International Journal of Security and Its Applications
Vol.9, No.1 (2015), pp.25-34
http://dx.doi.org/10.14257/ijsia.2015.9.1.03

Improved Tent Map and Its Applications in Image Encryption

Xuefeng Liao

Oujiang College, Wenzhou University, Wenzhou, China 325035
liaoxuef@163.com

Abstract

In this paper, an improved tent map chaotic system with better chaotic performance is introduced. The improved tent map has stronger randomness, larger Lyapunov exponents and C_0 complexity values compare with the tent map. Then by using the improved tent map, a new image encryption scheme is proposed. Based on all analysis and experimental results, it can be concluded that, this scheme is efficient, practicable and reliable, with high potential to be adopted for network security and secure communications. Although the improved tent chaotic maps presented in this paper aims at image encryption, it is not just limited to this area and can be widely applied in other information security fields.

Keywords: *Improved tent map; Chaos performance; Image encryption; Shuffle; Diffuse*

1. Introduction

With the rapid development in digital image processing and network communication, information security has become an increasingly serious issue. All the sensitive data should be encrypted before transmission to avoid eavesdropping. However, bulk data size and high redundancy among the raw pixels of a digital image make the traditional encryption algorithms, such as DES, IDEA, AES, not able to be operated efficiently. Therefore, designing specialized encryption algorithms for digital images has attracted much research effort. Because chaotic systems/maps have some intrinsic properties, such as ergodicity, sensitive to the initial condition and control parameters, which are analogous to the confusion and diffusion properties specified by Shannon [1]. Thus makes it natural to employ chaotic systems in image encryption algorithms [2–11]. Therefore, many chaotic image encryption algorithms have been developed by directly utilizing existing chaotic systems/maps [8-11]. But the chaotic performance by directly utilizing existing chaotic systems/maps is not perfect.

Chaotic ciphers are chaos-dependent. Their security and efficiency are typically affected by the underlying chaotic systems. One-dimensional chaotic systems are very suitable for real time image encryption because of their simple structure and easiness to implement [12]. But, one-dimensional chaotic systems also have some problems, such as the limited or/and discontinuous range of chaotic behaviors [13], the short period and the nonuniform data distribution of output chaotic sequences. Hence, improving the existing chaotic systems to get better chaotic performance is needed.

2. The Improved Tent Map

Tent map (TM) is a classical one-dimensional chaotic system, its definition is

$$x_{n+1} = f(x_n) = \begin{cases} 2x_n, & if\ 0 \le x_n \le 1/2 \\ 2(1-x_n), & if\ 1/2 < x_{n+1} \le 1 \end{cases} \quad (1)$$

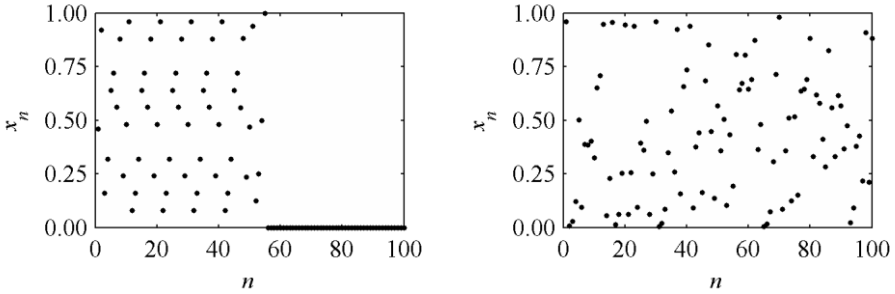
ISSN: 1738-9976 IJSIA

When $x_0 \in (0, 1)$, system (1) evolves into a chaotic state. To improve the chaotic performance, we generalize the tent map to a improved tent map (ITM) with chaotic behavior in the interval (0, 1). The improved tent map can be defined as

$$x_{n+1} = f(x_n) = \begin{cases} (x_n - a \times \text{floor}(x_n / a)) / a, & if\ \text{floor}(x_n / a)\ is\ even \\ a \times (\text{floor}(x_n / a) + 1) - x_n) / a, & if\ \text{floor}(x_n / a)\ is\ odd \end{cases} \quad (2)$$

Where $x_n \in (0,1)$, $a \in (0, 0.5)$, a is the system parameter. floor(x) is the function that returns the nearest integer smaller than or equal to x. Note that system (1) is the special case of system (2) with a=0.5.

