ICIC Express Letters
Part B: Applications
Volume **8**, Number **2**, February **2017**

 ISSN 2185-2766
pp. **227–236**

NOVEL IMAGE ENCRYPTION METHOD BASED ON DYNAMIC BLOCK TECHNIQUE AND CHAOTIC SYSTEMS

Xiayan Zhang[1], Guoji Zhang[2] and Xuan Li[3]

[1]School of Computer Science and Engineering
South China University of Technology
Higher Education Mega Centre, Guangzhou 510006, P. R. China
scutzhangxiayan@163.com

[2]School of Mathematics
South China University of Technology
No. 381, Wushan Rd., Guangzhou 510641, P. R. China
magjzh@scut.edu.cn

[3]School of Software
Fujian Normal University
Software Building, Fuzhou 350007, P. R. China
jessieli24@163.com

Received July 2016; accepted October 2016

Abstract. *In traditional block image encryption methods, the block size remains fixed and unchanged in each round. In this paper, a novel dynamic block image encryption algorithm is proposed. The block size of each encryption round is dynamically and randomly determined by the logistic map. In the encryption process, bitwise XOR operations and rotating shift operations are employed using keystreams generated from the logistic map and the two-dimensional Henon map. System time is introduced as an initial value when adopting the chaotic maps. Thus, when encrypting two images at different time, the cipher-images are different so that the method can resist known-plaintext attacks and chosen-plaintext attacks. Experiment results show that the proposed method can provide high security and efficiency.*
Keywords: Image encryption, Block cipher, Chaotic systems, Logistic map, Henon map

1. **Introduction.** Image encryption is different from text encryption due to some inherent features, such as bulk data capacity, high correlation among pixels and high redundancy. Thus, most conventional ciphers, such as DES (Data Encryption Standard), AES (Advanced Encryption Standard), and IDEA (International Data Encryption Algorithm), are not suitable for image encryption especially in real time encryption. Therefore, many encryption algorithms [1, 2, 3, 4, 5, 6, 7] were specially designed for images. Among them, chaotic systems are mostly widely used because of several significant features, such as ergodicity, sensitivity to initial conditions, control parameters and random-like behavior, which can be connected with some conventional cryptographic properties of good ciphers, such as confusion and diffusion. Matthews first proposed the chaotic encryption algorithm in 1989 [8] and since then increasing researches of image encryption algorithms based on chaotic systems were proposed [6, 7, 9, 10, 11, 12].

Logistic map is the most widely used chaotic map in image encryption, since it has simple structure but good performance [6, 7, 9, 10]. Other chaotic maps such as tent map [11, 12], and Chebyshev map [13], are also widely used. However, many one-dimensional chaotic systems have short cycle length resulting from the finite precision of computers, making them vulnerable to various attacks [14, 15]. Therefore, hyper-chaotic system has been investigated recently because it has more than one positive Lyapunov exponent, and

has more complex dynamical characteristics, which can extend the cycle length of chaotic systems [16, 17, 18, 19].

Some image encryption schemes [9, 13, 17] are insecure against chosen-plaintext attacks because the keystream generated by the chaotic map is completely depending on the secret key and remains unchanged to any plain-image [20, 21, 22]. To solve this problem, some approaches that generate plain-image dependent keystreams were proposed. [23] presented a block-based image encryption algorithm using wave function and chaotic system. The keystream is dependent on both the plain-image and the secret key. [24] presented a block image encryption algorithm which only adopts the diffusion function. The plain-image is randomly divided into two equal parts by vertical, horizontal, or diagonal directions and the encryption of one part depends on the other part. [25] proposed a new block image encryption scheme based on hybrid chaotic maps and dynamic random growth technique which uses cat map in a securer way and generates keystreams depending on the plain-image. In [26], Zhang et al. introduced the system time and system random identifier as an initial value of the key generation system so that the method is an approximately one-time pad.

In this paper, a chaos-based dynamic block image encryption algorithm is proposed. A new dynamic block cipher mode is proposed that the block size of each encryption round is dynamically and randomly determined by random parameters generated by the logistic map. Thus, the block size is unfixed and not kept the same in different encryption rounds, which can decrease the strong correlation of the same block in different encryption rounds and improve the complexity of the algorithm. The most widely used block cipher mode, CBC (Cipher Block Chaining) mode is applied and the initial block is generated randomly by the logistic map. Two chaotic maps, the logistic map and the two-dimensional Henon map are used to generate keystreams for the bitwise XOR operations and rotating shift operations, respectively. Also, the encryption operation differs when the image pixel number is odd and even. The initial value of the two chaotic maps is the key of the algorithm and system time is introduced to initialize the key to make it time-related. Thus, the encryption differs every time and the method can resist against known-plaintext and chosen-plaintext attacks. Experimental results show that the new scheme has feasibility and security.

