Computers and Mathematics with Applications 59 (2010) 3320–3327

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Color image encryption based on one-time keys and robust chaotic maps

Liu Hongjun [a,b,*], Wang Xingyuan [a]

[a] Department of Electronic and Information Engineering, Dalian University of Technology, Dalian 116024, China
[b] Weifang Vocational College, Weifang 261041, China

ARTICLE INFO

Article history:
Received 25 July 2009
Received in revised form 11 March 2010
Accepted 11 March 2010

Keywords:
Robust Chaotic map
Color image encryption
One-time keys
Perturbation
Pseudo-random sequence
Key stream

ABSTRACT

We designed a stream-cipher algorithm based on one-time keys and robust chaotic maps, in order to get high security and improve the dynamical degradation. We utilized the piecewise linear chaotic map as the generator of a pseudo-random key stream sequence. The initial conditions were generated by the true random number generators, the MD5 of the mouse positions. We applied the algorithm to encrypt the color image, and got the satisfactory level security by two measures: NPCR and UACI. When the collision of MD5 had been found, we combined the algorithm with the traditional cycle encryption to ensure higher security. The ciphered image is robust against noise, and makes known attack unfeasible. It is suitable for application in color image encryption.

1. Introduction

In recent years, the transmission of digital images over communication media has developed greatly. The problem of security in storage and transmission of confidential visual information is therefore growing in importance, and requires solutions for many applications. Most conventional ciphers, such as Data Encryption Standard (DES), International Data Encryption Algorithm (IDEA), Advanced Encryption Standard (AES), linear feedback shift register (LFSR), etc. [1,2] with high computational security consider plain image as either block cipher or data stream and are not suitable for image/video encryption in real time, because their speed is low due to a large data volume and strong correlation among image pixels. The implementation of traditional algorithms for image encryption is even more complicated when undertaken with commercial software.

Color image encryption is different from text encryption due to some inherent features of the image, such as bulk data capacity and high correlation among pixels. Recently, a number of chaos-based image encryption systems [3–5] and random number generation algorithms [6,7] based on discrete chaos were proposed, but security is generally not high enough [8]. In fact, many chaos-based cryptography schemes have shortcomings. Short cycle length is one of the important problems of chaotic key stream generators, which results from the finite precision of computers. It makes various attacks possible [9]. To extend the cycle length of chaotic systems, some chaotic stream ciphers utilized dual chaotic systems to generate a pseudo-random sequence [10]. But a new problem appears: the initial condition and parameter of perturbation map are selected randomly, and remain unchanged to any original image, which is not secure enough to withstand chosen-plain image attack [11].

In order to solve the problem, we designed the one-time key cryptosystem based on two robust chaotic maps. Shannon had proved that one-time pad is a theoretically unbreakable symmetric encryption cipher, and the true random number generator (TRNG) is much safer than pseudo-random number generator (PRNG) [12], then we use TRNG to generate the

* Corresponding author at: Department of Electronic and Information Engineering, Dalian University of Technology, Dalian 116024, China.
E-mail addresses: smithliu@126.com (H. Liu), wangxy@dlut.edu.cn (X. Wang).

0898-1221/$ – see front matter
doi:10.1016/j.camwa.2010.03.017

keys by the Message-Digest algorithm 5 (MD5) of the mouse positions. Our basic source of entropy comes from sampling the mouse positions to ensure that keys have a high level of entropy.

Every item in the key stream is generated by different initial condition from perturbation map, parameter and iteration times. Therefore, two images with different size, the ciphered images will be different obviously. For two original images with the same size, they may be one of the follows: (1) have only one different pixel; (2) have more than one different pixel; (3) the pixels are all same. The ciphered images will be different completely if being encrypted by the proposed cryptosystem, which ensures a satisfactory level of security by two measures: the number of pixel change rate (NPCR) and unified average changing intensity (UACI).

Finally, we provided the algorithm of encryption and decryption, the plain image can be recovered completely if the initial condition and parameter are exactly known. The cryptosystem is robust against cryptographic attacks, for the key space is large enough.