To study the dynamical behavior of the systems (1) and (2), we plot the graphs of the two systems under a large set of random parameters. Due to the similarity of the graphs, only the graphs of x_0=0.23 and a=0.24 are shown in Figure 1. From Figure 1(a), one can see that the state values of system (1) fall into the fixed point x_n=0 after 56 times iteration. However, the state values of system (2) distribute in the range (0, 1) randomly. So, the randomness of system (2) is stronger than that of system (1).



(a) The Tent Map (b) The Improved Tent Map

Figure 1. Graphs of the Chaotic Systems

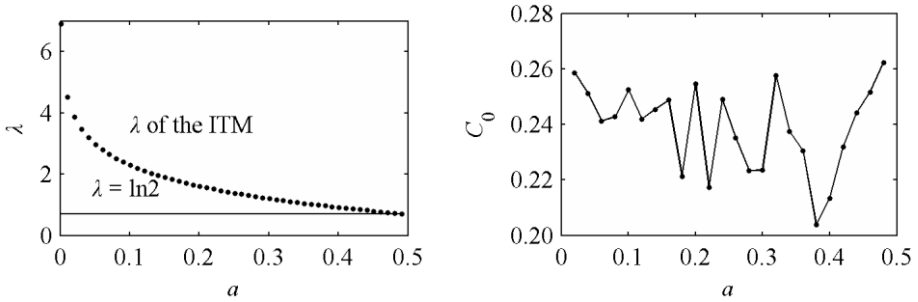
To compare further the chaotic performance between systems (1) and (2), one can use Lyapunov exponent to measure the chaotic behavior. A chaotic system has a positive Lyapunov exponent, and larger Lyapunov exponent imply better chaotic performance. For 1D systems, Lyapunov exponent can be calculated by Eq.(3).

$$\lambda = \lim_{n \to \infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln \left| f'(x_i) \right| \quad (3)$$

Where $f'(x_i)$ is the derivation of the function $f(x)$ at the point of $x=x_i$. It can be shown that the Lyapunov exponent of system (1) is equal to ln2. Thus, the aperiodic signals generated by system (1) are chaotic. But Lyapunov exponents of system (2) can be calculated by Eq.(3) numerically, which are shown in Figure 2(a) versus system parameters a. As a result, the Lyapunov exponents of system (2) are greater than ln2 as long as a<0.5. So, the chaotic performance of the system (2) is better than that of the system (1).

As a new method, C_0 complexity can be used to measure the complexities of chaotic systems [14]. Larger C_0 complexity values imply more complex. We use the C_0 complexity algorithm to measure the complexity of the tent map and the improved tent map. The C_0 complexity value of TM is equal to 0.0012, which is a constant. While ITM has different C_0 complexity values corresponding to different system parameters a. The C_0 complexity values versus system parameters a are shown in Figure 2(b). From

Figure 2 (b) one can see that ITM system is more complex than TM system in a wider parameter range.



(a) Lyapunov Exponents (b) C_0 complexity Values

Figure 2. Lyapunov Exponents and C_0 Complexity Values of the ITM

The results as stated above indicate that the ITM chaotic system has more excellent chaotic performance, and is more suitable for applications in the field of secure communication and information encryption.

3. The Proposed Image Encryption Scheme

To investigate the applications of the improved tent map in information security, we propose a new image encryption scheme by using the ITM chaotic system in this section. The proposed scheme has a permutation–diffusion architecture. The algorithm uses different key streams when encrypting different plain-images (even with the same key). The algorithm generates a **T** sequence with the same size of plain-image by the ITM chaotic system, which shuffles the positions of pixels totally. The key stream in the diffusion step depends on both the key (the initial value x_0 and the control parameters a of the ITM) and the plain-image.

The plain-image to be encrypted is a 256 gray-scale image of size $L=M_1\times M_2$, in which the pixel values range from 0 to 255. Its data can be treated as a one-dimensional vector $\mathbf{P}=\{p(1), p(2), \cdots, p(L)\}$. Suppose the permuted image pixels is denoted by $\mathbf{P}'=\{p'(1), p'(2), \cdots, p'(L)\}$, and the final cipher-image pixel sequence is denoted by $\mathbf{C}=\{c(1), c(2), \cdots, c(L)\}$. The secret keys include five parameters (x_{10}, a_1, x_{20}, a_2, s), where s is related to the plain-image. Further, two integers C_0 and N_0 are used, where $C_0 \in [1, 255]$, and N_0 is larger than 500.

3.1. Encryption Algorithm

3.1.1. Permutation Operator based on the ITM: Given a plain-image and parameters (x_{10}, a_1, x_{20}, a_2, C_0, N_0), we firstly generate a permutation sequence **T** to change the positions of pixels in the plain-image. Suppose $\mathbf{T}=\{t(1),t(2),\ldots, t(L)\}$, where $t(i)$ are integers, $t(i)\in[1, L]$ and $t(i)\neq t(j)$ if $i\neq j$. Our permutation scheme takes the following steps:

Step 1 Calculate the sum of all pixel values s, the minimum m_1 and the maximum m_2 of the plain-image $\mathbf{P}=\{p(1),p(2),\ldots, p(L)\}$. Note that $m_1=m_2$ if and only if $p(i)$ is a constant, where $i\in\{1,2,\ldots,L\}$. The permutation process can be simply ignored when this situation occurs during encryption. Otherwise, continue to execute Steps 2-10.

Step 2 Let $x\leftarrow x_{10}\times s/(L\times 255)$, $a\leftarrow a_1$, and initialize the permutation sequence $t(i)=i$, $i=1,2, \ldots,L$.

Step 3 Iterating the ITM for one times to obtain a new x, then computing a integer j by using current x according to the following formula:

$$j = \mathrm{mod}(\mathrm{floor}(x \times 10^{12}), \mathrm{L}) + 1. \quad (4)$$

Where mod(x, y) returns the remainder after division.

Step 4 Swapping the pixels in position 1 and j:

$$p'(j) \leftarrow p(1),\ p'(1) \leftarrow p(j),\ t(1) \leftarrow j,\ t(j) \leftarrow 1. \quad (5)$$

Step 5 Let $i \leftarrow 2$.

Step 6 Checking the value $t(i)$, if $t(i)=i$, then go to step 7; else then go to step 10.

Step 7 Iterating the ITM for one times to obtain a new x, then computing a integer j by using current x according to formula (4).

Step 8 Checking the value j, if $j=i$ *or* $j \neq t(j)$, then repeat Step 7; else go to Step 9.

Step 9 Swapping the pixels in position i and j, and swapping the values $t(i)$ abd $t(j)$:

$$p'(i) \leftarrow p(j),\ p'(j) \leftarrow p(i);\ \mathrm{temp} \leftarrow t(i),\ t(i) \leftarrow t(j),\ t(j) \leftarrow \mathrm{temp}. \quad (6)$$

Step 10 Let $i \leftarrow i+1$, repeat Steps 6-9 until i reaches L.

Note that the pixel at the position i in the original pixel sequence **P** is moved to the position $t(i)$ in the permuted pixel sequence **P'**.

3.1.2. Diffusion Process based on the ITM: Using another initial state value x_{20} and a_2 of ITM, we generate another chaotic sequence to diffuse the pixel values of the image **P'**. The diffusion procedure presented in the proposed scheme is composed of two rounds. In each diffusion round, a equivalent secret key sequence **K**={$k(1)$, $k(2)$, ⋯, $k(L)$} is generated according to the current chaotic state value x, which is related to the previous pixel value of current image pixel sequence. The operation steps in diffusion procedure are as follows:

Step 1 Let $x \leftarrow x_{20} \times s/(\mathrm{L} \times 255)$, $a \leftarrow a_2$, and iterating the ITM for N_0 times to obtain a new state value, denoted as x_1. To initialize the cipher-image sequence $c(i)=p'(i)$, i=1,2, …,L.

Step 2 Let $d \leftarrow C_0$, $I \leftarrow 1$.

Step 3 Let $i \leftarrow 1$.

Step 4 Let $x \leftarrow x_1 \times d/255$, and iterating the ITM for one times to obtain a new x.

Step 5 Computing $k(i)$ by using current x according to the following formula (7):

$$k(i) = \mathrm{mod}(\mathrm{floor}(x \times 10^{12}), 255) + 1 \quad (7)$$

Step 6 To encrypt the i-th pixel by using the following formula (8):

$$c(i) = \mathrm{mod}(k(i) + d, 256) \oplus c(i) \quad (8)$$

Step 7 Let $d \leftarrow c(i)$, $i \leftarrow i+1$.

