyan_sun@126.com

© 2011 Chinese Physical Society and IOP Publishing Ltd

be used for particular pixels is decided by the outcome of the CML, so multiple keystreams are generated from the CMLs by applying simple algebraic computations. To make the cipher more robust against any attack, after each encryption of a block of sixteen pixels, the secret key is modified. The cryptographic properties of the keystream are investigated numerically, the results show that the properties are satisfactory. The security of the cipher is evaluated by analysing its known-plaintext attack, differential attack, brute-force attack and chosen-plaintext/ciphertext attack. It is shown that the proposed cipher has higher security, lower computation cost and other similar good cryptographic properties. The rest of this paper is organized as follows. Section 2 describes the chaotic map lattice and the proposed image encryption method. In Section 3, we prove our method with a computer experiment by encrypting a colour image and study sensitivity of our cryptosystem to secret keys. In Section 4, the security of the new scheme is evaluated. In Section 5 we conclude the paper.

2. Image encryption based on spatiotemporal chaos

2.1. A spatiotemporal chaotic system

In this paper, CML is adopted as the basic model of a spatiotemporal chaotic system. It consists of nonlinear maps located on the lattice sites, named as local maps. Each local map is coupled with other local maps in terms of certain coupling rules. Because of the intrinsic nonlinear dynamics and the diffusion which is caused by the spatial coupling among the local maps, a CML can exhibit spatiotemporal chaos.[23] Further, by adopting various local maps and coupling methods, various CMLs can be constructed.

A CML consisting of logistic map is adopted to construct the spatiotemporal chaotic system. Its mathematical representation is given by

$$x_{n+1}^j = (1-\varepsilon)f(x_n^j) + \varepsilon f(x_n^{j-1}), \tag{1}$$

where x_n^j represents the state variable for the j-th site ($j = 1, 2, \ldots, L$, L is the lattice site in the CML), n is the discrete time index and ε is the coupling constant. In general, the periodic boundary condition $x_n^0 = x_n^L$ is assumed and the mapping function $f(x)$ is chosen to be the logistic map $f(x) = ax(1-x), a \in (0, 4]$.

2.2. The proposed method

The RGB image represents colour by the RGB colour model, which is an additive model that the three primary colours red, green and blue are combined to produce other colours. The integer numbers between [0, 255] are used to describe these gray levels. The final displayed colour is determined by these three components. In digital computer communication, each colour level for red, green and blue, is usually represented by an 8-bit value. The RGB image can be regarded as an extended gray-level image. When the encryption of an RGB image is required, one can firstly decompose the red $g_{\mathrm{r}}(x, y)$, green $g_{\mathrm{g}}(x, y)$ and blue $g_{\mathrm{b}}(x, y)$ components of each pixel and represent their values in binary mode described as follows:

$$g_{\mathrm{r}}(x, y) = p_{(7)}p_{(6)}p_{(5)}p_{(4)}p_{(3)}p_{(2)}p_{(1)}p_{(0)}, \tag{2}$$

$$g_{\mathrm{g}}(x, y) = p_{(7)}p_{(6)}p_{(5)}p_{(4)}p_{(3)}p_{(2)}p_{(1)}p_{(0)}, \tag{3}$$

$$g_{\mathrm{b}}(x, y) = p_{(7)}p_{(6)}p_{(5)}p_{(4)}p_{(3)}p_{(2)}p_{(1)}p_{(0)}. \tag{4}$$

Then the image is transformed into the bitstream of three channels. It is divided into blocks of 32-bit size and the block of data at index i is denoted as $D_{\mathrm{r}}(i), D_{\mathrm{g}}(i), D_{\mathrm{b}}(i)$ for red, green and blue, respectively.

Since we need to encrypt red, green and blue channels for each pixel of an RGB image, we introduce CML mode to serve as the keystream generator, which is described by

$$x_{n+1}^j = (1-\varepsilon)f(x_n^j, a_j) + \varepsilon f(x_n^{j-1}, a_{j-1}), \tag{5}$$

$$f(x_n^j, a_j) = (3.9 + 0.1a_j)x_n^j(1 - x_n^j), \tag{6}$$

where $j = 1, 2, \ldots, 8$, $x_n^0 = x_n^8$, ε is set to 0.99 throughout this paper. The system parameter $a_j \in [0, 1]$ and initial condition of f, namely, $x_0^j \in (0, 1)$ serve as the secret key, denoted as $a = \{a_1, a_2, \ldots, a_8\}$ and $x_0^j = \{x_0^1, x_0^2, \ldots, x_0^8\}$, respectively.

Considering the period of the keystream, which equals about 10^7,[20] the lattice site number is adopted as 8. Thus, the system can have eight output variables: $x_n^1, x_n^2, \ldots, x_n^8$, simultaneously, they are then further adopted to generate keystream K_n^j by applying simple algebraic computations given by

$$K_n^j = \mathrm{int}(x_n^j \times 2^u) \bmod 2^v, \tag{7}$$

since the number of significance bits of the binary representation of a number in a computer is 52, u and v are chosen as 52 and 32, respectively, v is the last 32 bits extracted from the 52 bits, considering the efficiency and security, we fix v as 32.

The encryption algorithm includes the following steps.

(i) Read the three consecutive 32 bits from the image file. These three 32 bits represent the value of the red, green and blue colour respectively and together form four pixels of the image.

(ii) Generates N by using the following formula:

$$N = \mathrm{mod}(x_n^j \times 10^{13}, 8). \tag{8}$$

As $N \in [0, 7]$, so from Table 1, we select the corresponding group used to perform encryption operation if N equals to the group number of sequence of the group.

Table 1. Different operations for RGB image encryption and decryption.

group No.	operation for encryption/decryption
0	NOT operation, i.e. invert the bits of all three RGB bytes
1	$D_{\mathrm{r}}(i) \oplus K_n^1, D_{\mathrm{g}}(i) \oplus K_n^2, D_{\mathrm{b}}(i) \oplus K_n^3$
2	encryption
	$S_{\mathrm{r}}(i) = (D_{\mathrm{r}}(i) + K_n^1) \mathrm{mod}\ 2^v, S_{\mathrm{g}}(i) = (D_{\mathrm{g}}(i) + K_n^2) \mathrm{mod}\ 2^v, S_{\mathrm{b}}(i) = (D_{\mathrm{b}}(i) + K_n^3) \mathrm{mod}\ 2^v$
	decryption
	$D_{\mathrm{r}}(i) = (S_{\mathrm{r}}(i) - K_n^1) \mathrm{mod}\ 2^v, D_{\mathrm{g}}(i) = (S_{\mathrm{g}}(i) - K_n^2) \mathrm{mod}\ 2^v, D_{\mathrm{b}}(i) = (S_{\mathrm{b}}(i) - K_n^3) \mathrm{mod}\ 2^v$
3	encryption
	$\mathrm{NOT}(D_{\mathrm{r}}(i) \oplus K_n^1), \mathrm{NOT}(D_{\mathrm{g}}(i) \oplus K_n^2), \mathrm{NOT}(D_{\mathrm{b}}(i) \oplus K_n^3)$
	decryption
	$[\mathrm{NOT}(D_{\mathrm{r}}(i))] \oplus K_n^1, [\mathrm{NOT}(D_{\mathrm{g}}(i))] \oplus K_n^2, [\mathrm{NOT}(D_{\mathrm{b}}(i))] \oplus K_n^3)$
4	Similar to group 1 except that K_n^4, K_n^5 and K_n^6 are used instead of K_n^1, K_n^2 and K_n^3, respectively.
5	Similar to group 2 except that K_n^4, K_n^5 and K_n^6 are used instead of K_n^1, K_n^2 and K_n^3, respectively.
6	Similar to group 3 except that K_n^4, K_n^5 and K_n^6 are used instead of K_n^1, K_n^2 and K_n^3, respectively.
7	no operation is made.

Here, $S_{\mathrm{r}}(i), S_{\mathrm{g}}(i)$ and $S_{\mathrm{b}}(i)$ are ciphered image, $\oplus$ means bitwise XOR, "int" extracts the integer part of a number and "mod" is the modulus operation.

(iii) We assign different types of operation in eight groups. The generated keystreams are then used to encrypt red, green and blue components by the same rule in each of these groups, simultaneously. In Table 1, we have given the corresponding operations in different groups. We repeat steps (i)–(iii) for next 12 pixels of the image file.

(iv) After encryption of a block of 16 pixels of the image file, one can modify the session keys a_n^j as follows:

$$a_j = 1 - \mathrm{mod}(x_n^j + x_n^{j+2}, 1), \quad j = 1, \ldots, 8. \tag{9}$$

After modifying the secret key in the above manner, we again repeat steps (i)–(iii) until the entire image file is exhausted.

3. Experimental analysis

Experimental analysis of the proposed image encryption algorithm in this paper has been done. Taking an image in 24-bit true colour with 512×512 pixels as an example, the original image is shown in Fig. 1(a), we have also shown the histograms of red, blue and green channels of the original image in Figs. 1(b)–1(d) respectively. The encrypted image by using the key ($a_1 = 0.234567, a_2 = 0.334567, a_3 = 0.814567, a_4 = 0.512345, a_5 = 0.123456, a_6 = 0.234567, a_7 = 0.4534589, a_8 = 0.654321, x_0^1 = x_0^2 = \cdots = x_0^8 = 0.3456789$) is shown in Fig. 1(e) and the histograms of red, blue and green channels of the encrypted image is shown in Figs. 2(a), 2(b) and 2(c) respectively. It is clear from Fig. 2 that the histograms of the encrypted image are fairly uniform and significantly different from the respective histograms of the original image. Figure 3(a) shows the decrypted image using the right key. However, if we make the slight modification in secret key ($a_1 = 0.234567, a_2 = 0.334567, a_3 = 0.814567, a_4 = 0.512345, a_5 = 0.123456, a_6 = 0.234567, a_7 = 0.4534589, a_8 = 0.654321, x_0^1 = x_0^2 = \cdots = x_0^8 = 0.3456788$), the resultant image is shown in Fig. 3(b). So it can be concluded that the chaotic encryption algorithm is sensitive to the keys, a small change of the key will generate a completely different decryption result and cannot obtain the correct original image.

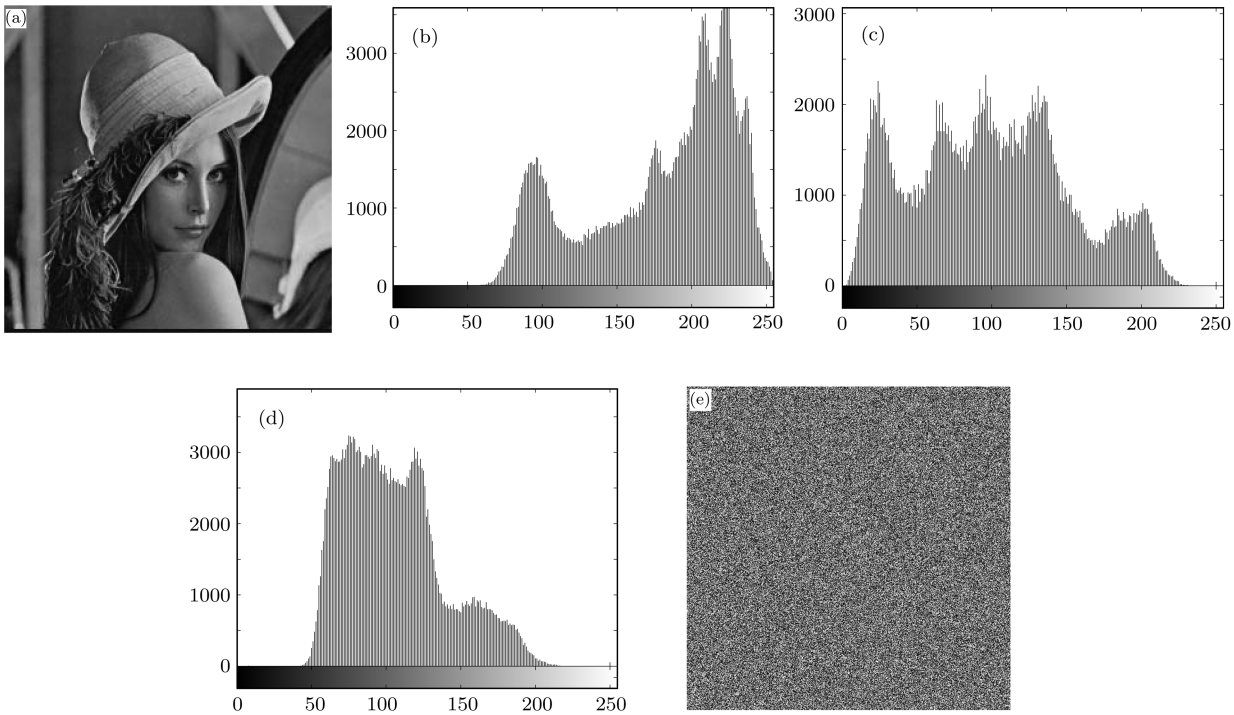


Fig. 1. Panel (a) shows a plain image. Panels (b), (c) and (d) respectively, show the histograms of red, green and blue channels of the plain image shown in frame (a). Frame (e) shows the encrypted image of the plain image.

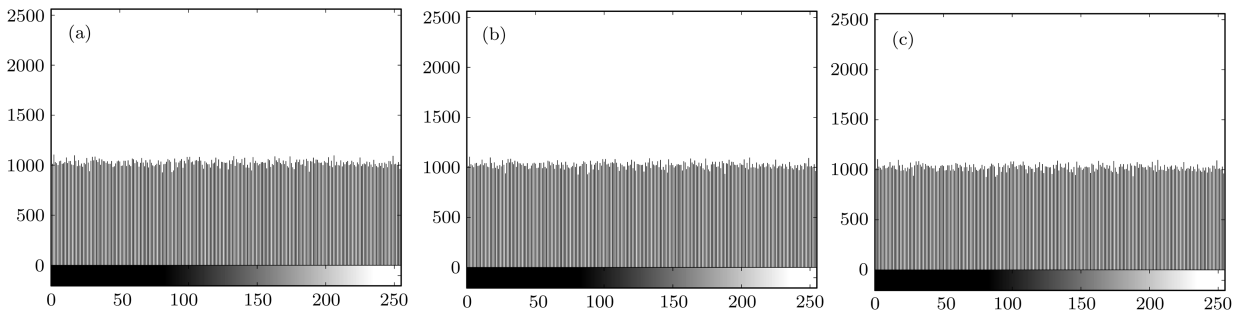


Fig. 2. Panels (a), (b) and (c) show the histogram of red, green and blue channels of the encrypted image, respectively.

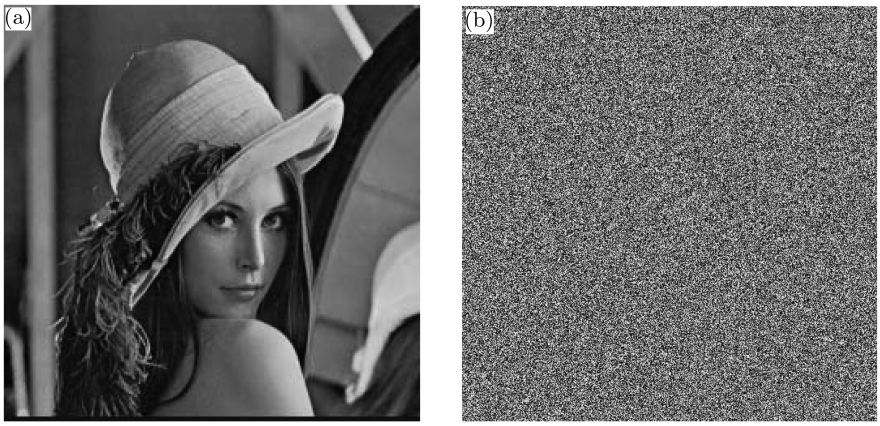


Fig. 3. Decrypted image: (a) decrypted image by correct key, (b) decrypted image by wrong key.

4. Security analysis

A good encryption scheme should resist all kinds of known attacks, it should be sensitive to the secret keys. Some security analysis has been performed on the proposed image encryption scheme, as demonstrated in the following.

4.1. Key space analysis

Key space size is the total number of different keys that can be used in the encryption. A good encryption scheme should be sensitive to the secret keys and the key space should be large enough to make brute-force attacks infeasible. In this algorithm, the initial conditions and parameters can be used as keys. If the precision is 10^{-14}, the key space size can reach to $10^{14\times16}$. One can see that the key space is incomparably large to resist the statistical attacks.

4.2. Correlation analysis

Correlation indicates the strength and direction of a linear relationship between two random variables. In image processing, it is usually employed to investigate the relationship between two adjacent pixels. In order to investigate the confusion effect of the encrypted image, we randomly select 1000 pairs of adjacent pixels in each direction (horizontal, vertical and diagonal) from the original and its encrypted images, we then calculate the correlation coefficients of two adjacent pixels in each component of original image and its encrypted version by using the following formulae:

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

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2, \tag{11}$$

$$\mathrm{cov}(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)), \tag{12}$$

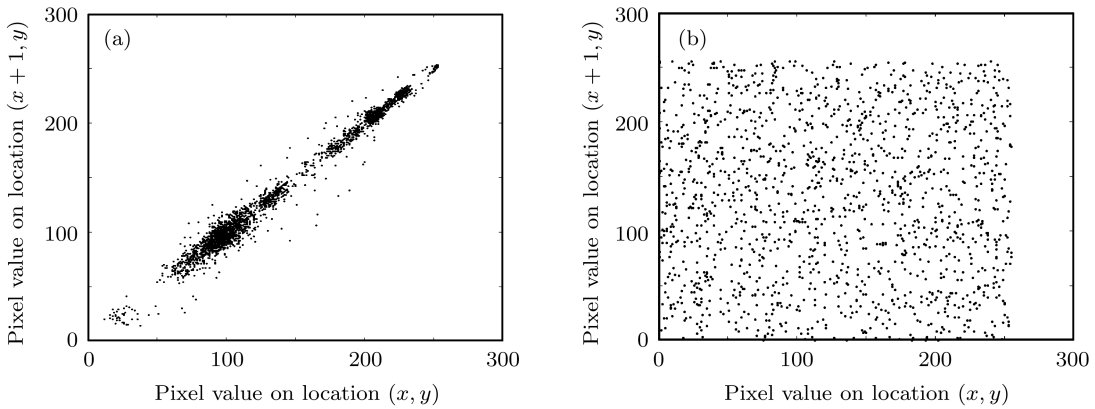
$$r_{xy} = \frac{\mathrm{cov}(x,y)}{\sqrt{D(x)D(y)}}, \tag{13}$$

where x and y are grey-scale values of two adjacent pixels in the image.

The results are shown in Table 2. From the first three columns, one can conclude that there is a strong correlation between adjacent pixels in all directions of each component of the original image, especially for the red component. However, judging from the data in the last three columns, the situation is totally different in the encrypted image as very little correlation is found. The correlation distributions between two vertically as well as horizontally adjacent pixels in the original and encrypted images are shown in Fig. 4. These correlation analyses prove that the chaotic encryption algorithm satisfies zero co-correlation.

Table 2. Correlation coefficient of two adjacent pixels in each component of original image and encrypted image.

	original image			encrypted image		
	red	green	blue	red	green	blue
horizontal	0.97365	0.877451	0.634216	0.00126	–0.00159	–0.00181
vertical	0.964523	0.862451	0.657874	0.00112	0.00134	0.00128
diagonal	0.951567	0.845175	0.610708	0.00106	0.00209	0.00273



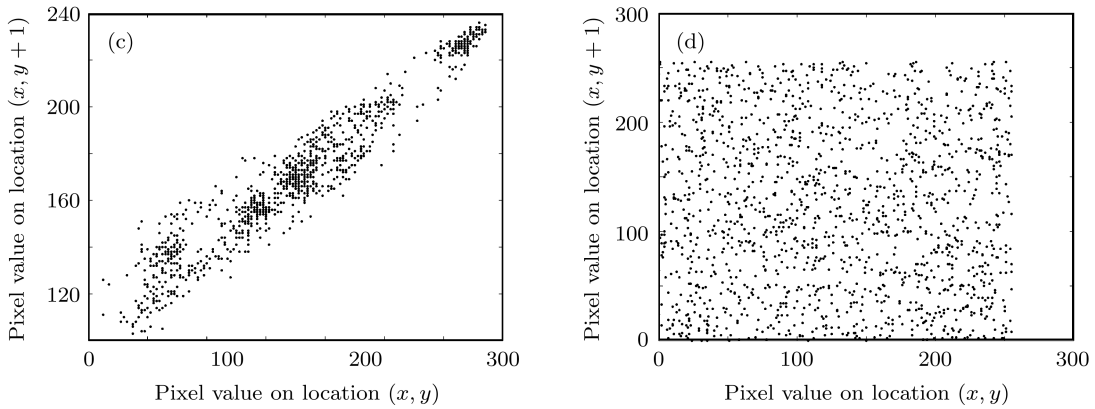


Fig. 4. Correlation of two adjacent pixels: panels (a) and (b) respectively, show the distribution of two horizontally adjacent pixels in the original and encrypted images; and panels (c) and (d) respectively, show the distribution of two vertically adjacent pixels in the original and encrypted images.

4.3. Differential attack

The number of pixels change rate (NPCR) R_{NPCR} was used to test the influence of changing a single pixel in the original image on the whole image encrypted by the proposed algorithm. For the calculation of R_{NPCR}, let us assume two ciphered images C_1 and C_2 whose corresponding original images have only one-pixel difference. The grey-scale values of the pixels of the ciphered image C_1 and C_2 at grid (i, j) are labeled as $C_1(i, j)$ and $C_2(i, j)$, respectively. Take an array D, with the same size as image C_1 or C_2. Then, $D(i, j)$ is determined by $C_1(i, j)$ and $C_2(i, j)$. So, if $C_1(i, j) = C_2(i, j)$ then $D(i, j) = 1$; otherwise, $D(i, j) = 0$. The R_{NPCR} is defined through the following formula:

$$R_{\mathrm{NPCR}} = \frac{\sum_{ij} D(i, j)}{W \times H} \times 100\%, \tag{14}$$

where W and H are the width and height of ciphered image. One obtained R_{NPCR} to be over 99.3% by using the proposed encryption scheme. The result shows that a swiftly change in the original image will result in a significant change in the ciphered image, so the proposed algorithm has a good ability to anti-differential attack.

4.4. Other attacks

Other attack models, such as known-plaintext attack and chosen-plaintext attack, are less effective and therefore they are not considered here because they are only valid in the situation when the same key is used repeatedly. The two models do not work because our encryption scheme is requested to change the secret key frequently. In addition, for a newly introduced attack model, namely, error function attack, it has proven that spatiotemporal chaotic ciphers based on coupled chaotic map lattices perform outstandingly well against error function attack because the key basin is very small.[17,18] Based on the same principle, the proposed scheme is undoubtedly immune to error function attack.

5. Conclusions

A spatiotemporal chaotic system adopting CML consisting of logistic map is proposed to serve as a keystream generator for RGB image encryption and decryption. Multiple keystreams are produced from the CML via eight different types of operations to encrypt the original image and which operation will be used for particular pixels is decided by the outcome of the CML. To make the cipher more robust against any attack, the secret key is modified after encrypting a block of sixteen pixels of the image. Some secure analyses are given to demonstrate that the key space of the new algorithm is large enough to make brute-force attacks infeasible, digital simulations have been carried out with detailed numerical analysis, demonstrating the high security of the new image encryption scheme. Compared to other stream ciphers, the proposed scheme has higher security, faster encryption and lower computation expense as well as other good cryptographic properties.

References

[1] Xiang F and Qiu S S 2008 *Acta Phys. Sin.* **57** 6132 (in Chinese)

[2] Wang X Y and Wang M Y 2008 *Acta Phys. Sin.* **57** 731 (in Chinese)

[3] Sun F Y, Liu S T and Lü Z W 2007 *Chin. Phys.* **16** 3616

[4] Schenier B 1998 *J. Electron Imaging* **7** 318

[5] Bourbakis N and Alexopoulos C 1992 *Pattern Recogn.* **25** 567

[6] Fridrich J 1998 *Int. J. Bifur. Chaos* **8** 1259

[7] Dachselt F and Schwarz W 2001 *IEEE Trans. Circ. Syst.* I **48** 1498

[8] Matthews R 1989 *Cryptologia* **13** 29

[9] Baptista M S 1998 *Phys. Lett.* A **240** 50

[10] Chang C C, Hwang M S and Chen T S 2001 *J. Syst. Software* **58** 83

[11] Chen G and Mao Y B 2004 *Chaos, Solitons and Fractals* **21** 749

[12] Mao Y B, Chen G and Lian S G 2004 *Int. J. Bifur. Chaos* **14** 3613

[13] Zhang L H, Liao X F and Wang X B 2005 *Chaos, Solitons and Fractals* **24** 759

[14] Pareek N K, Vinod P and Sud K K 2005 *Commun. Nonlinear Sci. Numer. Simul.* **10** 715

[15] Xiang T, Liao X, Tang G, Chen Y and Wong K W 2006 *Phys. Lett.* A **349** 109

[16] Li P, Li Z, Halang W A and Chen G 2006 *Phys. Lett.* A **349** 467

[17] Li P, Li Z, Halang W A and Chen G 2007 *Chaos, Solitons and Fractals* **32** 1867

[18] Lu H P, Wang S H, Li X W, Tang G N, Kuang J Y, Ye W P, *et al.* 2004 *Chaos* **14** 617

[19] Garcia P, Parravano A, Cosenza M G, Jimenez J and Marcano A 2002 *Phys. Rev.* E **65** 045201

[20] Wang S H, Liu W R, Lu H P, Kuang J Y and Hu G 2004 *Int. J. Mod. Phys.* B **18** 2617

[21] Wang S H, Kuang J Y, Li J, Luo Y, Lu H P and Hu G 2002 *Phys. Rev.* E **66** 065202(R)

[22] Tang G, Wang S, Lu H and Hu G 2003 *Phys. Lett.* A **318** 388

[23] Kaneko K and Tsuda I 1996 *Complex Systems: Chaos and Beyond* (Tokyo: Asakura)