2. Chaos-based image encryption scheme

Any image in 24-bit true color can be converted into its RGB components. The matrix (R, G or B) of each color is converted into a vector of integers between 0 and 255. Each vector has a length of L = width × height ($W \times H$). Then the plain image $\boldsymbol{P}(3L)$ will be encrypted.

$$\boldsymbol{P} = \{R_n^P, G_n^P, B_n^P\} \quad (n = 1, 2, \ldots, L). \tag{1}$$

For shorter EDT, the length of ciphered image should be equal to the length of plain image. The proposed encryption algorithm is to create ciphered image $\boldsymbol{C}$:

$$\boldsymbol{C} = \{R_n^C, G_n^C, B_n^C\} \quad (n = 1, 2, \ldots, L). \tag{2}$$

It is known that the security of an encryption algorithm is determined by its confusion and diffusion properties and its sensitivity to secret keys. The basic idea underlying our encryption algorithm is to use chaotic systems for both the confusion and diffusion processes [13].

2.1. Generation of the key stream

Firstly, we will generate the key stream sequence of $\boldsymbol{X}$:

$$\boldsymbol{X} = \{x_1^R = x_1, x_1^G = x_2, x_1^B = x_3, \ldots, x_i, \ldots, x_L^B = x_{3L}^B\}. \tag{3}$$

The key sequence of x_i is given by the piecewise linear chaotic map (PWLCM) in Eq. (4):

$$x_{i+1} = F_{p_i}(x_i) = \begin{cases} x_i/p_i, & 0 \le x_i < p_i \\ (x_i - p_i)/(0.5 - p_i), & p_i \le x_i < 0.5 \\ F_p(1 - x_i), & x_i \ge 0.5 \end{cases} \tag{4}$$

where $x_i \in [0, 1)$, control parameter $p_i \in (0, 0.5)$. When $p_i \in (0, 0.5)$, Eq. (4) evolves into chaotic state. PWLCM has uniform invariant distribution and very good ergodicity, confusion and determinacy, so it can provide excellent random sequence, which is suitable for cryptosystem. The plot of PWLCM system is showed in Fig. 1.

Here we set $p_i = z_i/2$ to ensure $p_i \in (0, 0.5)$, p_i can be served as secret key. Before iteration, the parameter p_i is modified by z_i from the perturbation sequence $\boldsymbol{Z}$ generated by the Chebyshev maps. The expression of Chebyshev maps is as follows:

$$z_{i+1} = \cos(w \cos^{-1} z_i), \quad -1 \le z_i \le 1 \tag{5}$$

where w is the degree of Chebyshev maps. Its corresponding invariant density is as follows.

$$\rho(z) = 1 \Big/ \left(\pi\sqrt{1 - z^2}\right). \tag{6}$$

Chebyshev maps have important properties of excellent cryptosystem [14]. If $w \ge 2$, the Lyapunov exponent of Chebyshev maps is positive, as it showed in Fig. 2, which predicates that Chebyshev maps are chaotic. The real number sequences generated by Chebyshev maps are orthogonal polynomial sequences. Furthermore, their correlation functions are all δ function.

According to Shannon's theory of perfect-secrecy, secret keys must be generated in a random way with a high level of entropy, our basic source for entropy comes from mouse-position. The initial condition of $x_0 \in [0, 1)$ and $z_0 \in (0, 1)$ as one-time keys are generated by truly random number generators (TRNG): the MD5 of the mouse-position from entropy.

In cryptography, MD5 is a widely used cryptographic hash function with a 128-bit hash value. A MD5 hash is typically expressed as a 32 digit hexadecimal number, we store it in array $M(i)$ ($i = 1, 2, \ldots, 32$). By Eq. (7) we get x_0 as the initial condition of map Eq. (4).

