Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 216048, 7 pages
http://dx.doi.org/10.1155/2014/216048

Research Article

An Image Encryption Algorithm Based on Balanced Pixel and Chaotic Map

Jian Zhang and Yutong Zhang

College of Information and Computer Engineering, Northeast Forestry University, Harbin 150040, China

Correspondence should be addressed to Jian Zhang; jian_zhang2014@163.com

Received 20 August 2014; Accepted 28 September 2014; Published 10 November 2014

Academic Editor: Guoqiang Zhang

Image encryption technology has been applied in many fields and is becoming the main way of protecting the image information security. There are also many ways of image encryption. However, the existing encryption algorithms, in order to obtain a better effect of encryption, always need encrypting several times. There is not an effective method to decide the number of encryption times, generally determined by the human eyes. The paper proposes an image encryption algorithm based on chaos and simultaneously proposes a balanced pixel algorithm to determine the times of image encryption. Many simulation experiments have been done including encryption effect and security analysis. Experimental results show that the proposed method is feasible and effective.

1. Introduction

With the development of technology, encryption and dissemination of digital images developed rapidly. They have been used in many fields such as military, medical, industrial, and multimedia fields [1]. In the past years, many encryption modes have been proposed including scan mode method [2], double random phase encoding [3], encoding and iterative random rotation converts vector encoding [4, 5], the quad tree coding [6], and coding and chaotic function Kolmogorov flow [7]. Due to the inner randomness and initial sensitivity, chaotic encryption has been used widely and developed fast [8]. Chaos encryption method is mainly to change the position and value of pixels in the image [9]. Among the existing methods, many methods need to be encrypted several times because of the unsatisfied effect [10–14], but there is no effective way to make sure of the exact number of encryption times. Considering this, the paper proposes an encryption algorithm based on chaos and iteration equation and gives the method to determine the number of encryption times. In the algorithm, we present an encryption method based on one-dimensional logistic map and change the position and value of pixels sufficiently. By balancing the pixels, the number of encryption times can be decided and the effectiveness of the algorithm through lots of experiments can be verified.

2. The Chaos Theory and Iterative Equation

2.1. The Chaos Theory. Chaotic system has the characteristic of inner randomness and sensitivity to initial value. The inner randomness is that random uncertainty from the chaotic system happens naturally, not because of the external environmental effect [15]. What is more, the sensitivity to initial value means two initial values which are close enough experience a long time along the trajectory and will get two totally different results.

Because this algorithm involves the choice of the initial value of seed g (the following part will express seed g), we need it to have two characteristics which are randomness and high sensitivity. So it is necessary to introduce the one-dimensional chaos equation logistic function which has these two characteristics.

The one-dimensional chaos equation logistic function is given by

$$X_{n+1} = \mu X_n (1 - X_n), \tag{1}$$

where $\mu \in [0, 4]$ and $X \in [0, 1]$. Research shows that when $X \in [0, 1]$ the function stays in a chaotic condition and shows a state of random uncertainty [16, 17]. The μ is closer to 4, X is closer to even distribution in $[0, 1]$. Therefore, the closer the value of μ is to 4, the better is the effect [18]. According to the experimental analysis, when $\mu = 3.9999$, for two quite close different initial values $X_0 = 0.675279000$ and $X_0 = 0.675279001$, the corresponding logistic sequences are almost the same in 30 times of iteration but are totally different after 30 times. So in this algorithm, select $\mu = 3.9999$, and each initial value X_0 will be iterated 35 times and then be used and be computed.

2.2. Iterative Equation. In general, adjacent pixels in the image have a strong correlation [19]. Therefore, in order to improve the quality of the encrypted image encryption, iterative equation should make the position of each pixel change as far as possible [20], and, in this paper, the encryption process will directly work on each pixel in the original image. The original image can be seen as a two-dimensional matrix of $N * M$ (N rows and M columns).

The iterative equation is

$$Y_{n+1} = \left[\left[{Y_n}^2 \bmod S\right] * Y_n + Y_g\right] \bmod S,$$
$$Y_0 = g, \qquad (2)$$
$$Y_g = g^2,$$

where $S = L - 1$ L is the size of the image I_0 and $L = N * M$ S decreases with each iteration. The seed g is less than L. Each loop of the whole algorithm needs a new value. The S is a initial value from $L - 1$ to 1. The position of pixel can be calculated by memory effect (e.g., the index i read out the size L of the input image and the new position of Y_n after each displacement and i). Therefore, the value of the position of each pixel can be determined after several changes. The significance of this iterative equation is that disrupting the initial position of all of the pixel from 1 to L in the original image.

