An Efficient Image Encryption Scheme Based on ZUC Stream Cipher and Chaotic Logistic Map

Hai Cheng[1], Chunguang Huang[1], Qun Ding*, and Shu-Chuan Chu[2]

[1] Heilongjiang University,Key Laboratory of Electronics Engineering, China
[2] School of Computer Science,
Engineering and Mathematics Flinders University, Australia
chenghaihd@163.com,dahuangr@163.com,qunding@aliyun.com

Abstract. Digital color image encryption is different from text encryption because of some inherent features of image such as huge data capacity and high correlation among the neighboring pixels. Because of the desirable cryptographic properties of the chaotic maps such as sensitivity to initial conditions and random-like behave, more and more researches use these properties for encryption. This paper proposed an efficient image encryption scheme. Logistic chaos-based stream cipher is utilized to permute the color image. The MD5 hash function and the ZUC stream cipher algorithm are combined to diffusion the color image. Theoretical and experimental analyses both confirm the security and the validity of the proposed algorithm.

Keywords: ZUC stream cipher, logistic chaotic map, color image encryption.

1 Introduction

With the great development of network and information technology, multimedia technology and its applications especial digital images is wildly used in the computer network. It has become important and necessary to protect the information security issues against illegal copying when the color image transmitted through over the internet and wireless networks. To meet the challenge, a variety of encryption schemes have been proposed. DES, RSA, AES, RC5[1] and other popular encryption method can be used to encrypt the digital color images. Instead of using traditional block cipher for image encryption, chaotic logistic map becomes popular new days.

Many researchers have used chaotic algorithm in image encryption schemes[2-5]. Pareek[6] et al. presented a new method to encrypt the image based on chaotic logistic map. Kwok and Tang[4] presented a fast chaos-based image encryption

* This paper is supported by Innovated Team Project of 'Modern Sensing Technology' in colleges and universities of Heilongjiang Province (No. 2012TD007), Institutions of Higher Learning by the Specialized Research Fund for the Doctoral Degree (No.20132301110004) and supported by Scientific Research Fund of Heilongjiang Provincial Education Department (No.12521422).

J.-S. Pan et al. (eds.), *Intelligent Data Analysis and Its Applications, Volume 2,*
Advances in Intelligent Systems and Computing 298,
DOI: 10.1007/978-3-319-07773-4_30,

system with a stream cipher structure based on a new pseudo-random. Gao[7] proposed a new image total shuffling matrix to shuffle the position of image pixels. Then relationship between plain image and cipher image is confused. Behnia[8] et al. proposed an implementation of digital image encryption scheme based on the mixture of chaotic system.

In this paper, a digital color image encryption using then combination of ZUC stream cipher [9] and chaotic logistic map function. The logistic chaotic map is used to permute the digital color image because of sensitivity to the initial value. The MD5 hash function [10] is also used to generate the 128-bit initial vector of ZUC stream cipher because of the sensitivity to the tiny change of the image. And the ZUC stream cipher is used to diffuse the permuted image.

This paper will be arranged as follows. In Section 2 the ZUC stream cipher is proposed. The proposed image cryptosystem is mentioned in section 3. In Section 4, performance of proposed encryption method is evaluated. Finally, some concluding remarks are drawn is Section 5.

2 ZUC Stream Cipher

ZUC is a new stream cipher due for possible inclusion in the Long Term Evolution standards for mobile devices. The ZUC algorithms are the new cryptographic algorithms recommended by CCSA to be used in 3GPP LTE (Long Term Evolution). And they have been made the work item by 3GPP SA3.The ZUC algorithms have been evaluated by the algorithm standardization group ETSI SAGE, and also by two other teams of eminent experts, and are believed to be strong and suitable for LTE.

ZUC is a word-oriented stream cipher. A 128-bit initial key and a 128-bit initial vector is used as input, and a keystream of 32-bit words is generated. This keystream can be used for encryption and decryption.

The execution of ZUC has two stages: initialization stage and working stage. In the first stage, a key and IV initialization are performed. During the first stage, the cipher is clocked without producing output. During the second stage, a 32-bit word of output is produced within every clock pulse.

2.1 Algorithm Description

ZUC has 3 logical layers which is shown in Fig. 1. The top layer is a linear feedback shift register (LFSR) of 16 stages. The middle layer is for bit-reorganization (BR), and the bottom layer is a nonlinear function F.