$$x_0 = \left(\sum_{i=1}^{32} M(i) \Big/ 571\right) \bmod 1. \tag{7}$$

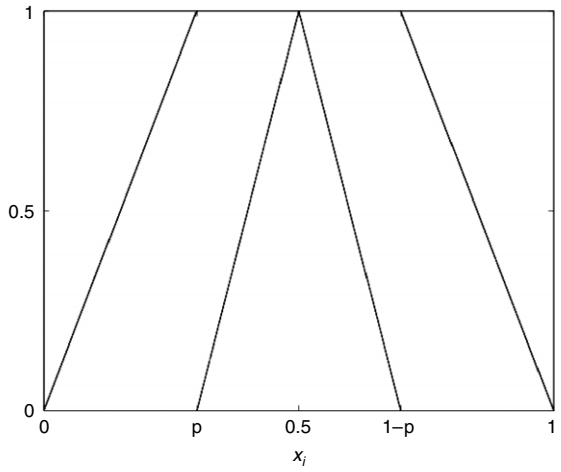
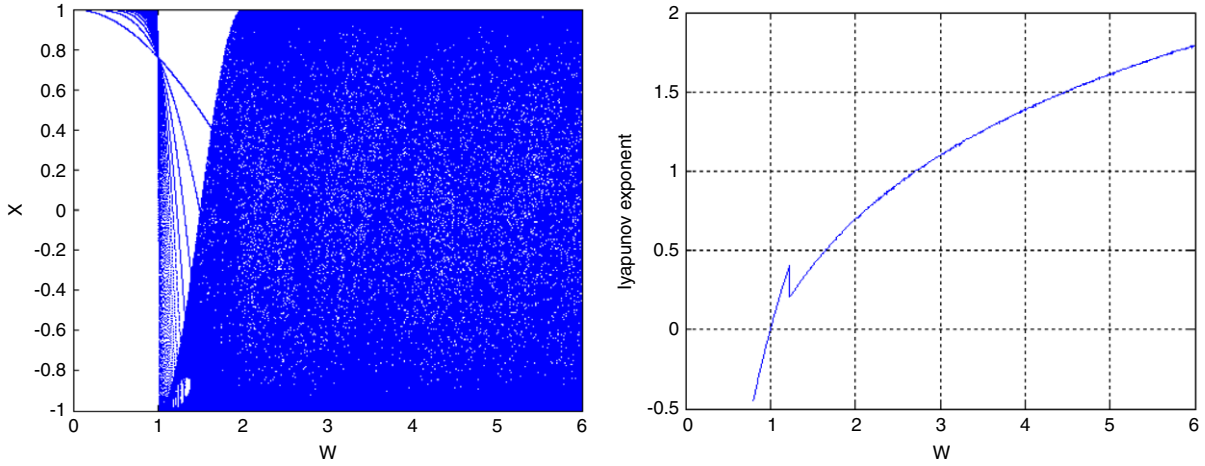


Fig. 1. The plot of PWLCM system.



(a) The bifurcate image.

(b) The Lyapunov exponent image.

Fig. 2. The bifurcate and Lyapunov exponent image of Chebyshev maps.

We select a prime number such as 571 to get a floating-point irrational number, and retain the 15-bit decimal. We can get z_0 in the same way as x_0. Finally we obtain the sequence $\boldsymbol{X}$.

Eq. (3) can be separated into Eqs. (8)–(10):

$$R_n^X = \{x_1^R, x_2^R, \ldots, x_L^R\}, \tag{8}$$

$$G_n^X = \{x_1^G, x_2^G, \ldots, x_L^G\}, \tag{9}$$

$$B_n^X = \{x_1^B, x_2^B, \ldots, x_L^B\}. \tag{10}$$

Then we can rewrite Eq. (3) as follows:

$$\boldsymbol{X} = \{R_n^X, G_n^X, B_n^X\} \quad (n = 1, 2, \ldots, L). \tag{11}$$

2.2. Design of the encryption scheme

First, we generate the key stream $\boldsymbol{X}$ by map Eq. (4), the rows of P being the R, G and B components. Then, the encryption transformation of the proposed scheme is given by:

$$R_n^C = (R_n^P + R_{n-1}^C) \bmod 256 \oplus R_n^X. \tag{12}$$

$$G_n^C = (G_n^P + G_{n-1}^C) \bmod 256 \oplus G_n^X. \tag{13}$$

$$B_n^C = (B_n^P + B_{n-1}^C) \bmod 256 \oplus B_n^X. \tag{14}$$

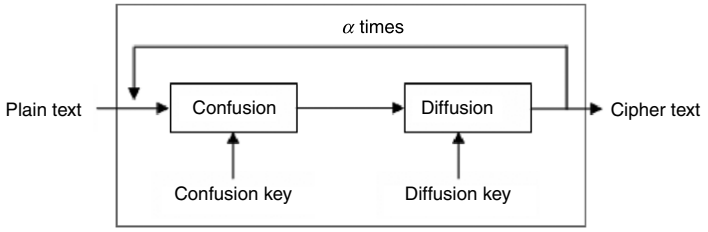


Fig. 3. General scheme of a cryptosystem.