3. A Balanced Pixel Algorithm for Image Encryption

This proposed image encryption algorithm combines the chaos theory and iterative equation, converting the position and the value of each pixel of the original image to finish one loop encryption. In order to obtain a better effect of encryption, the algorithm proposes a method in which the number of encryption times can be defined. The flowchart of the algorithm is shown in Figure 1.

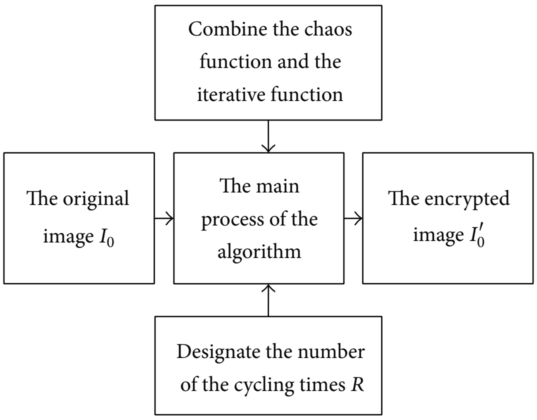

Combine the chaos function and the iterative function
The original image I_0
The main process of the algorithm
The encrypted image I_0'
Designate the number of the cycling times R

Figure 1: The flowchart of the algorithm.

3.1. The Theory and Steps of the Encryption. The process of the encryption is completed by the following six steps.

(1) The original image is with I_0, a two-dimensional matrix of N rows and M columns; transform I_0 into a one-dimensional sequence $I_0^b[i]$ as shown in

$$I_0 \longrightarrow I_0^b[i]. \qquad (3)$$

(2) Given an initial value of X_0, $\mu = 3.9999$, according to the chaotic equation (1) get a random sequence about X_n. As mentioned, logistic function will show better avalanche state after 30 times of iteration, so in this algorithm we use the data after 35 iterations, that is, to use the data from the beginning of X_{35}. Since the sequence X_n is a set of decimals between 0 and 1, while algorithm needs integer, so we expend each of the decimals we get 10^7 times and then get the last two digits as the value of g. Moreover, we also have to think about the value which is from the 35th of the sequence; thus, g is shown as

$$g_n = \left(10^7 X_{n+35}\right) \bmod 100, \qquad (4)$$

where n represents the number of cycling times of the whole algorithm.

(3) According to (2) and (4), a sequence about Y_n is given.

(4) Determine Z_1, which is a sequence from an one-dimensional I_0^b, shown as (5):

$$Z_1 = I_0^b[i]. \qquad (5)$$

(5) Determine Z_2, which is an algebraic equation got from Y_n about i, shown as

$$Z_2 = I_0^b[j] = I_0^b[Y_{i+1} + i - 1]. \qquad (6)$$

(6) Determine Z_3, which is a sequence got from the *XOR* operation between Z_1 and Z_2, as

$$Z_3 = Z_1 \oplus Z_2 \qquad (7)$$

Z_1, Z_2, and Z_3 are a loop. Each loop will get Z_3 and this Z_3 will be the new Z_1 of the next loop; then, according to (4) a new g to continue the next time of cycling is given.

3.2. Determination of the Number of Encrypting Times. In order to get a good effect, we need to repeat the six steps above several times (the six steps are the time of cycle of the

Table 1: The relationship between the image size and R_1.

Size	100 ∗ 100	100 ∗ 150	256 ∗ 256	240 ∗ 360	512 ∗ 512	500 ∗ 750
R_1	20	19	17	16	15	14

encryption and also the main process of the algorithm). Each time of the cycling will get different effect of the encrypted image. To achieve the most effective encrypted image, an analysis of the number of cycling times R is needed; then, the optimal value of the images with different size is given.

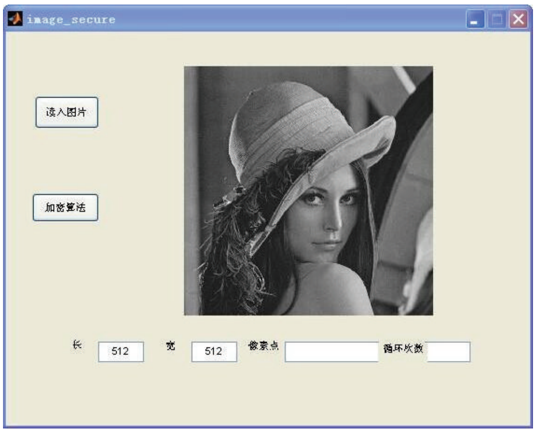
3.2.1. Determination of R_1. R is related to R_1 and R_2, R_1 is determined by key space, R_2 can be given by experimental analysis, and what follows in the passage will be the determination of R in detail.