Step 8 If i<=L, then repeat Steps 4-7; else goto Step 9.

Step 9 $I \leftarrow I+1$.

Step 10 If I<=2, repeat Steps 3-9; else the cipher-image is obtained.

3.2. Decryption Algorithm

The decryption procedures are similar to those of encryption except the following modifications: (1) Permutation and Diffusion are executed in reverse order, (2) Diffusion must also be executed reversely, namely, the order of round is I=2→1, and

the order of pixel is i=L→1. The secret key parameters (x_{10}, a_1, x_{20}, a_2, s) and constants C_0 and N_0 are known.

(1) Remove the effect of diffusion

In this process, we obtain **P'**={p'(1), p'(2), ⋯, p'(L)} from **C**={c(1), c(2),⋯, c(L)}.

Step 1 Let x←x_{20}×s/(L×255), a←a_2, and iterating the ITM for N_0 times to obtain a new state value, denoted as x_1. To initialize the permuted-image sequence $p'(i)=c(i)$, i=1,2, …,L.

Step 2 Let d←c(L-1), I←2.

Step 3 Let i←L.

Step 4 Let x←x_1×d/255, and iterating the ITM for one times to obtain a new x.

Step 5 Computing $k(i)$ by using current x according to the formula (7).

Step 6 To decrypt the i-th pixel by using the following formula (9):

$$p'(i) = \mathrm{mod}(k(i)+d,256) \oplus p'(i) \tag{9}$$

Step 7 Let i←i-1; if i=1 and I=2, then d←c(L) ; if i=1 and I=1, then d←C_0; else d←c(i).

Step 8 If i>=1, then repeat Steps 4-7; else goto Step 9.

Step 9 I←I-1.

Step 10 If I>=1, repeat Steps 3-9; else **P'** is obtained.

(2) Remove the effect of permutation

In this process, we obtain **P**={p(1), p(2),⋯, p(L)} from **P'**={p'(1), p'(2), ⋯, p'(L)}. All operations are the same as steps 2–10 in the permutation process except that Eq. (5) is replaced by Eq. (10), and Eq. (6) is replaced by Eq. (11), respectively.

$$p(1)\leftarrow p'(j),\ p(j)\leftarrow p'(1),\ t(1)\leftarrow j,\ t(j)\leftarrow 1. \tag{10}$$

$$p(j)\leftarrow p'(i),\ p(i)\leftarrow p'(j);\ \mathrm{temp}\leftarrow t(i), t(i)\leftarrow t(j),\ t(j)\leftarrow \mathrm{temp}. \tag{11}$$

4. Security and Efficiency Analysis

In the experiments, the images for testing are the 256×256 traditional images with 8-bit gray-scale. The secret key parameters are x_{10}=0.27, x_{20}=0.73, a_1=0.32, a_2=0.13 and s is related to the plain-image. The constants parameters are N_0 =1000, C_0=211.