The rest of this paper is organized as follows. The logistic map and Henon map used in the algorithm are introduced in Section 2. Section 3 presents the initialization and keystreams generation process, the CBC mode and the encryption and decryption process in detail. Experimental results and security analysis are shown in Section 4. Section 5 draws a conclusion for this paper.

2. The Chaotic Systems Used in the Algorithm.

2.1. Logistic map.

Logistic map is a most widely used one-dimensional chaotic system that has one single control parameter and simple structure. It is described as Equation (1),

$$x_i = \mu x_{i-1}(1 - x_{i-1}), \quad i = 1, 2, 3, \ldots, \tag{1}$$

where x_i is time series, μ is the control parameter and x_0 is the initial value. The system is a chaotic system when parameter $\mu \in [3.9, 4.0]$.

2.2. Henon map.

Henon map is a widely used two-dimensional chaotic system which can be described as Equation (2),

$$\begin{cases} y_i = 1 - a y_{i-1}^2 + z_i \\ z_i = b y_i \end{cases}, \quad i = 1, 2, 3, \ldots, \tag{2}$$

where a, b are control parameters and when $a = 1.4$, $b = 0.3$ the system is a chaotic system.

3. **Dynamic Block Image Encryption and Decryption Scheme.** Many block image encryption algorithms usually divide the plain-image into two or four sub-images of the same size [24, 25]. In our algorithm, the block size is dynamically and randomly determined by the logistic map. First, the logistic map is used to generate n random number $s_1, s_2, \ldots, s_n$ and $bs_i = 2^{s_i}$ is the block size of the i-th round of the encryption. Thus, the block size of each round is dynamically and randomly determined, and not kept the same in each round, which is different from other traditional block cipher. Also, we adopt the CBC mode and use the logistic map to randomly generate the initial block and the initial pixel value of each encryption round. The random sequence generated by the logistic map is used as keystream for bitwise XOR operations and the random sequences generated by Henon map are used as keystream for rotating shift operations. The initialization and keystreams generation process is showed below in Section 3.1.

Without loss of generality, we assume the plain-image is a 256 gray-scale image of size $M \times N$, which can be seen as a one-dimensional vector $P = (p_1, p_2, \ldots, p_{M\times N})$, where p_i denotes the gray level of the image pixel in the row $floor\ (i/N)$ column $\mathrm{mod}(i, N)$. The initial value of the logistic map x_0 and the initial values of Henon map y_0 and z_0 are the secret keys of the algorithm. Suppose the encryption round of our dynamic block algorithm is n, the encryption process is showed below.

3.1. Initialization and keystreams generation.

(1) We denote st as the system time of the computer system. Calculate the st related initial value $x_0^{'}$ and $\left(y_0^{'}, z_0^{'}\right)$ of the two chaotic systems by $x_0^{'} = (st - floor(st)) \times x_0$, $y_0^{'} = (st - floor(st)) \times y_0$ and $z_0^{'} = (st - floor(st)) \times z_0$.

(2) Iterate Equation (1) by $x_0^{'}$ and Equation (2) by $y_0^{'}$ and $z_0^{'}$ for T times to get rid of transient effect, where T is a constant.

(3) Continue to iterate Equation (1) by the current state for n times and get n random numbers $s_1, s_2, \ldots, s_n$. Calculate the block size of the i-th round of the encryption bs_i by $bs_i = 2^{\,\mathrm{mod}\,(floor(s_i\times 2^{44}),8)+8}$.

(4) Continue to iterate Equation (1) by the current state for $\sum_{i=1}^{n} M \times N/bs_i$ times and get random numbers $\left\{c_{0,1}^i, c_{0,2}^i, \ldots, c_{0,M\times N/bs_i}^i\right\}$, where $i = 1, 2, \ldots, n$.

(5) The key length of the i-th round of the block encryption algorithm is bs_i, and thus the summed length of the keystreams is $L = \sum_{i=1}^{n} bs_i$. Iterate the logistic map by the current state for L times and denote the sequence as $X = \{x_1, x_2, \ldots, x_L\}$.