According to today's computing speed, statistical analysis shows that key space of the encryption system that is larger than 2^{256} can resist brute force attacks. In this algorithm, each time of cycling will give an initial g (e.g., $R = 1$) and generate L different codes. Therefore, as the growth of the cycling number R, the total number of keys can be shown as L^R. To satisfy the relationship $L^R \geq 2^{256}$ and avoid brute force, the minimum of R is defined as

$$R_1 = R_{\min} = \text{Floor}\left[\frac{256}{\log_2^L}\right] + 1. \tag{8}$$

As can be seen from the equation above, $R_{\min}$ is a decreasing sequence, with the size of the image increasing, and the number of cycle times we need is decreasing. By (8) it can be calculated that R_1 is decreasing with the increasing size of the image. Table 1 shows the corresponding R_1 for different image size.

R_1 is a limitation on key space, and to finalize R, a more important indicator is needed, that is, R_2 balancing the image pixel value.



长 512 宽 512 像素点 循环次数

Figure 2: Original image.

3.2.2. Determination of R_2. Balance pixels, as the name suggests, are the average of pixels in an image. Grayscale images have 256 gray levels from 0 to 255. Each one is made up of several pixels of any level from 0 to 255. These 256 gray levels which are not necessary all appear in one image, and there is not a fixed value of them, so they are dispensable and can be more or less. A good encryption algorithm is to make the number of each encrypted image pixel be distributed evenly. Assuming an $N * M$ image I_0 needs to be encrypted, then the image has a total of $L = N * M$ pixels. First calculating that the pixel value of 0 has h_0 and pixel value of 1 has h_1, so the pixel value of 255 has h_{255}. Here h_n stands for the number of pixel values of n, $n \in [0, 255]$, and n is an integer. Evaluate a good encryption algorithm; a standard is to balance pixel as far as possible. Therefore, the experimental will give statistics about the balance of pixel after each time of cycle. Here the balance means that 256 gray levels are uniformly distributed in the whole image. Ideally there are $L/256$ pixels of each gray level in the image; the ideal value is recorded as H_0 as shown in

$$H_0 = \frac{L}{256}. \tag{9}$$

Do the difference operation between h_n and H_0 and get the absolute value $|h_n - L/256|$ which is the deviation between actual value and ideal value of each gray level pixel. It is recalled $\widetilde{H}$ as shown in

$$\widetilde{H} = |h_n - H_0|. \tag{10}$$

Do quotient operation between $\widetilde{H}$ and H_0 and get K_n which is the rate of deviation between the actual and ideal situations of each pixel as shown in

$$K_n = \frac{\widetilde{H}}{H_0}. \tag{11}$$

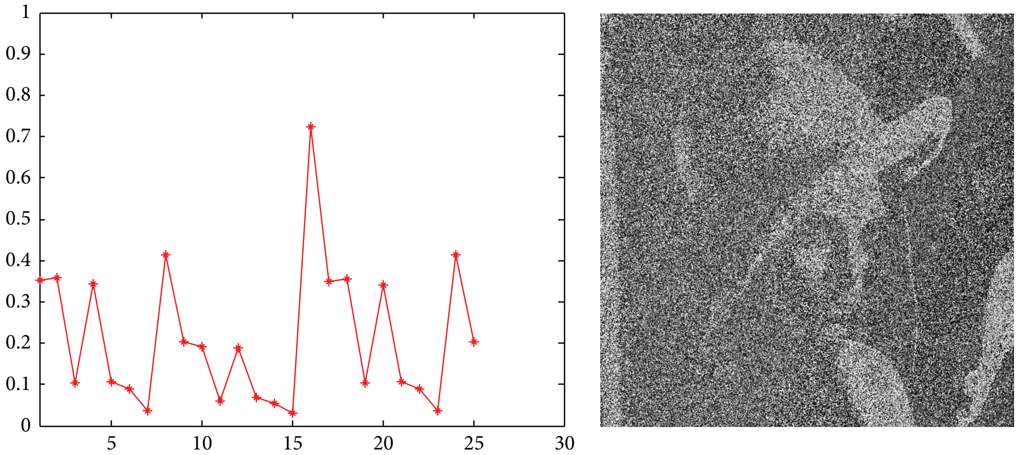
Thus the total rate of deviation is the sum of these 256 K_n recorded as K and is as shown in