(a) Original image. (b) Encrypted image. (c) Decrypted image.

Fig. 4. The image encryption/decryption experimental result.

(a) Original image. (b) Encrypted image. (c) Decrypted image.

Fig. 5. The image changed in small encryption/decryption experimental result.

We set the initial value of R_0^C, G_0^C and B_0^C to zero. On account of the collisions of MD5 have been found [15,16], here we substitute $\boldsymbol{C}$ to $\boldsymbol{P}$ and repeat Eqs. (12)–(14) α times to enhance the security. Typically, α is selected between 3 and 11 to provide shorter EDT. The key space of cryptosystem in Fig. 3 is defined as $S = (S_C S_D)^\alpha$.

It is clear that the generation of key stream $\boldsymbol{X}$ depends on the perturbation sequence $\boldsymbol{Z}$ and dynamical iteration times n_i, we set it change with z_i, it can be served as secret key, where

$$n_i = 200 + \lfloor z_i \times 100 \rfloor \,\%256. \tag{15}$$

Moreover, owning to the coupling structure of the algorithm, the stream cipher of the color components depend on each other. These features strengthen the encryption algorithm security, and make the encryption algorithm robust against the chosen-plain image attack presented in Ref. [17].

2.3. Design of the decryption scheme

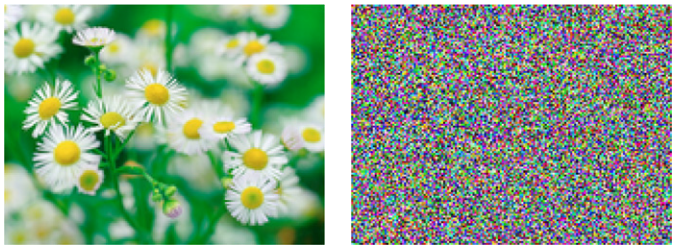
In the receiver side, as the encryption scheme, the receiver must generate the same key stream $\boldsymbol{X}$ by Eq. (4) to decrypt the ciphered image, according to the same initial condition x_0 and z_0 provided to the receiver. The ciphered image can be decrypted as follows:

$$R_n^P = (R_n^C \oplus R_n^X - R_{n-1}^C) \bmod 256, \tag{16}$$

$$G_n^P = (G_n^C \oplus G_n^X - G_{n-1}^C) \bmod 256, \tag{17}$$

$$B_n^P = (B_n^C \oplus B_n^X - B_{n-1}^C) \bmod 256. \tag{18}$$

Just like the encryption, the decryption will also substitute $\boldsymbol{P}$ to $\boldsymbol{X}$ and repeat Eqs. (16)–(18) α times.



(a) Original image.

(b) Encrypted image, $z_0 = 0.602451838879159$.

(c) Decrypted image, $z_0 = 0.6024518388791591$.

(d) Decrypted image, $z_0 = 0.602451838879159$.

Fig. 6. Key sensitive test.

Figs. 4 and 5 show the encryption and decryption of two color images, we set a black pixel on the white petal of the right flower in Fig. 5(a). Compared Fig. 4(b) with Fig. 5(b), we can see that the ciphered images are different completely.

3. Performance and security analysis

We have made several tests to check the security of the proposed cryptosystem. Statistical tests include histogram analysis, calculation of the correlation coefficients of adjacent pixels. Security tests against differential attack include calculation of the NPCR and UACI, and information entropy evaluation.

3.1. Key space

The high sensitive to initial conditions inherent to any chaotic system, i.e. exponential divergence of chaotic trajectories, ensure high security. To provide an encryption algorithm with high security, the key space should be large enough to make any brute force attack ineffective. Our encryption algorithm actually does have some of the following secret keys: (1) initial condition x_0 and z_0, (2) map parameters p_i, (3) iteration times n_i.

The sensitive of our algorithm to z_0 is illustrated in Fig. 6. When the error in the initial condition $\Delta z_0 = 10^{-16}$, the decrypted image is still indistinguishable, but when $\Delta z_0 = 10^{-17}$, the ciphered image can be decrypted successfully.

The variation of z_0 within chaotic attractor $A \in [0, 1]$ with a step 10^{-16}, so the key space for initial condition z_0 is $S_{z_0} \approx 10^{16}$.

To ensure a large divergence of a chaotic trajectory from the initial condition, the iteration times should be relatively large but not too much so, for short EDT (usually $n < 1000$). We provided an increase of the key space dimension in Eq. (5) by 10^3, i.e., $S_n \approx 10^3$.

The variation of the parameter w in the chaotic region is between 2 and 6 with a step of 10^{-7}, so $S_w \approx 5 \times 10^7$. Because of $p_i = z_i/2$, and $z_i \in [0, 1]$, then $S_{p_i} = S_{z_0} \approx 10^{16}$.

The total key space includes the key spaces for confusion and diffusion processes, S_C and S_D. In our cryptosystem, $S_C = S_w S_{p_i} S_n \approx 5 \times 10^{26}$ and $S_D = S_{z_0} \approx 10^{16}$. The cryptosystem key space total dimension can be calculated as $S = (S_C S_D)^{\Delta\alpha} = (S_c S_{\mu_i} S_n S_{z_0})^{\Delta\alpha}$, in general, $\Delta\alpha = \alpha_{\max} - \alpha_{\min} = 11 - 3 = 8$, then $S = (S_w S_{p_i} S_n S_{z_0})^{\Delta\alpha}$. The simple estimation shows that the proposed image cipher has $S \approx 3.9 \times 10^{341}$ different combinations of the secret keys.

3.2. Differential attack

As a general requirement for all the image encryption schemes, the encrypted image should be greatly different from its original form. Such difference can be measured by means of two criteria namely, the NPCR and the UACI. The proposed one-time pad cryptosystem can ensure two ciphered images different completely, even if there is only one bit difference between them.

Table 1
The mean NPCR and UACI of ciphered images by changing their original images one pixel.

Image	Mean NPCR (%)			Mean UACI (%)		
	R	*G*	*B*	*R*	*G*	*B*
Fig. 3(a)	99.5983	99.6022	99.5392	33.6883	33.1105	33.5039
Fig. 4(a)	99.6679	99.5885	99.6431	33.4289	33.7420	33.2026

Table 2
Correlation coefficients of plain image and ciphered image.

Correlation	Vertical	Horizontal	Diagonal
Plain image	0.9856	0.9682	0.9669
Ciphered image	−0.0318	0.0965	0.0362

Here are the formulas to calculate $\mathrm{NPCR}_{R,G,B}$ and $\mathrm{UACI}_{R,G,B}$:

$$\mathrm{NPCR}_{R,G,B} = \frac{\sum_{i,j} D_{R,G,B}(i,j)}{W \times H} \times 100\%,$$

$$\mathrm{UACI}_{R,G,B} = \frac{1}{L}\left[\sum_{i,j} \frac{\left|C_{R,G,B}(i,j) - C'_{R,G,B}(i,j)\right|}{255}\right] \times 100\%$$

where W and H represent the width and height of the image respectively. $C_{R,G,B}$ and $C'_{R,G,B}$ are respectively the ciphered images before and after one pixel of the plain image is changed. For the pixel at position (i,j), if $C_{R,G,B}(i,j) \neq C'_{R,G,B}(i,j)$, let $D_{R,G,B}(i,j) = 1$; else let $D_{R,G,B}(i,j) = 0$. Table 1 shows the results according to the proposed algorithm of $\mathrm{NPCR}_{R,G,B}$ and $\mathrm{UACI}_{R,G,B}$. We have found that the NPCR is over 99% and the UACI is over 33%, showing thereby that the encryption scheme is very sensitive with respect to small changes in the plain image. Here we set the encryption rounds α to 11.

3.3. Statistical analysis

Randomly select 1000 pairs of adjacent pixels (in vertical, horizontal and diagonal directions) from the plain image and ciphered image, and calculate the correlation coefficients of two adjacent pixels according to the following formula [18]:

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

where

$$\mathrm{cov}(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)), \quad E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i, \quad D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2.$$

Table 2 shows the results of correlation coefficients of two adjacent pixels in Figs. 4(a), 5(a). The result indicates that the correlation of two adjacent pixels of the plain image is significant, while that of the ciphered image is very small, so the encryption effect is rather good.

3.4. Information entropy analysis

Information entropy is the most important feature of randomness. Let m be the information source, and the formula for calculating information entropy is:

$$H(m) = \sum_{i=0}^{2^n-1} p(m_i)\log_2 \frac{1}{p(m_i)} \tag{19}$$

where $p(m)$ represents the probability of symbol m. Assume that there are 2^8 states of the information source and they appear with the same probability, according to Eq. (19), we can get the ideal $H(m) = 8$, which shows that the information is random. Hence the information entropy of the ciphered image should be close to 8 after encryption. The closer it gets to 8, the less possible for the cryptosystem to divulge information. We use Eq. (19) to calculate the information entropy of the ciphered image of Fig. 4(b), Table 3 shows the entropy of the three color components (R, G, B): which are all close to the ideal value 8, so the probability of accidental information leakage is very little.

Table 3
The result of information entropy.

Color	R	G	B
$H(m)$	7.9851	7.9852	7.9832

Table 4
Comparison of experiments results between different algorithms.

Algorithms	Encryption speed (MB/s)
Algorithms 1	4.06
Algorithms 2	16.10
Algorithms 3	4.62
Proposed algorithm	9.28

(a) 0.1% salt & pepper noise. (b) 0.5% salt & pepper noise. (c) 1% salt & pepper noise.

Fig. 7. When the ciphered image was affected, the decrypted images maintain the overall original image information.

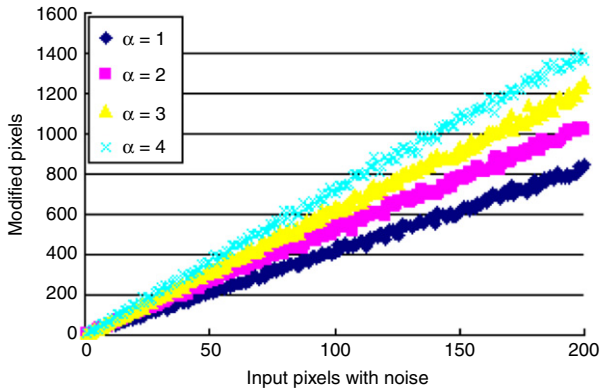


Fig. 8. Sensitivity to noise, number of erroneous output pixels modified by noise, $W \times H = 100 \times 100$.

3.5. Encryption speed

We have used Matlab 7.0 to run encryption and decryption programs in a personal computer with a Pentium 4 CPU 2.1 GHz, 512 MB memory and 80 GB hard-disk capacity, and the operation system is Microsoft Windows XP. Table 4 shows the comparison of experiments results between the proposed cryptosystem and other one-dimensional chaotic cryptosystems without any improvement. Algorithm 1 uses a logistic map and a two-value threshold function to generate the 0–1 stream [17]. Algorithm 2 magnifies the states of the logistic map, then encrypts the plain image after rounding and module operations [10]. Algorithm 3 is the same to Algorithm 2 except for replacing the logistic map with Chebyshev maps [19]. Compared to other one-dimensional chaotic cryptosystems, we can see that the operation speed of the proposed cryptosystem is fast.

3.6. Robustness against noise

Most of the commonly used cryptosystems are very sensitive to noise, a small change in the ciphered image may induce a strong distortion in the decrypted image, which does not allow one to recuperate the original image. Instead, the proposed cryptosystem is robust against noise. Fig. 7 shows when the ciphered image was affected by salt & pepper noise with different

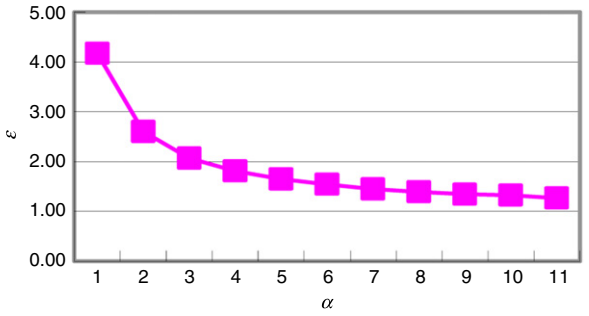


Fig. 9. The curve of $\varepsilon \approx N_{out}/(\alpha N_{in})$.

percent and $\alpha = 3$, the decrypted images maintain the overall original image information for the human eye. Fig. 8 shows the sensitivity of our cryptosystem to noise, The number of erroneous output pixels modified by noise (N_{out}) is proportional to the number of noise added to the ciphered image (N_{in}).

In our cryptosystem, we calculate the mean value of $\varepsilon \approx N_{out}/(\alpha N_{in})$ with the cycle times α from 1 to 11, we can see from Fig. 9 that ε is not stationary, but decreases with the increase of α, i.e., the growth rate of N_{out} becomes slower with the increase of α and N_{in}.

4. Conclusion

In this paper, we have presented the color image encryption based on one-time keys and robust chaotic maps, got the satisfactory NPCR and UACI, and solved the problem of degradation caused by finite precision of computer. The proposed algorithm combines good confusion and diffusion properties by repeating encryption α times. The analysis shows that the proposed cryptosystem has higher security due to an extremely large key space. The experimental results demonstrate its robustness against noise and small external disturbances allows us to recover original image in the presence of noise. The encryption and decryption algorithms are symmetric, making it suitable for color image encryption.

Acknowledgements

The research is supported by the National Natural Science Foundation of China (Nos. 60973152 and 60573172), the Superior University doctor subject special scientific research foundation of China (No. 20070141014), the National Natural Science Foundation of Liaoning province (No. 20082165) and China Visiting Scholars Program for Outstanding young college teachers in Shandong Province.

References

[1] B. Schneier, Applied Cryptography — Protocols, Algorithms, and Source Code, second ed., C. John Wiley & Sons, Inc., New York, 1996.
[2] J. Daemen, B. Sand, V. Rijmen, The Design of Rijndael: AES — The Advanced Encryption Standard, Springer-Verlag, Berlin, 2002.
[3] G. Chen, Y. Mao, C.K. Chui, A symmetric image encryption scheme based on 3D chaotic cat maps, Chaos, Solitons & Fractals 21 (3) (2004) 749–761.
[4] E. Alvarez, A. Fernandez, P. Garcia, J. Jimenez, A. Marcano, New approach to chaotic encryption, Physics Letters A 263 (4–6) (1999) 373–375.
[5] P. Li, Z. Li, W.A. Halang, G. Chen, A stream cipher based on a spatiotemporal chaotic system, Chaos, Solitons & Fractals 32 (5) (2007) 1867–1876.
[6] M.S. Baptista, Cryptography with chaos, Physics Letters A 240 (1–2) (1998) 50–54.
[7] T. Xiang, X. Liao, G. Tang, Y. Chen, K.W. Wong, A novel block cryptosystem based on iterating a chaotic map, Physics Letters A 349 (1–4) (2006) 109–115.
[8] X. Wu, H. Hu, B. Zhang, Parameter estimation only from the symbolic sequences generated by chaos system, Chaos, Solitons & Fractals 22 (2) (2004) 359–366.
[9] D.R. Stinson, Cryptography: Theory and Practice, CRC Press, Boca Raton, FL, 1995.
[10] A.N. Pisarchik, M. Zanin, Image encryption with chaotically coupled chaotic maps, Physica D 237 (20) (2008) 2638–2648.
[11] X.Y. Wang, C.H. Yu, Cryptanalysis and improvement on a cryptosystem based on a chaotic map, Computers and Mathematics with Applications 57 (3) (2009) 476–482.
[12] Y. Hu, X.F. Liao, K. Wong, Q. Zhou, A true random number generator based on mouse movement and chaotic cryptography, Chaos, Solitons & Fractals 40 (3) (2009) 2286–2293.
[13] C.E. Shanon, Communication theory of secrecy systems, Bell System Technical Journal 28 (4) (1949) 656–715.
[14] J.M. Amigoa, L. Kocarevb, J. Szczepanski, Theory and practice of chaotic cryptography, Physics Letters A 366 (3) (2007) 211–216.
[15] E. Thompson, MD5 collisions and the impact on computer forensics, Digital Investigation 2 (1) (2005) 36–40.
[16] C. Cid, Recent developments in cryptographic hash functions: security implications and future directions, Information Security Technical Report 11 (2) (2006) 100–107.
[17] N.K. Pareek, V. Patidar, K.K. Sud, Image encryption using chaotic logistic map, Image and Vision Computing 24 (9) (2006) 926–934.
[18] R. Rhouma, S. Meherzi, S. Belghith, OCML-based colour image encryption, Chaos, Solitons & Fractals 40 (1) (2009) 309–318.
[19] L.H. Zhang, Cryptanalysis of the public key encryption based on multiple chaotic systems, Chaos, Solitons & Fractals 37 (3) (2008) 669–674.