2.2 The Linear Feedback Shift Register

The linear feedback shift register (LFSR) has 16 cells. Each cell $S_i (0 \leq i \leq 15)$ has 31 bits. And each of them is restricted to take values from the following set $\{1, 2, 3, ..., 2^{32} - 1\}$.

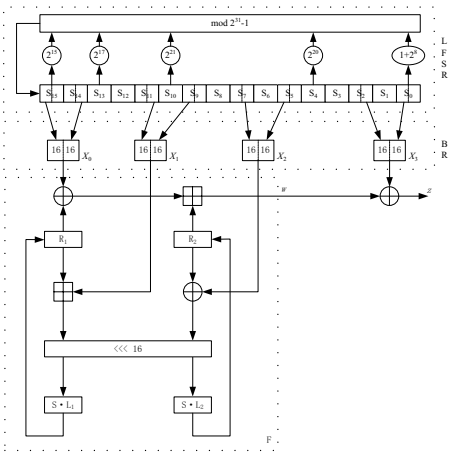

Fig. 1. General structure of ZUC

The linear feedback shift register has 2 mode of operations: the initialization mode and the working mode.

In the initialization mode, the LFSR receives a 31-bit input word u , which is obtained by removing the rightmost bit from the 32-bit output W of nonlinear function F , i.e., $u = W >> 1$. More specifically, the initialization mode works as follows:

LFSRWithInitialisationMode(u)

{

1. $v = 2^{15}S_{15} + 2^{17}S_{13} + 2^{21}S_{10} + 2^{20}S_4 + (1 + 2^8)S_0 \bmod (2^{31} - 1)$;
2. $S_{16} = (v + u) \bmod (2^{31} - 1)$;
3. $if\ \ S_{16} = 0,\ \ then\ \ set\ \ S_{16} = 2^{31} - 1\ \ ;$
4. $(S_1, S_2, ..., S_{15}, S_{16}) -> (S_0, S_1, ..., S_{14}, S_{15})$.

}

In the working mode, the LFSR does not receive any input, and it works as follows:

{

1. $S_{16} = 2^{15}S_{15} + 2^{17}S_{13} + 2^{21}S_{10} + 2^{20}S_4 + (1 + 2^8)S_0 \bmod (2^{31} - 1)$;
2. $if\ \ S_{16} = 0,\ \ then\ \ set\ \ S_{16} = 2^{31} - 1\ \ ;$
3. $(S_1, S_2, ..., S_{15}, S_{16}) -> (S_0, S_1, ..., S_{14}, S_{15})$.

}

2.3 The Bit-Reorganization

The middle layer of ZUC algorithm is the bit-reorganization (BR) procedure. It extracts 128 bits from the cell of the LFSR and forms 4 cells (X_0 ,X_1 , X_2 ,

X_3)each of which has 32-bit words. The first three words will be used by the nonlinear function F in the bottom layer, and last word will be used to generate the keystream.

The bit-reorganization forms 4 cells from the above cells as follows:

Bitreorganization()

{

1. $X_0 = S_{15H} || S_{14L}$;
2. $X_1 = S_{11L} || S_{9H}$;
3. $X_2 = S_{7L} || S_{5H}$;
4. $X_3 = S_{2L} || S_{0H}$;

}

3 The Proposed Cryptosystem

This section presents the proposed scheme for color image encryption in the framework of logistic chaotic map which is used as confusion and ZUC stream cipher which is used as diffusion. First, digital color image (P) of size $M \times N$ is converted into RGB components. Afterwards, logistic chaotic map is used to generate a chaotic shuffling sequence. And the position of pixels is shuffled by the logistic sequence. Each colors matrix (R, G, B) is converted into a vector of integers within. MD5 is used Each vector has a length of $L = M \times N$. Then, the stream cipher ZUC is used to encrypt the plaintext image.

3.1 Permutation Based on Logistic Chaotic Map

In this step, logistic chaotic map is used to shuffle the positions of the positions of the image pixels. As we known logistic chaotic map[2] is defined as follow:

$x_{n+1} = \lambda x_n(1 - x_n)$,

Where $\lambda \in (0, 4), n = 0, 1, ...$. The parameter λ and initial value x_0 may represent the key. The parameter λ can be divided into three segments.

The result shows that when $\lambda \in [3.5699465, 4)$, the characteristics of the logistic chaotic map is used to encrypt the image.