$$K = \sum_{n=0}^{255} K_n. \tag{12}$$

Then get the average rate of deviation $\overline{K}$ which is shown in

$$\overline{K} = \frac{K}{256}. \tag{13}$$

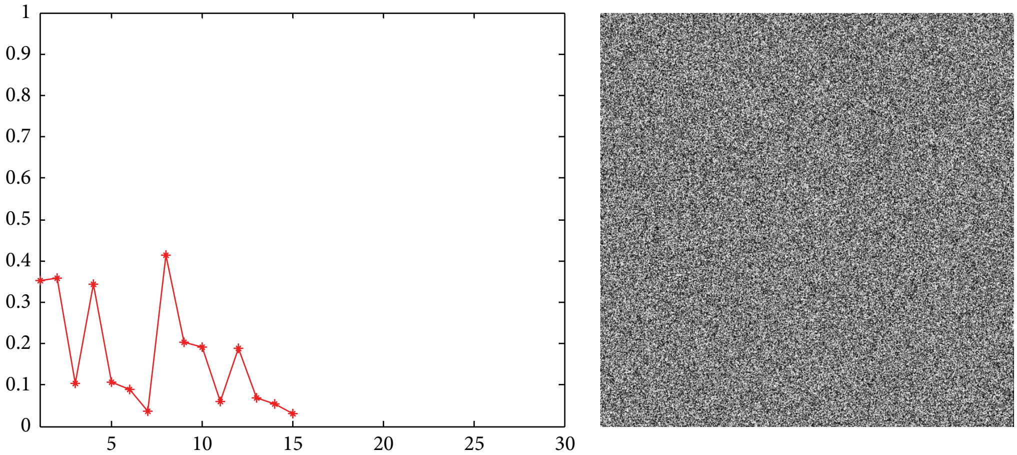
What we need to do is getting a $\overline{K}$ after each time of cycle and make it as low as possible, and when it achieves the lowest point or a value that we can accept, stop cycling and get R_2. Through comparing to R_1, R_2 should be selected larger than R_1, and smaller than $\overline{K}$. This R_2 is also the number of cycle times of the whole encryption algorithm.



(a) The line chart

(b) The encrypted image

Figure 3: The line chart and the encrypted image.



(a) The line chart

(b) The encrypted image

Figure 4: The line chart and the encrypted image.

4. Experimental Analysis

4.1. Analysis on Encryption Effectiveness. This section will make specific experiment on algorithm and give the effectiveness analysis. Take an image of the size 512 ∗ 512 as an example and give the experimental process. The original image is shown as in Figure 2. When the file is a color image, the system will convert the image to grayscale automatically. The length and the width shown in the picture are just the size of the image, as this figure is 512 ∗ 512.

Start to encrypt; you can fill in the number of cycles needed. As can be seen from Table 1, the smallest possible value of R_1 is 15 for the 512 ∗ 512 size image. To facilitate the observation of the image trend, input 25 times directly. Then press the button encrypt; program begins to run and several seconds later, the line chart is as shown in Figure 3(a) and the encrypted image is as shown in Figure 3(b).

Figure 3 is the encrypted image of 25 times of cycle, which obviously have not achieved the desired effect, while it can be seen from the line chart that the lowest points of the plotline are the 7th, 15th, and 23rd times of cycle. To satisfy the condition not less than R_1, reset the number of cycle times to 15, given the line chart and the encrypted image as shown in Figure 4. This time experiment gets a good effective encrypted image.

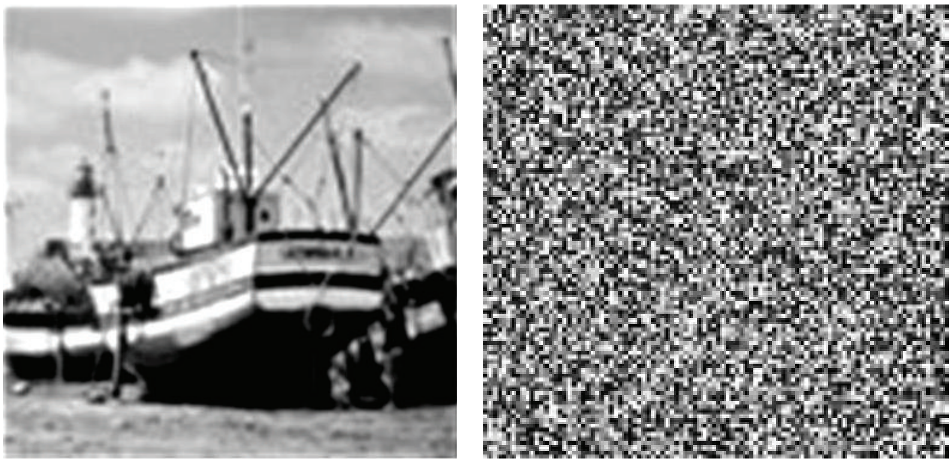
Several experiments statistical analysis can draw a conclusion that images of different sizes require different ideal numbers of cycle times shown in Table 2.