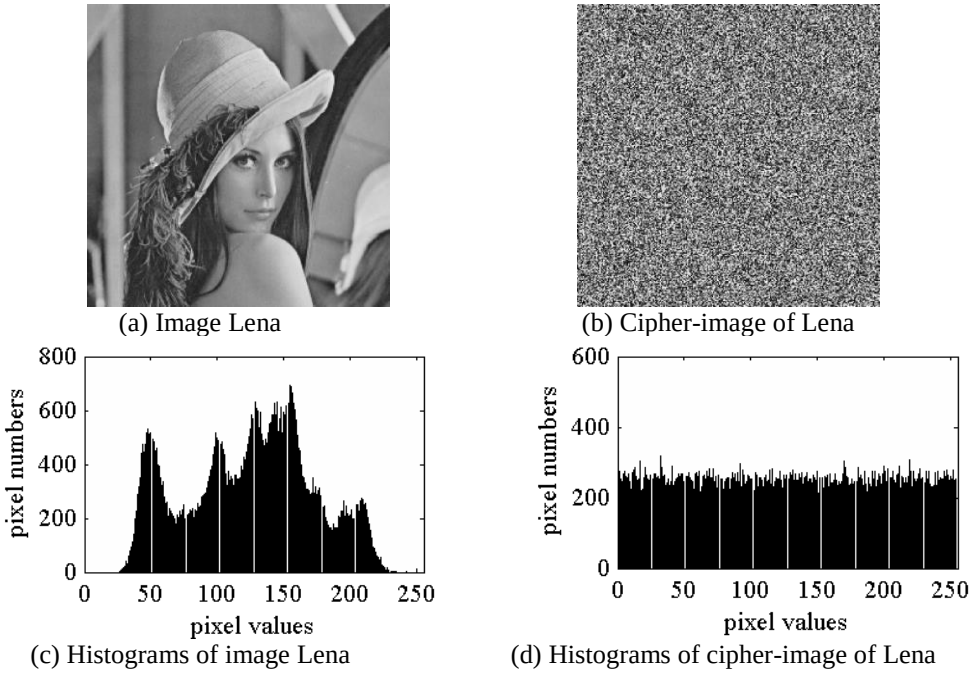
4.1. Key Space Analysis

Key space size is the total number of different keys which can be used in the encryption process. In the proposed scheme, the secret keys include {x_{10}, a_1, x_{20}, a_2, s}. They are all double-precision numbers except for s, If the computational precision of double-precision numbers is 10^{-16}, Therefore, the key space is larger than $10^{16}\times10^{16}\times10^{16}\times10^{16}=10^{64}$, which is much larger than 2^{212}. So the encryption algorithm has a large enough key space to resist all kinds of brute force attacks.

4.2. Statistical Analysis

It is well known that the statistical property of a cipher-image is enormously vital and an ideal image algorithm should be robust against any statistic attacks. Histogram, correlation of two adjacent pixels and information entropy are three important indicators of statistical analysis.

4.2.1. Histograms of Encrypted Images: The histograms of Lena and its cipher image are plotted, through which we can intuitively see the number of pixels of each value. A good image algorithm should make the histogram of cipher image as much as possible flat. Figures 3(a) and (b) are Lena image and its cipher-image; Figures 3(c) and (d) are the histograms of plain-image and cipher-image of Lena. As shown in Figure 3(d), all the gray-scale values of the cipher-image are distributed uniformly over the interval [0, 255], which is significantly different from the original distribution shown in Figure 3(c).



(a) Image Lena (b) Cipher-image of Lena

(c) Histograms of image Lena (d) Histograms of cipher-image of Lena

Figure 3. The Histograms of Lena and Its Cipher Image

4.2.2. Correlation of Adjacent Pixels: Generally speaking, two adjacent pixels of a plain image would come near to each other and a good image encryption algorithm could smash this relation between them. To test the correlation between horizontally, vertically and diagonally adjacent pixels, we calculate the correlation coefficients in each direction by

$$\mathrm{cov}(x,y)=\frac{\frac{1}{L}\sum_{i=1}^{L}(x_i-\overline{x})(y_i-\overline{y})}{\sqrt{(\frac{1}{L}\sum_{i=1}^{L}(x_i-\overline{x})^2)\cdot(\frac{1}{L}\sum_{i=1}^{L}(y_i-\overline{y})^2)}} \tag{12}$$

where $\overline{x}=\frac{1}{L}\sum_{i=1}^{L}x_i$, $\overline{y}=\frac{1}{L}\sum_{i=1}^{L}y_i$, (x_i, y_i) is the i-th pair of adjacent pixels in the same direction and L is the total number of pixel pairs. The result of the correlation coefficients along the three directions of the plain-image Lena and its corresponding cipher-image are listed in Table 1. Table 1 shows that the correlation is almost reduced to 0 after encryption, and the absolute values of the proposed method are the smallest. That is, the result is optimal in this paper compared with Ref. [9-10]. The lesser the correlation of two adjacent pixels, the safer the algorithm.

Table 1. Correlation Coefficients of the Cipher-image Lena

Algorithm	Horizontal	Vertical	Diagonal
The proposed method	0.000272	0.000735	0.002389
Ref. [9]	0.006300	0.006200	0.006900
Ref. [10]	0.00350	0.002470	0.001070

4.2.3. Information Entropy Analysis: For a message source with 2^n symbols s_i , the information entropy can be calculated by:

$$H(s) = -\sum_{i=0}^{2^n-1} P(s_i)\log_2[P(s_i)] \tag{13}$$

Where $P(s_i)$ is the probability of occurrence of the symbol s_i which is determined by the source. In the experiment, 256 gray-scale images are used, so the ideal $H(s)$ should be 8. The information entropies of cipher-images using our algorithm are shown in Table 2. From Table 2, it is known that the information entropies are close to 8, so the algorithm proposed has good property of information entropy.

Table 2. Information Entropy of the Cipher-images

Algorithm	Lena	Barboon	Pepper
The proposed method	7.9978	7.9972	7.9823
Ref. [9]	7.9973	/	/
Ref. [10]	7.9973	7.9971	7.9969

4.3. Differential Attack

If the minor modification in the plain-image or secret keys generates significant and unpredictable results in the cipher-image, the differential attack will become inefficient and useless. So, a good encryption algorithm should very sensitive to the plain-text and the secret keys. Two most common measures of sensitivity are NPCR (number of pixels change rage) and UACI (unified average changing intensity) which test the degree of difference between two images. If $c_1(i, j)$ and $c_2(i, j)$ denotes two images (i=1,2,…,M_1, j=1,2,…,M_2), NPCR and UACI can be calculated as follows:

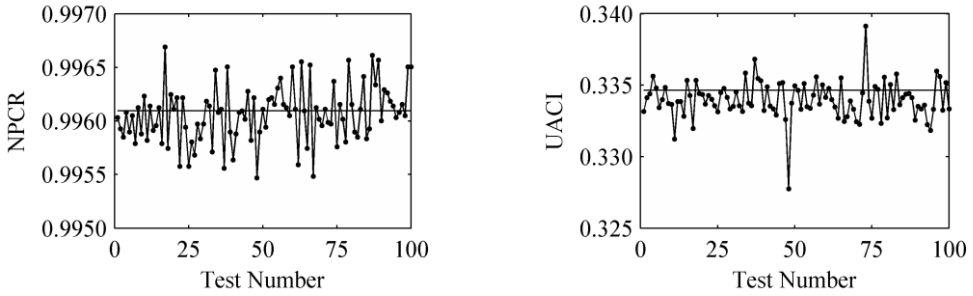
$$\mathrm{NPCR} = \frac{1}{M_1 \times M_2}\sum_{i,j} D(i, j)\times 100\% \tag{14}$$

$$\mathrm{UACI} = \frac{1}{M_1 \times M_2}\sum_{i,j} \frac{\left|c_1(i, j) - c_2(i, j)\right|}{255}\times 100\% \tag{15}$$

Where $D(i, j)$ represents the difference between $c_1(i, j)$ and $c_2(i, j)$. If $c_1(i, j)=c_2(i, j)$ then $D(i, j)=0$, otherwise $D(i, j)=1$. For a 256 gray-scale image, The expected estimates are: $\mathrm{NPCR_E}$ = 99.6094%, $\mathrm{UACI_E}$ = 33.4635%.

To test the sensitivity to plain-text, We randomly choose one pixel of the plain-image Lena and change its value by adding or subtracting 1. Then calculate NPCR and UACI between each pair of cipher images. We get 100 pairs of cipher images, and show the results of

NPCR and UACI in Figure 4(a) and Figure 4(b), respectively. It is clear that the NPCR and UACI values remain in the vicinity of the expected values (shown by the horizontal lines), i.e. the proposed image encryption technique is very sensitive to the plaintext. Ref [10] reported the mean NPCR and UACI values are 99.6041% and 33.4198%, respectively. In the proposed scheme, the mean NPCR is 99.6062% and the mean UACI is 33.3970%.



(a) NPCR for100 modified plain-images Lena (b) UACI for 100 modified plain-images Lena

Figure 4. NPCR and UACI Between 100 Pairs of Cipher Images

To test the sensitivity to the secret keys. The NPCR and UACI between two encrypted images with keys (x_{10}, a_1, x_{20}, a_2)=(0.27,0.32,0.73,0.13) and slightly varied keys (only one of the four parameters has varied 10^{-10}) are calculated, and the results are shown in Table 3. From Table 3, one can see that the NPCR and UACI are all close to the ideal values. Therefore, the encryption algorithm is very sensitive to the secret keys.

Table 3. NPCR and UACI Values with Slightly Varied Keys

Modified keys	$\Delta x_{10}=10^{-10}$	$\Delta a_1=10^{-10}$	$\Delta x_{20}=10^{-10}$	$\Delta a_2=10^{-10}$
NPCR (%)	99.5804	99.6216	99.6262	99.5926
UACI (%)	33.5021	33.4890	33.4327	33.4972

5. Conclusion

In this paper, an improved tent map with better chaotic performance is introduced. By using graphs, Lyapunov exponents and C_0 complexity of the system, the excellent performance of the improved tent map is demonstrated. Further more, based on the improved tent map, a new image encryption scheme with the improved permutation-diffusion structure is proposed. As the permutation and diffusion key sequence are all related to the plain-image, one can develop different permutation and diffusion sequences for different plain-images, which makes the scheme immune to known/chosen plaintext attack. Experimental tests demonstrate that the scheme possesses large key space, uniform distribution of cipher-images and high sensitivity to plain image and keys. So the proposed scheme has a good ability to resist brute-force attacks, statistical analysis attacks and differential attacks. With high-level security, it can be used in secure image communications.

Acknowledgements

This work was supported by Colleges and Universities Cooperation Project of Wenzhou University of China (Grant No. 2013Z005).

References

[1] C. E. Shannon, "Communication theory of secrecy systems", *Bell Syst Tech J*, vol. 28, no. 4, (**1949**), pp. 656-715.
[2] J. Fridrich, "Symmetric ciphers based on two-dimensional chaotic maps", International Journal of Bifurcation and Chaos, vol. 8, no. 6, (**1998**), pp. 1259-1284.
[3] G. Chen, Y. Mao and C. K. Chui, « A symmetric image encryption scheme based on 3D chaotic cat maps", Chaos, Solitons & Fractals, vol. 21, no. 3, (**2004**), pp. 749-61.
[4] J. Giesl, L. Behal, K. Vlcek, "Improving chaos image encryption speed", International Journal of Future Generation Communication and Networking, vol. 2, no. 3, (**2009**), pp. 23-36.
[5] S. Wang, W. Sun, Y. Guo, H. Yang and S. Jiang, "Design and analysis of fast image encryption algorithm based on multiple chaotic systems in real-time security car", International Journal of Security and Its Applications, vol. 7, no. 6, (**2013**), pp. 229-240.
[6] C. Zhu, "A novel image encryption scheme based on improved hyper-chaotic sequences", Optics Communications, vol. 285, no. 1, (**2012**), pp. 29-37.
[7] M. Francois and T. Grosges, "A new image encryption scheme based on a chaotic function", Signal Processing: Image Communication, vol. 27, no. 3, (**2012**), pp. 249-59.
[8] V. Patidar, N. Pareek and K. Sud, "Modified substitution-diffusion image cipher using chaotic standard and logistic maps", Communications in Nonlinear Science and Numerical Simulation, vol. 14, no.7, (**2010**), pp. 2755-2765.
[9] X. Wang and K. Guo, "A new image alternate encryption algorithm based on chaotic map", Nonlinear Dynamics, vol. 76, no. 4, (**2014**), pp. 1943-1950.
[10] L. Y. Zhang, X. B. Hu and Y. S. Liu, "A chaotic image encryption scheme owning temp-value feedback", Communications in Nonlinear Science and Numerical Simulation, vol. 19, no. 10, (**2014**), pp. 3653-3659.
[11] G. A. Sathishkumar, K. B hoopathy bagan and N. Sriraam, "Image encryption based on diffusion and multiple chaotic maps", International Journal of Network Security & Its Applications, vol. 3, no. 2, (**2011**), pp. 181-194.
[12] V. Patidar, N. Pareek and K. Sud, "A new substitution–diffusion based image cipher using chaotic standard and Logistic maps", Communications in Nonlinear Science and Numerical Simulation, vol. 14, no. 7, (**2009**), pp. 3056–3075.
[13] D. Arroyo, J. Diaz and F. B. Rodriguez, "Cryptanalysis of a one round chaos-based substitution permutation network", Signal Process, vol. 93, no. 5, (**2013**), pp. 1358-1364.
[14] K. Sun, S. He and C. Zhu, "Analysis of chaotic complexity characteristics based on C_0 algorithm", Acta Electronica Sinica, vol. 41, no. 9, (**2013**), pp. 1765-1771.

Author

Xuefeng Liao, she received the M.S. degree in Computer Applied Technology from Central South University, P. R. China in 2007. She has published six papers on chaotic cryptography. Her research interest includes chaotic cryptography and information security. Concurrently, she is working as a Lecturer in the Oujiang College, Wenzhou University, Wenzhou, Zhejiang, China.