In this subsection, a position generator based on logistic chaotic map is used to shuffle the plain image pixels position. The detailed permutation process is stated as follows:

Step1: To facilitate the operation of positions path generation of an image of size $L = M \times N$, where M and N represent the width and the height of the image. The pixel of the image can be labeled by lable $(((i-1) \times M) + j)$, where $(i = 1..M$) and $(j = 1..N$).

Step2:To generate a logistic shuffling sequence, the chaotic map system parameter λ and initial value x_0 are given. A chaos sequence $[x_1, x_2, ..., x_L]$ will be obtained by iterating logistic equation.

Step3:Sorting this set from smallest to largest, a new set $[\overline{x}_1, \overline{x}_2, ..., \overline{x}_L]$ will be get. To obtain the position path, an array of length L is created and each element of this array takes its index array.

Step4:The original position of $\overline{x}_i$ in $[x_1, x_2, ..., x_L]$ can be found, the shuffling sequence is $MD = [m_1, m_2, ..., m_L]$ can be generated. When the shuffling sequence is generated, the pixel position of digital color image can be reordered according to $MD = [m_1, m_2, ..., m_L]$.

Here, we randomly set λ =3.66 and initial value x_0 =0.9. Fig. 2(a) is the original image of Lena, and Fig. 2(b) is encrypted image of Lena shuffled by logistic chaotic map. Fig. 3(a) is the decrypted image of Lena and the Fig. 3(b) is the decrypted image of Lena with fault parameter.

Fig. 2. (a) Original image of Lena (b) Encrypted Image of Lena shuffled by logistic chaotic map

3.2 Diffusion Based MD5 and ZUC Algorithm

Permutation based on logistic chaotic map is to shuffle the pixel of image which is shown in Fig. 5. The histograms of shuffled image is same to the original image. To overcome those limitations, this paper intends to propose MD5 hash and ZUC stream cipher to diffuse the digital color image.

The Message-Digest algorithm 5 (MD5) is wild used cryptographic function with a 128-bit hash value. This algorithm compresses packet message input with any length into a fixed 128-bit value. If one pixel of image changed, the result of output will also make great adjustment. Thus, it is difficult to reverse the original image depending on summary value. The detailed permutation process is stated as follows:

Step1: Use image which permuted based on logistic chaotic map described above go generate a 128-bit MD5 hash value $MD(i), (i = 1, 2, ..., 128)$ as the 128-bit initial vector iv;

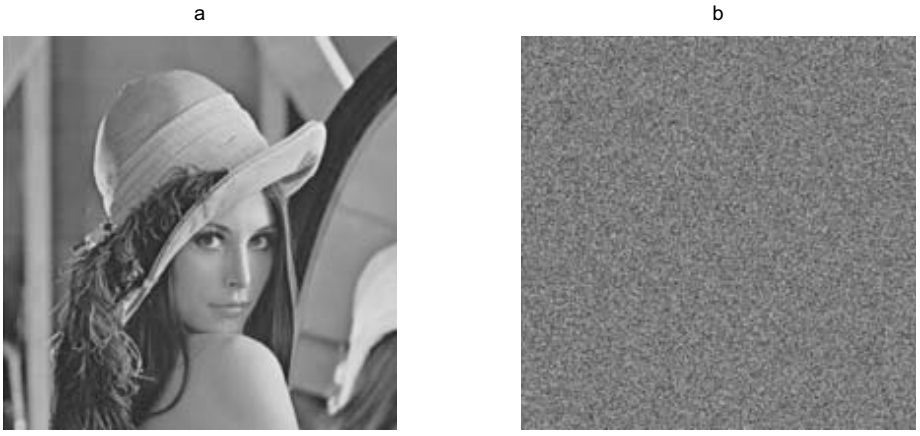

Fig. 3. (a)Decrypted image of Lena (b) Decrypted image of Lena with fault perameter

Step2: During the initialization of ZUC algorithm, the $MD(i)$, $(i = 1, 2, ..., 128)$ and 128-bit initial key k are called by the algorithm. After the initialization stage, the algorithm moves into the working stage. Then the algorithm goes into the stage of producing keystream. For each iteration, a 32-bit word Z_i is produced as an output.

Step3: Reshape the sequence $Z = \{Z_1, Z_2, ..., Z_{M\times N}\}$ to two-dimensional value matrix as shown in the formula 1. Assume a consecutive sequence of plain image pixels P_{ij} as shown in the formula 2 where i $(i = 1..M)$ and j $(j = 1..N)$ donate the location of digital color image. Then the diffused image P' is generated as shown in the formula 3 where $\oplus$ represents the exclusive XOR operation bit by bit.

$$Z = \begin{bmatrix} Z_1 & Z_2 \ ... & Z_M \\ ... & ... \ ... & ... \\ ... & ... \ ... & ... \\ Z_{M\times(N-1)+1} & ... \ ... & Z_{M\times N} \end{bmatrix} \tag{1}$$

$$P = \begin{bmatrix} P_{11} & P_{12} \ ... & P_{1N} \\ ... & ... \ ... & ... \\ ... & ... \ ... & ... \\ P_{M1} & ... \ ... & P_{MN} \end{bmatrix} \tag{2}$$

$$P' = Z \oplus P = \begin{bmatrix} Z_1 & Z_2 \ ... & Z_M \\ ... & ... \ ... & ... \\ ... & ... \ ... & ... \\ Z_{M\times(N-1)+1} & ... \ ... & Z_{M\times N} \end{bmatrix} \oplus \begin{bmatrix} P_{11} & P_{12} \ ... & P_{N1} \\ ... & ... \ ... & ... \\ ... & ... \ ... & ... \\ P_{M1} & ... \ ... & P_{MN} \end{bmatrix} \tag{3}$$

Fig. 4. (a) is original image, and (b) is the encryption image

3.3 The Decryption Process

In the decryption procedure is similar in reverse way to that of encryption process which is mentioned above. Note that the receiver must have the same keystream to be able to decrypt the color image.

4 Performance and Security Analysis

A good encryption procedure should be robust against all kinds of cryptanalytic, statistical and brute-force attack. In this section, some experiments has been conducted to evaluate the performance of proposed encryption algorithm. These experiments include encryption and decryption process, histogram analysis of plain-image and cipher-image. In this section, a 512 x 512 digital color LenaRGB.bmp is used. Experiments are carried out and the data are analyzed using MATLAB.

4.1 Key Space Analysis

The initial key of ZUC and in addition to the seed of logistic chaos map are considered the key of this proposed cipher. As we known that a good encryption algorithm should have large key space to prevent brute-force attacks which is defined to exhaust all the possible key until the correct one. Then output of encryption system should be sensitive to the initial cipher keys. In the proposed algorithm, key space analysis and testing have been carefully performed and completely carried out.

The 128-bit initial key k and the 128-bit initial vector iv which is generated by the MD5 hash function of the ZUC stream cipher is needed. So this algorithm is a 256-bit encryption scheme, with the key space size $2^{256} \approx 1.1579 \times 10^{77}$.

Moreover, when a key is used to encrypt an image, another modified key is used to decrypt the ciphered image, the decryption is also completely fails.

4.2 Histogram Analysis

Image histogram depicts statistical distribution of color intensities. Encrypted color image can be characterized by flat histograms for all colors in which the intensities are distributed evenly over the whole color scale. Fig. 5 shows the histograms of RGB colors for the original image of Lena and the encrypted image respectively. The figures show clearly the uniformity and random-like appearance of the histogram in the ciphered image.

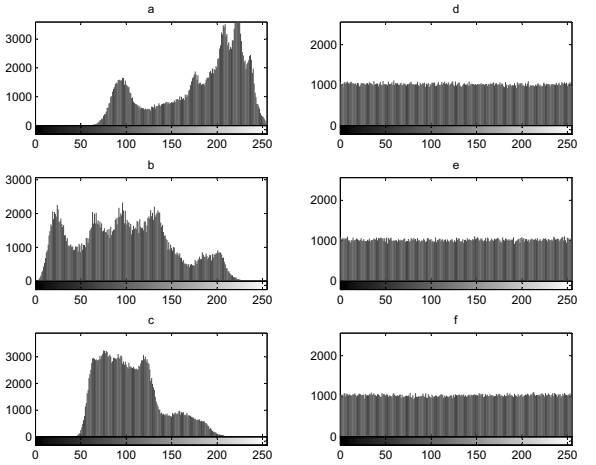

Fig. 5. Histogram of image, a,b,c is histograms of RGB components for the original image. d,e,f is histograms of RGB components for the ciphered the image.

4.3 Pixel Correlation Analysis

It is well known that adjacent image pixel are highly correlated either in horizontal, vertical and diagonal directions. An effective encryption algorithm should make the correlation between adjacent pixels in the images as minimally as possible.