When given the size of an image, you can input the number of the cycle times directly according to Table 2. Also the cycle times can be calculated. First get the restriction of R_1 and then input a number which is larger than R_1 as the cycle time (usually the number is larger than R_1 10); observe the line chart that appeared and finally take the lowest point as the number of cycle times.

Go on taking an image of size 100 ∗ 100 as the original figure. Based on the conclusion above, you can determine directly that it needs 23 times of cycling. The original and

Table 2: Ideal number of cycling times.

Image size	100 ∗ 100	100 ∗ 150	256 ∗ 256	240 ∗ 360	512 ∗ 512	500 ∗ 750
Ideal number	23	23	23	23	15	15



(a) The original image

(b) The encrypted image

Figure 5: The original image and encrypted image.

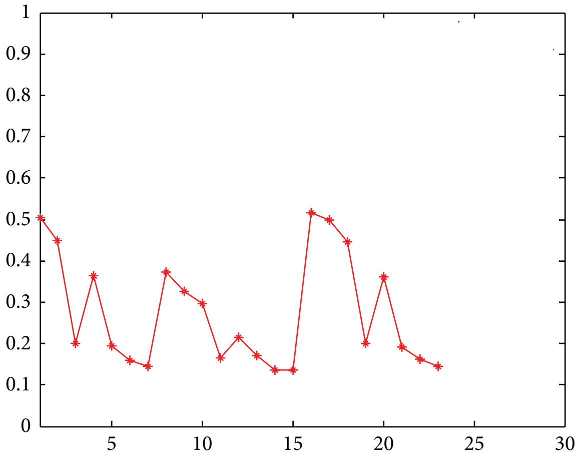


Figure 6: The line chart.

encrypted images are shown as in Figure 5 and the chart line is shown as in Figure 6. It displays that the lowest points are 7th, 15th, and 23rd points and the highest point is 16th point. Here the encrypted image and line chart of 16 times of cycling are shown as in Figure 7. It can be seen that the result is not that ideal but uneven.

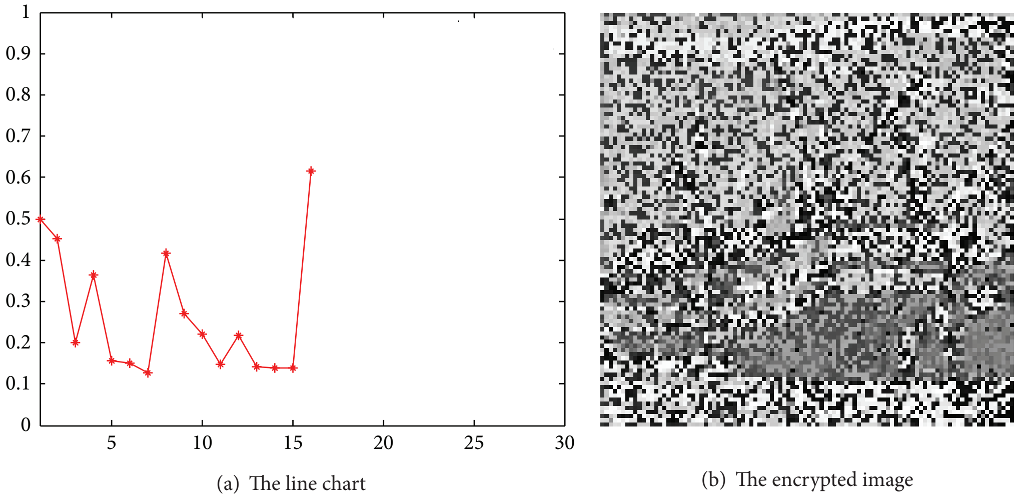
4.2. Analysis on Key Sensitivity. The experiments above can prove the effectiveness and practicality of the algorithm; a good algorithm should also have strong security to resist brute attack. This section will give the analysis on key sensitivity. The same 512 ∗ 512 picture of Lena is given as the original image. The initial value $X_0 = 0.642235000$. The original image and encrypted image are as shown in Figures 8(a) and 8(b). When X_0 is changed to $X_0' = 0.642235001$, the encrypted image cannot be decrypted successfully, which is shown in Figure 8(c). It shows that the algorithm has strong sensitivity to initial value.