(6) Obtain an 8-bit random code sequence $DX = \{dx_1, dx_2, \ldots, dx_L\}$ according to the following formula,

$$dx_k = \mathrm{mod}\left(floor\left(x_k \times 2^{46}\right), 256\right), \quad k = 1, 2, \ldots, L. \tag{3}$$

Denote $DX = \{DX_i\}$, $i = 1, 2, \ldots, n$ and $DX_i = \{dx_1^i, dx_2^i, \ldots, dx_{bs_i}^i\}$ is the bitwise XOR keystream of the i-th round.

(7) Iterate Equation (2) by the current state for L/2 times to produce two chaotic sequences denoted as $Y = \{y_1, y_2, \ldots, y_{L/2}\}$ and $Z = \{z_1, z_2, \ldots, z_{L/2}\}$.

(8) Obtain an 8-bit random code sequences $DY = \{dy_1, dy_2, \ldots, dy_{L/2}\}$ and $DZ = \{dz_1, dz_2, \ldots, dz_{L/2}\}$ according to the following formulas,

$$dy_k = \mathrm{mod}\left(floor\left({y_k}^i \times 2^{48}\right), 256\right), \quad k = 1, 2, \ldots, bs_i/2, \tag{4}$$

$$dz_k = \mathrm{mod}\left(floor\left({z_k}^i \times 2^{50}\right), 256\right), \quad k = 1, 2, \ldots, bs_i/2. \tag{5}$$

Denote $DY = \{DY_i\}$, $DZ = \{DZ_i\}$, $i = 1, 2, \ldots, n$, then $DY_i = \left\{dy_1^i, dy_2^i, \ldots, dy_{bs_i/2}^i\right\}$ and $DZ_i = \left\{dz_1^i, dz_2^i, \ldots, dz_{bs_i/2}^i\right\}$ are the rotating shift keystreams of the i-th round.

(9) Continue to iterate Equation (1) by the current state for L times and get sequence $V = \{v_1, v_2, \ldots, v_L\}$.

(10) Obtain an 8-bit random code sequence $IV = \{iv_1, iv_2, \ldots, iv_L\}$ according to the following formulas,

$$iv_k = \mathrm{mod}\left(floor\left(v_k \times 2^{52}\right), 256\right), \quad k = 1, 2, \ldots, L. \tag{6}$$

Denote $IV = \{IV^1, IV^2, \ldots, IV^n\}$, where the size of the i-th block IV^i is bs_i and IV are random initial blocks of the CBC mode.

3.2. **The CBC mode.** The CBC mode is the most widely used block cipher mode which is shown in Figure 1, where P is the plaintext, C is the ciphertext, E_k is the block cipher encryption using key k, IV is the Initialization Vector and $\oplus$ is bitwise XOR operator. The encryption of CBC mode can be described by Equation (7) and the decryption of CBC mode can be described by Equation (8),

$$C_j = E_k\left(P_j \oplus C_{j-1}\right), \quad C_0 = IV, \tag{7}$$

$$P_j = D_k(C_j) \oplus C_{j-1}, \quad C_0 = IV, \tag{8}$$

where P_j is the j-th plaintext block and C_j is the j-th ciphertext block. We adopt the CBC mode in this paper and IV of each round of encryption is randomly generated by the logistic map as in Step 10 of Section 3.1.

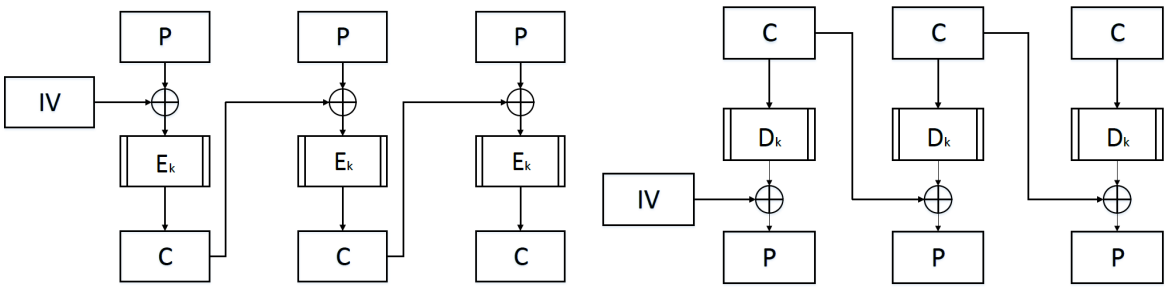


FIGURE 1. The CBC mode of block cipher: (a) encryption, (b) decryption

3.3. **The encryption scheme.**

(1) Let $i = 1$ and $B = P$.

(2) Without loss of generality, we assume that $M \times N$ can be divided by bs_i. Divide B into $M \times N/bs_i$ groups and there are bs_i values in each block, where we denote the block as $B^i = \left\{b_1^i, b_2^i, \ldots, b_{bs_i}^i\right\}$.

(3) For the i-th round of the encryption, compute the pixel value of the cipher-image block C^i from a block B^i by the following formula,

$$c_k^i = \begin{cases} \left(b_k^i <<< dz_{(k+1)/2}^i\right) \oplus c_{k-1}^i \oplus dx_k^i, & \text{if } k \bmod 2 = 1, \\ \left(\left(b_k^i <<< dy_{k/2}^i + dx_k^i\right) \bmod 256\right) \oplus c_{k-2}^i, & \text{if } k \bmod 2 = 0, \end{cases} \tag{9}$$

where $k = 1, 2, \ldots, bs_i$, $\oplus$ is bitwise XOR operator, $<<<$ is the rotating left shift operation, c_k^i is the gray level of the cipher-image pixel and c_0^i is a random value calculated in Step 4 of Section 3.1 and $c_0^i = c_{0,j}^i$ for the j-th ($j = 1, 2, \ldots, M \times N/bs_i$) block.

(4) Use the CBC mode as described in Section 3.2 and adopt Equation (9) to all the blocks. Denote the encrypted image as B.

(5) Let $i = i + 1$ and return to Step 2 until i reaches n. We denote the final cipher-image as $C = \{c_1, c_2, \ldots, c_{M \times N}\}$.

3.4. **The decryption scheme.** The decryption process is similar to the encryption process and is the inverse of the encryption process.

(1) Given st and initialize and generate keystreams as described in Section 3.1.
(2) Let $i = n$.
(3) Divide C into $M \times N/bs_i$ groups and there are bs_i values in each block, where we denote the cipher-image block as $C^i = \left\{c_1^i, c_2^i, \ldots, c_{bs_i}^i\right\}$.
(4) For the i-th round of the decryption, compute the pixel value of the plain-image block B^i from a block C^i by the following formula,

$$b_k^i = \begin{cases} \left(b_k^i \oplus c_{k-1}^i \oplus dx_k^i\right) >>> dz_{(k+1)/2}^i, & \text{if } k \bmod 2 = 1, \\ \left(b_k^i \oplus c_{k-2}^i - dx_k^i\right) \bmod 256 >>> dy_{k/2}^i, & \text{if } k \bmod 2 = 0, \end{cases} \tag{10}$$

where $k = 1, 2, \ldots, bs_i$, $\oplus$ is bitwise XOR operator, $>>>$ is the rotating right shift operation, c_k^i is the gray level of the cipher-image pixel and c_0^i is a random value calculated in Step 4 of Section 3.1 and $c_0^i = c_{0,j}^i$ for the j-th ($j = 1, 2, \ldots, M \times N/bs_i$) block.
(5) Use the CBC mode as in Section 3.2 and adopt Equation (10) to all the blocks. Denote the decrypted image as C.
(6) Let $i = i - 1$ and return to Step 3 until i reaches 1. Then we can get the final decrypted plain-image P.

4. Experimental Results and Security Analysis.

4.1. **Key space analysis.** Key space size is the total number of different keys which can be used in the encryption. A secure encryption algorithm should have large enough key space to make brute-force attack impossible. In our algorithm, the initial values of the two chaotic maps x_0, y_0 and z_0 are the secret keys, where $x_0 \in (0, 1)$. According to the IEEE floating-point standard, the computational precision of the 64-bit double-precision numbers is 2^{-52}. Therefore, the key space is at least $2^{52\times3} = 2^{156}$. This is a large key space that can prevent against brute-force attacks.

4.2. **Statistical analysis.**

4.2.1. *Histogram analysis.* Image histogram is a very important feature in image analysis. We use $(x_0, y_0, z_0) = (0.338688, 0.556444, 0.12098688)$ as a key to encrypt plain-image "Lenna". Figures 2(a)-2(d) depict the plain-image, cipher-image and the histograms of them, respectively. From these figures, we can see that the histogram of the cipher-images is fairly uniform and is significantly different from that of the plain-image.

4.2.2. *Correlation analysis.* Correlation between pixels is an important intrinsic feature of an image. To resist against statistical analysis, a good encryption algorithm should remove the high correlation of the plain-image. We select all the pairs of adjacent pixels in vertical, horizontal, and diagonal direction from plain-image and cipher-image and calculate the coefficient by Equations (11)-(14),

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

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} [x_i - E(x)]^2, \tag{12}$$

$$\operatorname{cov}(x, y) = \frac{1}{N}\sum_{i=1}^{N} [x_i - E(x)][y_i - E(y)], \tag{13}$$

$$r_{xy} = \frac{\operatorname{cov}(x, y)}{\sqrt{D(x)D(y)}}, \tag{14}$$

where x and y are grey-scale values of two-adjacent pixel in the image and N is total number of pixels.

We test the correlation coefficients of adjacent pixels in the plain-images "Lenna", "Baboon" and "Pepper" and their cipher-images and the results are listed in Table 1. The results of the proposed algorithm and three existing algorithms [19, 24, 25] are shown in Table 2. Results show that the correlation coefficients of adjacent pixels in the cipher-image are around zero and negligible. Moreover, we choose 1000 pairs of horizontally adjacent pixels and the correlation of them is shown in Figure 3. These results show that the proposed algorithm has removed the strong correlation among neighboring pixels of the plain-image.

4.2.3. *Information entropy analysis.* The information entropy is an important feature of the randomness and the information entropy $H(s)$ of a message source s with 2^N symbols

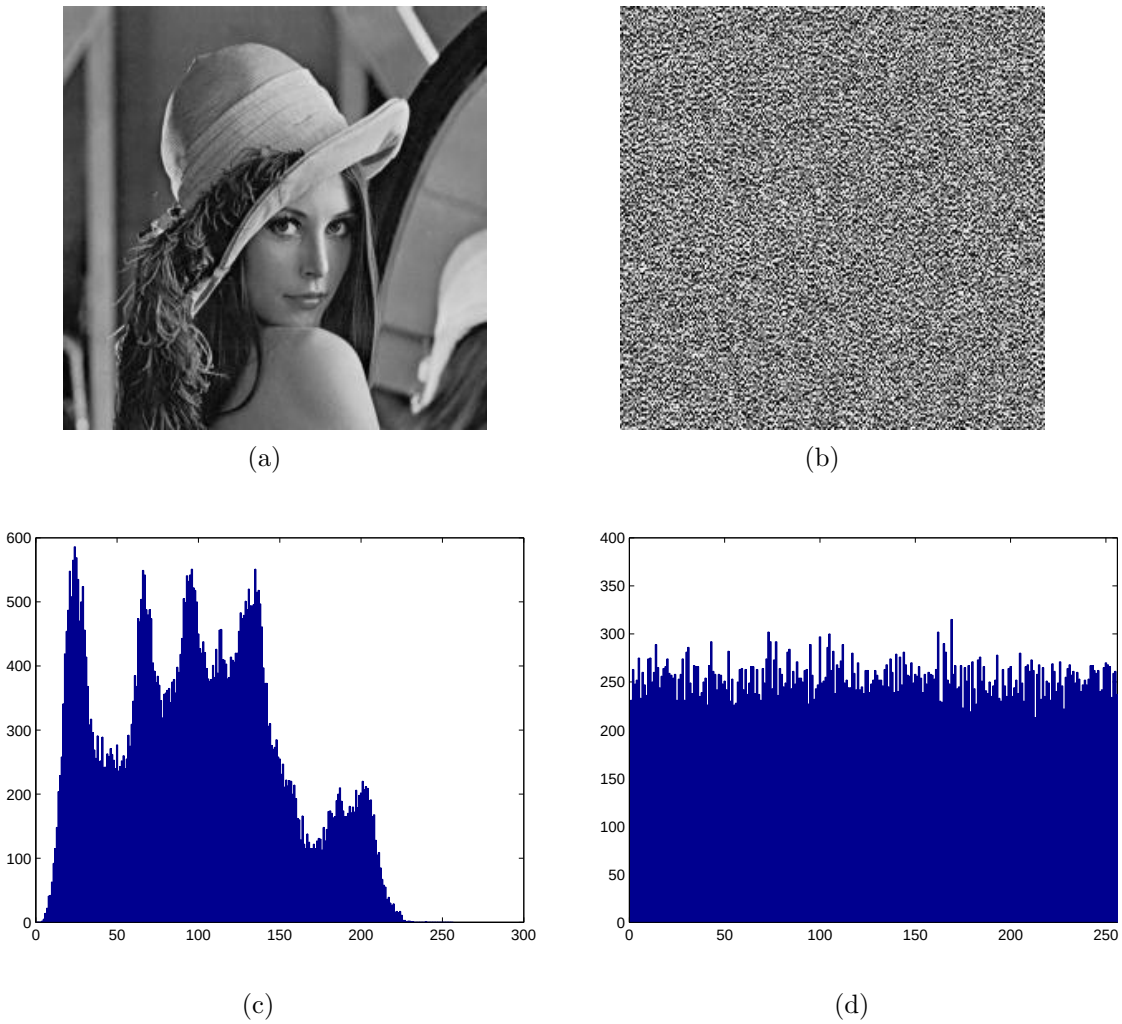


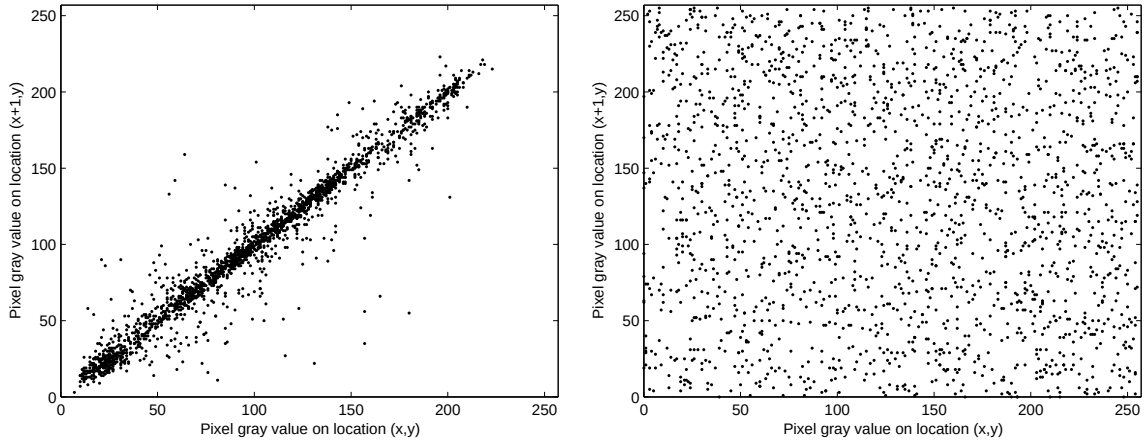
FIGURE 2. (a) Plain-image, (b) cipher-image, (c) histogram of the plain-image, and (d) histogram of the cipher-image

TABLE 1. Correlation coefficients of two adjacent pixels in the plain-image and cipher-image

Direction	Plain-image Lenna	Cipher-image Lenna	Plain-image Baboon	Cipher-image Baboon	Plain-image Pepper	Cipher-image Pepper
Horizontal	0.939918	−0.000742	0.864410	0.002477	0.976663	−0.004669
Vertical	0.969235	−0.002019	0.758829	0.001085	0.979115	−0.000272
Diagonal	0.937176	−0.000830	0.726145	−0.002096	0.963847	−0.002371

TABLE 2. Correlation coefficients of two adjacent pixels in the plain-image and cipher-image of Lenna

Direction	Plain-image	Our algorithm	Ref. [19]	Ref. [24]	Ref. [25]
Horizontal	0.939918	−0.000742	0.0058	−0.068521	0.00190641811860
Vertical	0.969235	−0.002019	0.0094	0.008744	0.00381759867381
Diagonal	0.937176	−0.000830	0.0214	−0.073928	−0.00194828025125



(a) (b)

FIGURE 3. Correlations of two horizontally adjacent pixels: (a) correlation of the plain-image, (b) correlation of the cipher-image

TABLE 3. Entropy values for the cipher-images

Plain-image	Entropy value
Lenna	7.99632
Baboon	7.99902
Pepper	7.99927

can be calculated as

$$H(s) = \sum_{i=0}^{2^N-1} p(s_i)\log_2\frac{1}{p(s_i)}, \tag{15}$$

where $p(s_i)$ denotes the probability of symbol s_i. For a true random 256 gray-scale image, the entropy should be 8. The entropy values of cipher-images of three plain-images "Lenna", "Baboon" and "Pepper" are shown in Table 3. The entropy values of the cipher-images are very close to the theoretical value 8 which means the information leakage in the encryption process is negligible.

4.3. **Sensitivity analysis.** *NPCR* (number of pixels change rage) and *UACI* (unified average changing intensity) are two most common quantities used to evaluate the strength of image encryption algorithms against differential attacks, which can measure the different range between two images. It is described in Equation (16) and Equation (17),

$$NPCR = \frac{\sum_{i,j} D(i,j)}{W \times H} \times 100\%, \tag{16}$$

$$UACI = \frac{1}{W \times H}\left[\sum_{i,j}\frac{|c_1(i,j) - c_2(i,j)|}{255}\right] \times 100\%, \tag{17}$$

where c_1 and c_2 are two images with the same size $W \times H$. If $c_1(i,j) = c_2(i,j)$, $D(i,j) = 1$; otherwise, $D(i,j) = 0$.

4.3.1. *Key sensitivity analysis.* Key sensitivity is an essential feature for any good cryptosystem which guarantees the security of the cryptosystem against the brute-force attack to some extent. In the proposed algorithm the key K is composed of three parts (x_0, y_0, z_0), so we run several tests for each part of the key K separately. We randomly change one part of (x_0, y_0, z_0) and keep the other two unchanged to get keys with slight differences. We encrypt image "Lenna" using key $K_0 = (x_0, y_0, z_0) = (0.123456789, 0.234567891, 0.345678912)$ and these slightly changed keys and calculate the *NPCR* and *UACI* of the cipher-images using Equation (16) and Equation (17). The results are shown in Table 4.

TABLE 4. *NPCR* and *UACI* between cipher-image with key K_0 and other cipher-images with slightly different keys

Key	*NPCR* (%)	*UACI* (%)
$(0.1234567891, 0.2345678910, 0.3456789120)$	99.6307	33.4638
$(0.1234567890, 0.2345678911, 0.3456789120)$	99.6384	33.4257
$(0.1234567890, 0.2345678910, 0.3456789121)$	99.6078	33.4961

4.3.2. *Plaintext sensitivity.* In order to resist differential attack, a tiny change in the plain-image should cause a substantial change in the cipher-image. Given a 256 gray-scale plain-image P of size 256×256 and get a P' which only has a single pixel difference in a random position (i,j), where $i, j = 0, 1, l, \ldots, 255$, by the following formula,

$$P'(i,j) = \begin{cases} P(i,j) + 1, & \text{if } p(i,j) < 255, \\ 0, & \text{otherwise.} \end{cases} \tag{18}$$

We encrypt P and P' and get the cipher-images C and C', then we calculate the *NPCR* and *UACI* values of the cipher-images. We test 3 groups of plain-images for 100 times of each group and get the average *NPCR* and *UACI* shown in Table 5. It can be seen that the results are very close to the expectation.

TABLE 5. *NPCR* and *UACI* between cipher-images with slightly different plain-images

Plain-image	*NPCR* (%)	*UACI* (%)
Lenna	99.5858	33.4104
Baboon	99.5673	33.4263
Pepper	99.5878	33.4418

4.4. **Speed analysis.** We have used Matlab R2013b to run the encryption and decryption programs in a personal computer with a Intel Core i3 CPU 2.53 GHz, 2 GB memory and 250 GB hard-disk capacity, and the operation system is Microsoft Windows 7. The average time of encryption/decryption on 256 gray-scale images of size 256×256 is shorter than 0.3 s which can be used in practical situations.

5. **Conclusions.** This paper presents a new dynamic block chaotic image encryption method. A new dynamic block cipher mode is proposed that the block size of each round of the encryption is dynamically and randomly determined by a logistic map. In this way, different rounds of encryption may have different block sizes which improves the complexity of the scheme. Two chaotic maps, the logistic map and the Henon map are used to generate keystreams. Also, the encryption operation differs depending on that the

image pixel number is odd or even. System time is used as a random initial value with the key together to obtain the new and actually used initial value of the two chaotic maps. Thus, the encryption differs every time and the method can resist against known-plaintext and chosen-plaintext attacks. We have also carried out key space analysis, statistical analysis, key sensitivity analysis and plaintext sensitivity analysis to demonstrate the security of the new image encryption algorithm. It is showed that the new dynamic block cipher mode has efficiency and security in image encryption. Future work will be undertaken for other data encryption such as text encryption, and video encryption.

REFERENCES

[1] A. Cheddad, J. Condell, K. Curran and P. McKevitt, A hash-based image encryption algorithm, *Optics Communications*, vol.283, no.6, pp.879-893, 2010.

[2] X. Wang and D. 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, 2013.

[3] P. Ping, F. Xu and Z.-J. Wang, Image encryption based on non-affine and balanced cellular automata, *Signal Processing*, vol.105, pp.419-429, 2014.

[4] X. Wei, L. Guo, Q. Zhang, J. Zhang and S. Lian, A novel color image encryption algorithm based on DNA sequence operation and hyper-chaotic system, *Journal of Systems and Software*, vol.85, no.2, pp.290-299, 2012.

[5] L. Liu, Q. Zhang and X. Wei, A RGB image encryption algorithm based on DNA encoding and chaos map, *Computers & Electrical Engineering*, vol.38, no.5, pp.1240-1248, 2012.

[6] X. Wang and L. Teng, An image blocks encryption algorithm based on spatiotemporal chaos, *Nonlinear Dynamics*, vol.67, no.1, pp.365-371, 2012.

[7] L. Sui, K. Duan, J. Liang, Z. Zhang and H. Meng, Asymmetric multiple-image encryption based on coupled logistic maps in fractional fourier transform domain, *Optics and Lasers in Engineering*, vol.62, pp.139-152, 2014.

[8] R. Matthews, On the derivation of a "chaotic" encryption algorithm, *Cryptologia*, vol.13, no.1, pp.29-42, 1989.

[9] N. K. Pareek, V. Patidar and K. K. Sud, Image encryption using chaotic logistic map, *Image and Vision Computing*, vol.24, no.9, pp.926-934, 2006.

[10] S.-J. Deng, G.-C. Huang, Z.-J. Chen and X. Xiao, Self-adaptive image encryption algorithm based on chaotic map, *Journal of Computer Applications*, vol.31, no.6, pp.1502-1504, 2011.

[11] M. Amin, O. S. Faragallah and A. A. A. El-Latif, A chaotic block cipher algorithm for image cryptosystems, *Communications in Nonlinear Science and Numerical Simulation*, vol.15, no.11, pp.3484-3497, 2010.

[12] G. Zhang and Q. Liu, A novel image encryption method based on total shuffling scheme, *Optics Communications*, vol.284, no.12, pp.2775-2780, 2011.

[13] X. Huang, Image encryption algorithm using chaotic chebyshev generator, *Nonlinear Dynamics*, vol.67, no.4, pp.2411-2417, 2012.

[14] D. Xiao, X. Liao and P. Wei, Analysis and improvement of a chaos-based image encryption algorithm, *Chaos, Solitons & Fractals*, vol.40, no.5, pp.2191-2199, 2009.

[15] C. Li, S. Li, G. Chen and W. A. Halang, Cryptanalysis of an image encryption scheme based on a compound chaotic sequence, *Image and Vision Computing*, vol.27, no.8, pp.1035-1039, 2009.

[16] T. Gao and Z. Chen, A new image encryption algorithm based on hyper-chaos, *Physics Letters A*, vol.372, no.4, pp.394-400, 2008.

[17] C. Gangadhar and K. D. Rao, Hyperchaos based image encryption, *International Journal of Bifurcation and Chaos*, vol.19, no.11, pp.3833-3839, 2009.

[18] C. Zhu, A novel image encryption scheme based on improved hyperchaotic sequences, *Optics Communications*, vol.285, no.1, pp.29-37, 2012.

[19] H. Zhu, C. Zhao and X. Zhang, A novel image encryption-compression scheme using hyper-chaos and Chinese remainder theorem, *Signal Processing: Image Communication*, vol.28, no.6, pp.670-680, 2013.

[20] C. Li, S. Li, M. Asim, J. Nunez, G. Alvarez and G. Chen, On the security defects of an image encryption scheme, *Image and Vision Computing*, vol.27, no.9, pp.1371-1381, 2009.

[21] C. Li, Y. Liu, L. Y. Zhang and M. Z. Chen, Breaking a chaotic image encryption algorithm based on modulo addition and XOR operation, *International Journal of Bifurcation and Chaos*, vol.23, no.4, 2013.

[22] X. Wang, D. Luan and X. Bao, Cryptanalysis of an image encryption algorithm using Chebyshev generator, *Digital Signal Processing*, vol.25, pp.244-247, 2014.
[23] G. Ye, A block image encryption algorithm based on wave transmission and chaotic systems, *Nonlinear Dynamics*, vol.75, no.3, pp.417-427, 2014.
[24] G. Ye and J. Zhou, A block chaotic image encryption scheme based on self-adaptive modelling, *Applied Soft Computing*, vol.22, pp.351-357, 2014.
[25] X. Wang, L. Liu and Y. Zhang, A novel chaotic block image encryption algorithm based on dynamic random growth technique, *Optics and Lasers in Engineering*, vol.66, pp.10-18, 2015.
[26] X. Zhang, G. Zhang, X. Li, Y. Ren and J. Wu, Image encryption using random sequence generated from generalized information domain, *Chinese Physics B*, vol.25, no.5, 2016.