The correlation coefficient between two adjacent pixels x_i and y_i can be calculated[11] between two vertically adjacent pixels, two horizontally adjacent pixels and two diagonally adjacent pixels respectively. Where x and denote two adjacent pixels and N is the total number of the pixels from the image for the calculation.

Table 1 summarizes the auto correlation coefficients for horizontal, vertical and diagonal orientations of the original and ciphered image.

Table 1. Correlation coefficients of two adjacent pixels

	original image			ciphered image		
	red	green	blue	red	green	blue
Horizontal	0.9798	0.9691	0.9327	-0.0022	0.0006	0.0005
Vertical	0.9893	0.9825	0.9576	0.0010	0.0018	0.0030
Diagonal	0.9697	0.9555	0.9183	0.0031	0.0017	0.0022

4.4 Differential Analysis

In general the opponent may make a slight change even one pixel of the encrypted image to observe the change of the result. In this way, the relationship between the plain-image and the cipher-image can be found. If one minor change in the plain-image can cause a significant change in the cipher-image, then differential attack would become very inefficient.

Such difference can be measured by means of two criteria namely, the number of pixel change rate (NPCR) and the unified average changing intensity(UACI). (reference)

Table 2 depicts the mean values of the NPCR and UACI tests for the image of Lena.

$$D(i,j) = \begin{cases} 0\ , & C_1(i,j) = C_2(i,j) \\ 1\ , & C_1(i,j) \neq C_2(i,j) \end{cases}$$

$$NPCR = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{i=1}^{N} D(i,j)$$

$$UACI = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{i=1}^{N} \left(\frac{|C_1(i,j) - C_2(i,j)|}{255}\right) \times 100$$

Table 2. Differential analysis result for the image of Lena

	Expected value(%)	proposed scheme(%)
NPCR	99.60937	99.611
UACI	33.46354	33.515

5 Conclusion

This paper has proposed a novel image encryption algorithm which uses a logistic chaotic map to confuse the image. The MD5 hash function and ZUC stream cipher to diffuse the image. This encryption system has enhance the cryptosystem security. And security analyses such as key space analysis, histogram analysis, pixel

correlation analysis and differential analysis have been conducted for several image to prove the security of the image encryption system. From the result, this technique outperforms other encryption techniques and can be used for real-time image encryption, real-time video encryption and other transmission applications.

References

1. Ahmed, H.E.H., Kalash, H.M., Allah, O.S.F.: Encryption quality analysis of the RC5 block cipher algorithm for digital images. Opt. Eng. 45, 107003-107003-7 (2006)
2. Barakat, M.L., Mansingka, A.S., Radwan, A.G., et al.: Hardware stream cipher with controllable chaos generator for colour image encryption. IET Image Proc. 8, 33–43 (2014)
3. Behnia, S., Akhshani, A., Mahmodi, H., et al.: A novel algorithm for image encryption based on mixture of chaotic maps. Chaos, Soliton. Fract. 35, 408–419 (2008)
4. Gao, T., Chen, Z.: Image encryption based on a new total shuffling algorithm. Chaos, Soliton. Fract. 38, 213–220 (2008)
5. Kwok, H.S., Tang, W.K.S.: A fast image encryption system based on chaotic maps with finite precision representation. Chaos, Soliton. Fract. 32, 1518–1529 (2007)
6. Matthews, R.: On the derivation of a chaotic encryption algorithm. Cryptologia 13, 29–42 (1989)
7. Mazloom, S., Eftekhari-Moghadam, A.M.: Color image encryption based on Coupled Nonlinear Chaotic Map. Chaos, Soliton. Fract. 42, 1745–1754 (2009)
8. Pareek, N.K., Patidar, V., Sud, K.K.: Image encryption using chaotic logistic map. Image and Vision Comput. 24, 926–934 (2006)
9. Kitsos, P., Sklavos, N., Provelengios, G., et al.: FPGA-based performance analysis of stream ciphers ZUC, Snow3g, Grain V1, Mickey V2, Trivium and E0. Microprocess. Microsy. 37, 235–245 (2013)
10. Zhu, H., Zhao, C., Zhang, X.: A novel image encryption compression scheme using hyper-chaos and Chinese remainder theorem. Signal Processing-Image 28, 670–680 (2013)
11. Liu, H., Wang, X.: Color image encryption based on one-time keys and robust chaotic maps. Compu. Math. Appl. 59, 320–332 (2010)