Table 3: The comparison of the correlation coefficients of Lena's images.

Correlation	Horizontal direction	Vertical direction	Diagonal direction
Original image	0.9768	0.9145	0.9353
Encrypted image	0.0027	0.0016	−0.0028
Reference [21]	0.0046	0.0043	0.0049
Reference [22]	0.0059	−0.0052	0.0190

4.3. Analysis on Correlation. The correlation coefficient is the evaluation criterion for the degree of linear correlation between two random variables. We randomly selected 2,500 pairs of adjacent pixels (in vertical, horizontal, and diagonal directions) from both the original and encrypted images of Lena and calculated the correlation coefficient for each pair of adjacent pixels according to (14). Table 3 shows the results of correlation coefficients of two adjacent pixels, which are compared with the results in [21, 22]. The results indicate that the correlation of two adjacent pixels of the plain image is significant. Therefore, the encryption effect of this algorithm is pretty good. Therefore, the encryption effect of this algorithm is pretty good:

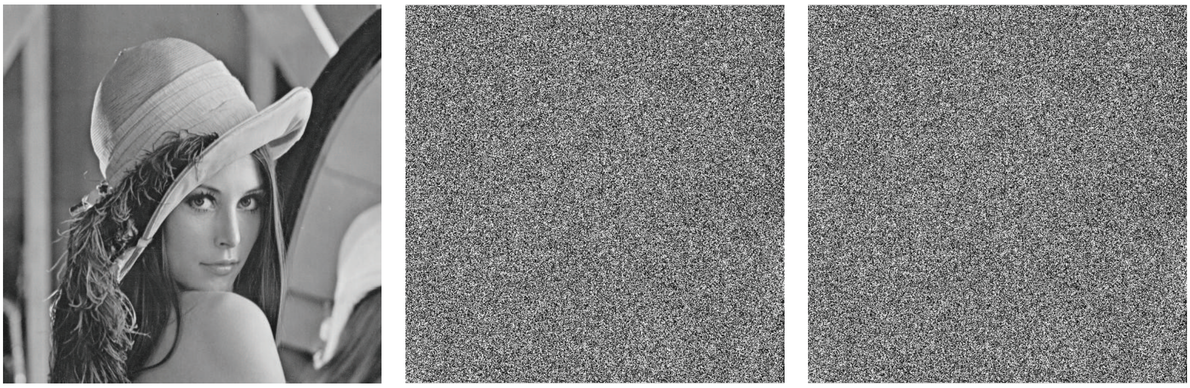
$$r_{xy} = \frac{\operatorname{cov}(x, y)}{\sqrt{D(x)}\sqrt{D(y)}}, \tag{14}$$



(a) The line chart

(b) The encrypted image

Figure 7: The line chart and the encrypted image.



(a) Original image

(b) Encrypted image

(c) Decrypted image

Figure 8: The effect of encrypted and decrypted images.

where $\operatorname{cov}(x, y) = (1/N)\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y))$, $E(x) = (1/N)\sum_{i=1}^{N} x_i$, and $D(x) = (1/N)\sum_{i=1}^{N}(x_i - E(x))^2$.

5. Conclusion

This algorithm uses a chaos function acting on an iterative equation and then combines with XOR operation to encrypt images. The chaos function improves sensitivity to initial value, the iterative equation plays a role of changing the position, and the XOR operation transforms the value of pixels. Such a combination can enhance the effectiveness and security of the encryption algorithm. In addition, this algorithm has a characteristic of cycling the main encryption process many times to achieve an optimal encryption effect. Through several experiments, it can also be proved that the encryption speed is also relatively fast. An ordinary size of the image can be encrypted completely in a few seconds. To summarize the algorithm has the features of being fast, efficient, safe, and convenient. Certainly, to encrypt faster and more sophisticatedly, improvements on chaos and iterative equation can be made.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the Fundamental Research Funds for the Central Universities (DL13CB04), Nature Science Foundation of Heilongjiang Province (LC2012C33), Science Technology Research Projects of Heilongjiang Province Education Department (12533006), and the National Natural Science Foundation of China under Grant 61300098.

References

[1] S. Som and S. Sen, “A non-adaptive partial encryption of grayscale images based on chaos,” *Procedia Technology*, vol. 10, pp. 663–671, 2013.

[2] M. François, T. Grosges, D. Barchiesi, and R. Erra, “A new image encryption scheme based on a chaotic function,” *Signal*

Processing: Image Communication, vol. 27, no. 3, pp. 249–259, 2012.

[3] H. Liu, X. Wang, and A. Kadir, "Image encryption using DNA complementary rule and chaotic maps," *Applied Soft Computing Journal*, vol. 12, no. 5, pp. 1457–1466, 2012.

[4] Y. Zhou, L. Bao, and C. L. P. Chen, "A new 1D chaotic system for image encryption," *Signal Processing*, vol. 97, pp. 172–182, 2014.

[5] A. Bakhshandeh and Z. Eslami, "An authenticated image encryption scheme based on chaotic maps and memory cellular automata," *Optics and Lasers in Engineering*, vol. 51, no. 6, pp. 665–673, 2013.

[6] R. Enayatifar, A. H. Abdullah, and M. Lee, "A weighted discrete imperialist competitive algorithm (WDICA) combined with chaotic map for image encryption," *Optics and Lasers in Engineering*, vol. 51, no. 9, pp. 1066–1077, 2013.

[7] H. Li and Y. Wang, "Double-image encryption based on discrete fractional random transform and chaotic maps," *Optics and Lasers in Engineering*, vol. 49, no. 7, pp. 753–757, 2011.

[8] Q. Zhang, L. Guo, and X. Wei, "Image encryption using DNA addition combining with chaotic maps," *Mathematical and Computer Modelling*, vol. 52, no. 11-12, pp. 2028–2035, 2010.

[9] 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, pp. 2943–2959, 2012.

[10] J. W. Yoon and H. Kim, "An image encryption scheme with a pseudorandom permutation based on chaotic maps," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 12, pp. 3998–4006, 2010.

[11] Q. Zhang and X. Wei, "A novel couple images encryption algorithm based on DNAsubsequence operation and chaotic system," *Optik*, vol. 124, no. 23, pp. 6276–6281, 2013.

[12] M. Shan, J. Chang, Z. Zhong, and B. Hao, "Double image encryption based on discrete multiple-parameter fractional Fourier transform and chaotic maps," *Optics Communications*, vol. 285, no. 21-22, pp. 4227–4234, 2012.

[13] D. Kong and X. Shen, "Multiple-image encryption based on optical wavelet transform and multichannel fractional Fourier transform," *Optics & Laser Technology*, vol. 57, no. 4, pp. 343–349, 2014.

[14] R. Enayatifar, A. H. Abdullah, and I. F. Isnin, "Chaos-based image encryption using a hybrid genetic algorithm and a DNA sequence," *Optics and Lasers in Engineering*, vol. 56, no. 5, pp. 83–93, 2014.

[15] M. François, T. Grosges, D. Barchiesi, and R. Erra, "A new image encryption scheme based on a chaotic function," *Signal Processing: Image Communication*, vol. 27, no. 3, pp. 249–259, 2012.

[16] Y. Liu, J. Tang, and T. Xie, "Cryptanalyzing a RGB image encryption algorithm based on DNA encoding and chaos map," *Optics and Laser Technology*, vol. 60, pp. 111–115, 2014.

[17] X. Wang, L. Teng, and X. Qin, "A novel colour image encryption algorithm based on chaos," *Signal Processing*, vol. 92, no. 4, pp. 1101–1108, 2012.

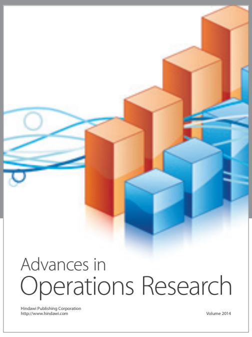
[18] Q. Zhang, L. Guo, and X. Wei, "Image encryption using DNA addition combining with chaotic maps," *Mathematical and Computer Modelling*, vol. 52, no. 11-12, pp. 2028–2035, 2010.

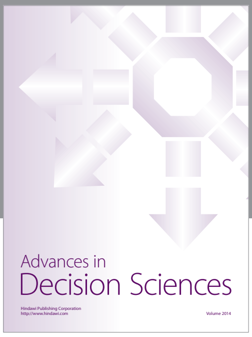
[19] L. Liu, Q. Zhang, and X. Wei, "A RGB image encryption algorithm based on DNA encoding and chaos map," *Computers & Electrical Engineering*, vol. 38, no. 5, pp. 1240–1248, 2012.

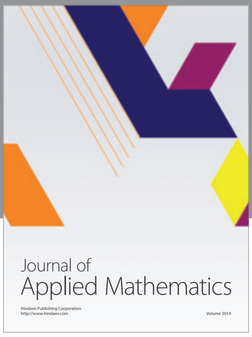
[20] A. ur Rehman, X. Liao, A. Kulsoom, and S. A. Abbas, "Selective encryption for gray images based on chaos and DNA complementary rules," *Multimedia Tools and Applications*, 2014.

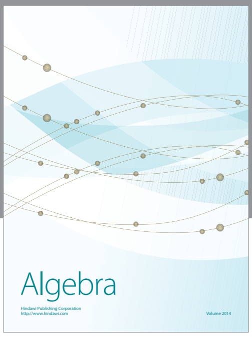
[21] I. Hussain, T. Shah, and M. A. Gondal, "Application of S-box and chaotic map for image encryption," *Mathematical and Computer Modelling*, vol. 57, no. 9-10, pp. 2576–2579, 2013.

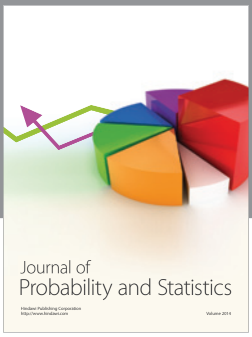
[22] Y. Liu, X. Tong, and S. Hu, "A family of new complex number chaotic maps based image encryption algorithm," *Signal Processing: Image Communication*, vol. 28, no. 10, pp. 1548–1559, 2013.


Advances in
Operations Research

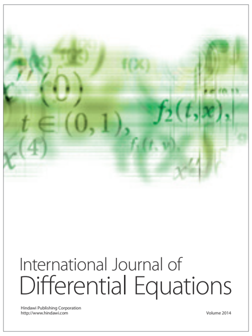

Advances in
Decision Sciences


Journal of
Applied Mathematics

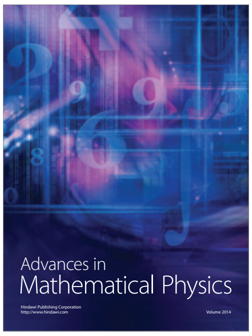

Algebra

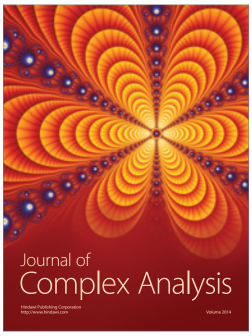

Journal of
Probability and Statistics

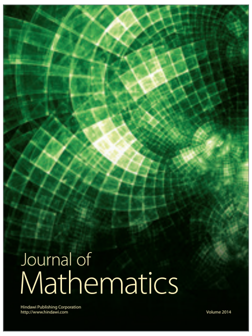
The Scientific
World Journal

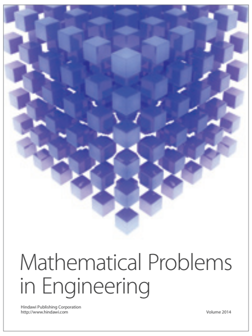

International Journal of
Differential Equations

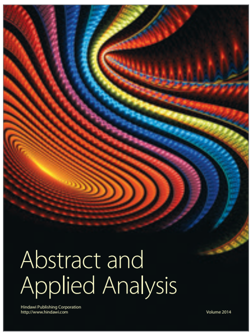
International Journal of
Combinatorics

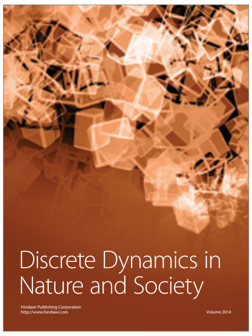

Advances in
Mathematical Physics

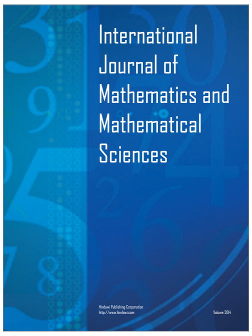

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

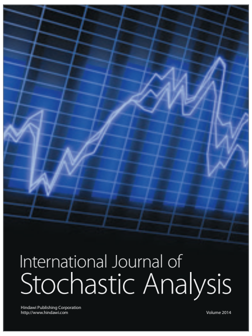

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization