Send Orders for Reprints to reprints@benthamscience.ae

The Open Automation and Control Systems Journal, 2015, *7*, 259-265

Analysis and Improvement of a Self-Adaptive Algorithm on Image Encryption

Li Dongmei[1], Wang Aiju[2] and Yang Xinfeng[1,*]

[1]*School of Computer & Information Engineering, Nanyang Institute of Technology, Henan, Nanyang, 473000, P.R. China*

[2]*Information Engineering Institute, Zhengzhou Institute of Engineering Technology, Henan, Zhengzhou, 450044, P.R. China*

Abstract: With the research on the digital image encryption, one purpose of this paper is to propose an improved algorithm which compensates for the security flaws found in the traditional Self-adaptive Algorithm. The other is to develop an image encryption system based on the improved algorithm. Improved Self-adaptive Algorithm on image encryption can effectively resist the chosen-plaintext attack, cover the probability distribution of the pixels in the plaintext image and losslessly recover the plaintext image.

Keywords: Image Encryption, Cryptography, Self-adaptive, Algorithm.

1. INTRODUCTION

With the rapid development of science and technology, the ways of communication have become more and more diverse, and the media that can communicate are increasing. The Internet has become an important tool for people to communicate with each other. In recent years, the multimedia data, such as voice, image, video and so on, have become the mainstream carrier of the network transmission data because of its richer and more vivid information than words. Since it involves the transmission of information, we have to deal with the important issue of information security. Due to the different data structure of various information carriers, there are different research methods for security strategy.

The digital image encryption is a combination of digital image processing technology and information security technology. It is also one of the conventional means to ensure the safe transmission of digital images in the network. By encrypting the digital image to be transmitted, the information contained in it can be hidden and protected, and the information can't be tampered with or acquired by illegal users, and only legitimate users can use the correct key to decrypt and restore the original image, so as to effectively ensure the safe transmission of digital images in the network.

Unlike general text secondary data, the digital image has the spatial order. By changing the orderliness, a fast digital image encryption algorithm can be designed. However, this scrambling algorithm, which only changes the image pixel position without changing the image pixel value, is usually not effective against known plaintext attacks. The attacker uses known plaintext attacks, which requires mastering the encryption algorithm, intercepting some ciphertext, and also knowing one or more plaintext ciphertext pairs; According to these information, the attacker may obtain the key from the known plaintext to the ciphertext, so as to achieve the purpose of breaking the plaintext. The known plaintext attack is one of four types of attacks on cryptographic system. According to the amount of information obtained by attackers from less to more, the four types of attacks are: ciphertext only attack, known plaintext attack, chosen plaintext attack, chosen ciphertext attack [1].

In 2005, Chen Gang proposed a different self-adaption image encryption algorithm [2], which could effectively resist the known plaintext attacks by using the original image data itself to control the arrangement of image pixels. The literature [3] on adaptive image encryption algorithm was analyzed and improved, the adaptive image encryption algorithm could not effectively resist the selection of plaintext attacks, and its improved algorithm was proposed. The improved algorithm had better performance in security and encryption speed. The idea of adaptive image encryption to control the arrangement of image pixels by means of the original image data itself was used in the literature [4]. In the process of encryption, the propagation of analog wave was used, and a new digital image encryption algorithm based on wave propagation was proposed.

2. ANALYSIS AND IMPROVEMENT OF ADAPTIVE IMAGE ENCRYPTION ALGORITHM

The permutation process of self-adaption image encryption algorithm is not only related to the key, but also related to the plaintext. This also overcomes the shortcomings of the traditional image permutation algorithm, which can effectively resist the known plaintext attack. However, when the

*Address correspondence to this author at the School of Computer and Information Engineering, Nanyang Institute of Technology, Nanyang, 473004, P.R. China; Tel: 13803770071; E-mail: ywind2005@163.com

1874-4443/15

attacker has more information, he can choose the plaintext message and obtain the corresponding ciphertext generated by the adaptive image encryption algorithm. The attacker can select plaintext attacks. Through the analysis of security defects in document [5], it is found that the following two security defects exist in the adaptive image encryption algorithm.

Security defect 1: the adaptive image encryption algorithm cannot effectively resist the chosen plaintext attack. When the length of the key is L, the key of the algorithm is completely restored by using the specific choice of plaintext ciphertext pair [6]. The main attack ideas are as follows: Set a special plaintext image so that the image is not changed after being encrypted by the previous L-1 bit key, and after the last key is encrypted, it can be deduced whether the last key is 0 or 1.This step is then repeated to obtain the previous L-1 bit key.

Security defect 2: the probability distribution of the pixel gray value of the ciphertext image is not changed compared to the plaintext image. Although the arrangement process of the image encryption algorithm is related to the key and the plaintext, the location of the image pixels is still only replaced, and the statistical information of the pixel values of the image is not changed; it can also be said that the histogram of the image before and after encryption is unchanged. Therefore, the probability distribution of the pixel gray value of the plaintext image is still exposed to the attacker [7].

In view of the two security defects mentioned above, some improvement methods are put forward to solve the defects.

Improvement method 1: The Hash Function is used to obtain the Hash value of the plaintext image, and the Hash value is transformed through the format to generate the required fixed length keys [8]. The implementation method is as follows: the plaintext image is read using a stream file, and then the MD5 hash function is used to calculate the hash value of the plaintext image. The hash value is usually a hex string sequence of 32 bits, and the key requires a binary string sequence with a length of at least 128 bits. After the conversion, the hash value is converted to the binary number of 128-bit, which is exactly the length of our minimum requirement.

Improvement method 2: Using the 32-bit hexadecimal hash value that has been obtained, repeated cycle operations are used to perform XOR operations on the top four bit planes of the ciphertext image.

Conclusion: Combined with the above two improvement measures, the improved adaptive image encryption algorithm has become:

Step 1: Read the plaintext image, use MD5 hash function to calculate its hash value, and through the conversion to get the length of 128 secondary key. Let i = 0;

The Step 2 and Step 3 are the same as the traditional adaptive image encryption algorithm; since the Step 4 is introduced, in the Step 3, the algorithm is terminated and the change is entered the Step 4.

Step 4: Using the 32-bit hexadecimal hash value that has been obtained, repeated cycle operations are used to perform XOR operations on the top four bit planes of the ciphertext image.

In this paper, the improved adaptive image encryption algorithm is adopted to merge sort. The merge sort [9] is a stable sorting algorithm with the order of the average sort time, the worst case sort time and the order of the keywords being O (n logn), where n is the number of image pixels to be sorted. In each round of encryption process, because the first half of the image is arranged on the other half, then the order is then arranged again, so twice merge sort is applied in each round of encryption.

3. DEVELOPMENT NEEDS AND DESIGN AN ADAPTIVE IMAGE ENCRYPTION SYSTEM

3.1. Development Needs Adaptive Image Encryption System

The image encryption system not only needs to process the plaintext image to realize the function of its system encryption, but also needs the convenience of the system according to the user's specific use requirements. In the image encryption system, the validity of the authentication name and password should be realized in the process of user land, and the password can be easily changed after land.

The adaptive image encryption system should implement the following functions and objectives:

(1) The user land the system through the specified user name and the corresponding user password. It is necessary to verify the validity and the standardization of the user name and user password. The wrong user name or user password is warned. After the successful landing, the landing time and the last landing time are recorded, and there is a corresponding message.

(2) After the successful user login, you can modify the user password. In the process of modification, the validity of the old password and the standardization of the new setting password should be verified. The old password authentication information and non-standard password settings should have a corresponding warning. In the system, the password should be composed of 4-16 bits letters and Numbers.

(3) The user can autonomously select the plaintext image to be encrypted, and use the improved adaptive image encryption algorithm to encrypt it.

(4) The user can decrypt the encrypted ciphertext image, and ensure that the encrypted ciphertext image can decrypt the information represented by the plaintext image by decrypting. The unencrypted image cannot be decrypted. It is necessary to ensure the logic of encryption and decryption functions.

(5) The encrypted cipher image can be preserved by an adaptive image encryption system.

(6)The decrypted cipher image can be preserved by an adaptive image encryption system.

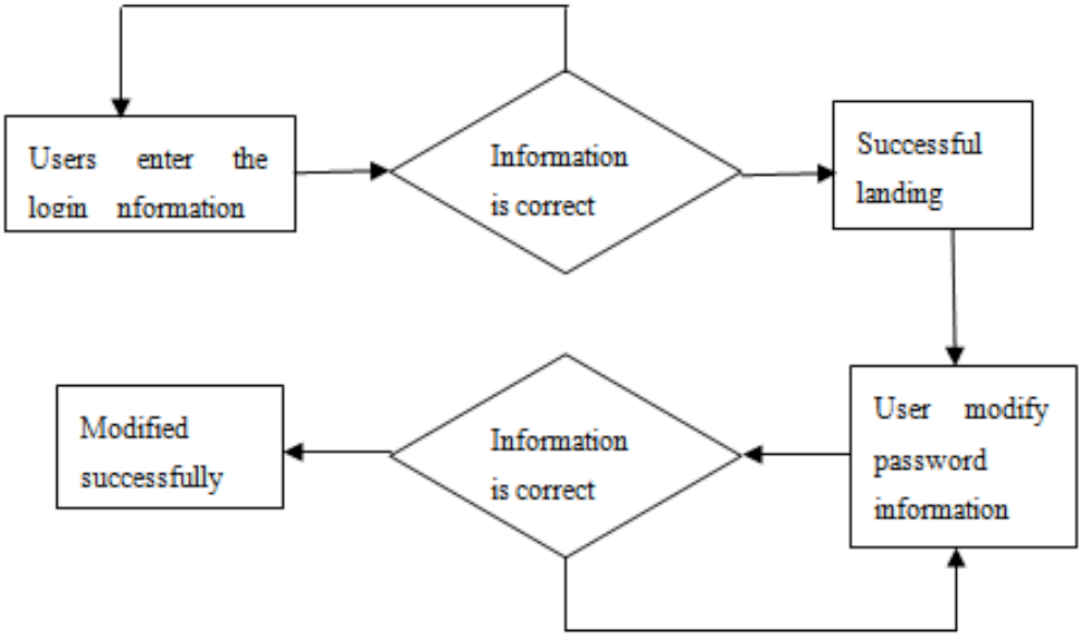

Fig. (1). User Login Information Related Function Flow Chart.

3.2. Structural Design Adaptive Image Encryption System

The system is an improved adaptive image encryption system, which will be used in the user's personal computer. It belongs to the category of application development of the form, and its used environment does not involve the use of the network, regardless of the communication between the client and the server. The system is an adaptive image encryption system, so the system architecture can be expressed as the following three layers:

(1) Interactive layer: In this system, the interactive layer is the graphical user interface, referred to as GUI (Graphical User Interface)[10]. It mainly includes three graphical user interfaces: user login interface, user password modification interface, image encryption and decryption interface. Through related system controls such as Button, Textbox, PictureBox, etc. to realize the development of the system's various graphical user interfaces.

(2) Core layer: In this system, the core layer is hidden in the graphical user interface by the program code to achieve a variety of functions of the sub-module, and which contains a variety of related algorithms and functions, such as the system used in the merge sort algorithm, MD5 hash Function and so on. Its function sub-module mainly includes landing information verification, password verification and the most important image encryption and image decryption function.

(3) Data layer: In this system, the use of the data mainly includes the table in the database and image files stored in the computer hard disk .The data table is connected to the database through the system to read the required information; and the image file is read to the system through the FileSteam class in the system, so that the use of image pixel information in the functional module is convenient.

3.3. Functional Processes of Adaptive Image Encryption System

In this system, the functional process is roughly divided into two parts. One part is the function flow related to the user's personal information, the other is the function flow of the user's encryption and decryption. The two processes are described below.

(1) User personal information related functional processes:

First, the user needs to enter the login information, including the user name and user password. If the login information is correct, the login will be displayed successfully. The system logs the current login time and displays the last login time. Then, the user enters the change password information, including the old password and the new password twice, if the modified information is correct, the display changes successfully. The specific flow diagram is shown in Fig. (**1**).

(2) Function flow of encryption and decryption:

First, the user selects the plaintext image to be encrypted by accessing the hard disk directory, then the system displays a successful plaintext image. When the user clicks the encryption button, the system encrypts the encrypted image with the improved adaptive image encryption algorithm and displays the encrypted ciphertext image. When the user clicks the decryption button, the system decrypts the decrypted image with the reverse operation of the improved adaptive image encryption algorithm and shows the decrypted image after decryption. The specific flow diagram is shown in Fig. (**2**).

In the implementation of encryption, four sub functions are involved: the realization of merge sorting algorithm, the realization of image pixel value and gray value interchange, the realization of XOR operation and the realization of image

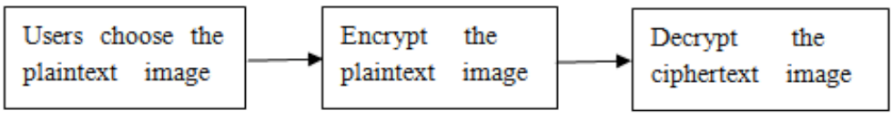

Fig. (2). Functional Process About Using the System of Encryption and Decryption

gray value replacement according to ordered two-dimensional array. The four sub functions are described below.

(1) Implementation of merge sorting algorithm

The merge sort algorithm is mainly realized by the following steps.

1) The array P to be sorted is divided into two equal parts array P1 and array P2;

2) The array P1 and the array P2 are sorted by the merge algorithm respectively to form the ordered array P1 'and the array P2'.

3) The values in the two ordered arrays P1 'and P2' are compared one by one to obtain a complete ordered sequence P '.

4. REALIZATION OF THE SYSTEM

The whole realization step reflects the important idea of recursion. In the concrete implementation of the system, the merge sort algorithm is encapsulated into a class named Merger, so that the algorithm can be called in the encryption and decryption process.

(2) Realization of image pixel value and gray value interchange

The pixel value of the image is different from the gray value of the image. For the color image, the value of a pixel is actually represented by the data structure of the array, which records the four components of the pixel's ARGB. Where A represents the alpha channel and RGB represents the red, green and blue colors. The grayscale value is the average of three values of RGB. For grayscale, the pixel values are the same as the gray values.

In practice, the gray scale is also stored in accordance with the data structure of the color image. In the image of "Bmp", the image pixel values are read into the system and stored in the color type. And this type is actually an array. The type is converted to an integer type that stores only one gray value, and the gray value corresponding to the pixel can be correctly represented. It can be implemented by formula 1:

color =Bitmap GetPixel (w, h);

value[h,w]=(color.R +color.G +color.B)/3. (1)

In Formula 1, w represents the width value of the image in which the current pixel is located, and H represents the height of the image in which the current pixel is located. GetPixel (w, h) represents the pixel value of the current pixel Color type, and value [h, w] represents the gray value corresponding to that pixel.

According to the principle of pixel value and gray value change, the pixel value represented by Color type can be converted to the gray value of integer type by formula 2.

P=value[h,w]

Bitmap .Set.Pixel (w ,h ,Color. FromArgb (p, p, p)); (2)

(3) Implementation of XOR operation

The XOR operation belongs to logic operation and has the following characteristics: a⊕b⊕b =a。Among them, ⊕ represents XOR operation. This feature allows the ciphertext to be recovered by XOR after the XOR has been processed.

Suppose there are multidimensional arrays:

value[h, w] (h =0,1,L ,Height -1;w =0,1,…,Width-1)

Where Width represents the width of the image, and Height indicates the height of the image. Value[, h, and w] represents grayscale values of pixels high in H and W in width.

In the implementation of the XOR operation, the 128-bit binary key Key [i] (i = 0,1, L127) is converted to a 32-bit hexadecimal hash value Key '[j] = (J = 0,1, L31); Then the grayscale value of the image pixel is used to perform the following operation: Let low be the lower four bits of value [h, w], high for the high four bits of value [h, w].Then we use formula 3 to make the high and the hash value Key '[j] XOR to get high'.

high =high⊕Key [j] (3)

In the following equation, the image pixel value after XOR operation is obtained by Equation 4.

value [h ,w]=high <<4+low (4)

(4)Realization of image grayscale value replacement based on ordered two-dimensional array

Suppose there are two-dimensional arrays:

temp[i, j] (i =0,1;j =0,1,…,n -1;n =Height ×Width/2)

Temp[0, j] stores the gray values of half images. After merging and sorting, the ordered sequence is arranged from small to large, and temp[1 and j] indicates whether the gray value is accessed. When temp [1, j] = 0, indicating that the gray value is not accessed, when temp [1, j] = 1, indicating that the gray value has been accessed.

Suppose there are multidimensional arrays:

value[h, w] (h =0,1,L ,Height -1;w =0,1,…,Width-1)

The element value [h, w] in the array represents the gray value of the pixel with height h and width w in the image.

The following example illustrates how to implement a replacement. In the two-dimensional array, temp [0, j] stores the gray value of the lower half of the image after it is merged; and all grayscale values are not accessed: temp[1,j] =0(j =0,1,…,n-1).Now, according to this two-dimensional array, the gray value of the upper part of the image is replaced. Traverse the lower half of the corresponding image

Fig. (3). Expressly image lena.bmp.

in the multidimensional array, whichever is the gray value [h, w].

When value [h, w] = temp [0, j], and temp [1, j] = 0,

Let x = [j / Width], y = j mod Width;

Description: In a two-dimensional array, the gray part of the pixel in the lower half of the image should be in the position (x, y) after sorting. But what you want to replace is the gray value of the upper half of the image. The column value is constant, and the row value is less than half the height of the image, and the position of the pixel in the upper half of the image corresponding to the lower half of the image can be determined. Therefore, the substitution is achieved by formula 5.

$$value[x, y] = value[h - Height/2, w] \quad (5)$$

According to the improved algorithm of adaptive image encryption algorithm, the encryption is implemented step by step. While reading the plaintext image to be encrypted, the 128-bit binary key after the MD5 hash has been calculated. The 128 bit binary string stored is a string-type. Each character of the string is accessed by each function, and through the character comparison to determine whether the current key value is 1 or 0.Next, the implementation of step 2 of the encryption algorithm is entered. Since the whole step 2 is half the image encrypted through the image, there is a certain similarity between the upper and lower sides. Therefore, only one case will be introduced, and the rest can be treated by analogy. Here we introduce the process of encrypting the upper half of the image from the bottom half of the image.

First, the pixel value of the entire plaintext image is converted to a gray scale value and stored with a Height × Width multidimensional array. Where Height represents the width of the image and Width represents the height of the image. Then, the gray value of the lower part of the image is passed through the loop, which forms a two-dimensional array. This two-dimensional array represents the grayscale values of the lower half of the image and the conditions of its access. Using the merge sort algorithm, the gray value of the two-dimensional array is sorted. Finally, according to the order of the sorted two-dimensional array, the pixels of the upper part of the image are replaced. After 128 rounds of encryption, we can enter the implementation of step 4.The procedure is to perform XOR operations on the high bit planes of the image. So far, the system encryption process is over.

5. EXPERIMENTAL EXAMPLES

Under different experimental circumstances, the same experimental data and experimental system show that the experimental results will be affected by environmental factors. It is therefore necessary to provide a simple description of the experimental environment. All of the experiments in this paper are carried out in the following experimental environments:

Operating system: Windows7SP1 32-bit;

CPU：T5600 1.83GHz；

RAM：2GB；

Graphics card:GeForce Go 7300；As shown in Fig. (**3**).

5.1. Experimental Examples

Experiment 1: Test the sensitivity of the MD5 generation key to the plaintext image.

Experimental process:

In this experiment, we first change the gray value of the last pixel in the subject "lena.bmp", only incrementing the gray value by 1 and saving it as "lenaEndAdd1.bmp".Then the adaptive image encryption system on the experimental object "lena.bmp" and "lenaEndAdd1.bmp" with the MD5 hash function is calculated for each of the 32 sixteen hexadecimal hash values.

Experimental results:

A=d41d 8cd9 9f00 b204 e980 0998 ecf8 427e；

B=b25c 09bb 916e da9e f8dd b839 aa6c c562.

Where A is the MD5 hash value for "lena.bmp"; B is the MD5 hash value for "lenaEndAdd1.bmp".Since encryption is controlled by each binary bit of the key, the A and B are converted into 128 bit binary numbers. A different percentage of 44.5% is found by comparison.

Experiment 2: Half-round encryption and decryption

Experimental process:

Fig. (4). The Half-Round Encrypted and Decrypted Image lenaEnH.bmp&lenaDeH.bmp.

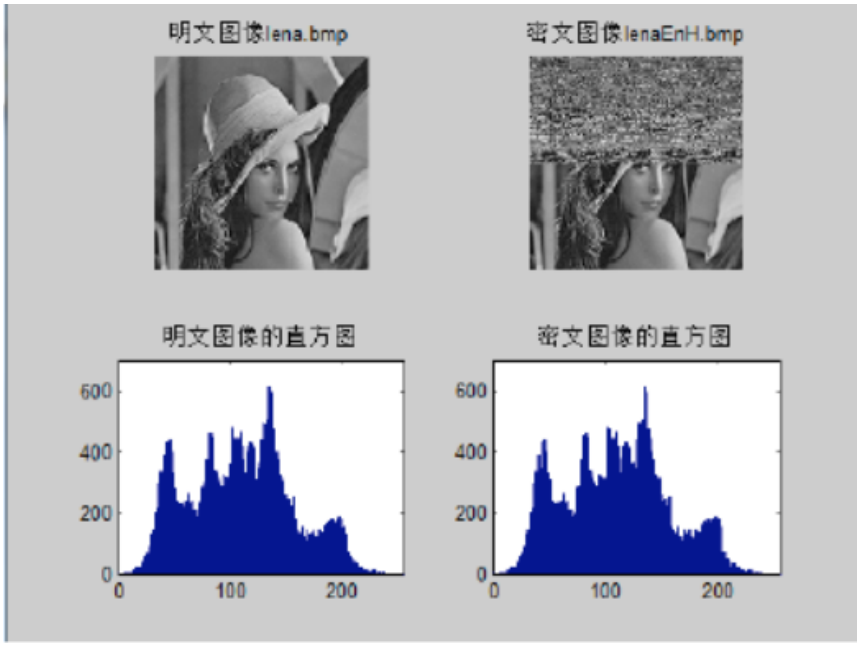

Fig. (5). The Contrast of the Histogram About Plaintext Image and the Encrypted Image in Experiment 2.

In this experiment, the image is encrypted and decrypted by the adaptive image encryption system. That is, the system allows the lower half of the image to encrypt the upper part of the image, get the ciphertext image; and then decrypt the ciphertext to get the decrypted image. It should be noted that in the half-round encryption and decryption process, there is no use of plain text image MD5 hash value to achieve XOR operation.

Experimental results:

The ciphertext after half rounds is called "lenaEnH.bmp";The decrypted image after half rounds of decryption is called "lenaDeH.bmp".The specific image is shown in Fig. (**4**).

5.2. Experimental Analysis and Conclusion

To analyze the experimental results of experiment 1, we found that even if the difference is only one pixel value of the two images, the use of MD5 hash function to calculate the hash value will be very different. This indicates that the process of generating the key has a certain sensitivity to the plaintext image.

To analyze the experimental results of experiment 2, we see that after the half-frame encryption of the plaintext image, the upper part of the obtained ciphertext image has been obviously confused. The upper part of the image information has been well hidden, and the encryption of the upper part of the plaintext image is realized. After decrypting the cube image, the resulting decrypted image has no visual difference from the plaintext image. It is found that the pixel values of the decrypted image and the plaintext image are exactly the same, by comparing the specific pixel values of the decrypted image and the plaintext image by the related software. In addition, the histogram of the plaintext image and the encrypted image (see Fig. **5**) is analyzed and found to be the same.

6. SUMMARY AND OUTLOOK

In this paper, the digital image encryption algorithm is studied. We analyze and improve the traditional adaptive image encryption algorithm, and develop an improved adaptive image encryption application system.

First of all, the existing digital image encryption algorithm is described for the digital image security problem. In this paper, the adaptive image encryption algorithm is analyzed. In view of the existing security flaws, the improved image encryption algorithm and the improvement steps are proposed, and the improved adaptive image encryption algorithm is also analyzed.

According to the requirements of adaptive image encryption system in daily use and the basic requirements of application system development, the design goal and design scheme of adaptive image encryption system are put forward. Then, in the process of actual development environment, the algorithms and development techniques involved in the process of adaptive image encryption are introduced. Finally, the development tools are used to realize the interface and function of the system.

Finally, the design of the adaptive image encryption system is completed. According to the experimental results and the information contained in the experimental results, it is proved that the improved adaptive image encryption algorithm makes up the main defects of the traditional adaptive image encryption algorithm. The improvement of the adaptive image encryption algorithm is supported by the experimental data.

In the research aspect, although the improvement of traditional adaptive image encryption algorithm is proposed, the core strategy of the algorithm is not fundamentally changed. The improved adaptive image encryption algorithm still has some shortcomings and mainly reflected in the encryption speed. The improved adaptive image encryption algorithm still requires 128 rounds of encryption. Although the time of single round encryption is short, it still takes time to complete the 128 rounds of encryption. A plaintext image of 128×128, which is processed by an improved adaptive image encryption algorithm takes about five minutes. In addition, in the process of analyzing the improved adaptive encryption algorithm, its reference object is the traditional adaptive image encryption algorithm, which is not compared with other image encryption algorithms, such as the image encryption algorithm based on neural network, the image encryption algorithm based on the Feistel network structure [11] and so on. In the next key research work, we can start from the research and analysis of the other new image encryption algorithm to start, and learn from its core ideas, which can not only guarantee the encryption effect, but also reduce the encryption time of the new algorithm. A new algorithm which can guarantee the encryption effect and reduce the encryption time will be proposed.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] C.E. Shannon, "Communication theory of secrecy systems", *Bell Systems Technical Journal*, vol. 28, pp. 656-715, 2000.

[2] Chen Gang, Zhao Xiaoyu, Li Junli,"An Adaptive Image Encryption Algorithm", *Journal of Software*, vol. 16, pp. 1975-1982, 2005.

[3] S.S. Maniccam, and N.G. Bourbakis, "Lossless image compression and encryption using SCAN", *Pattern Recognition*, vol. 34, pp. 1229-1245, 2001.

[4] I. Parbeny, "An efficient algorithm for the Knight's tour problem", *Discrete Applied Mathematic*s, vol. 73, pp. 251-260, 2007.

[5] S.S. Maniccam, and N.G. Bourbakis, "Image and video encryption using SCAN patterns", *Pattern Recognition*, vol. 37, pp. 725-737, 2004.

[6] G. Alvarez, A.H. Eneinas, L.H. Eneinas, and A. Martin Rey, "A secure scheme to share secrete color images", *Computer Physics Communications*, vol. 173, pp. 9-16, 2005.

[7] R.J. Chen, W.K. Lu, and J.L. Lai, "Image encryption using progressive cellular automata substitution and SCAN", *IEEE International Symposium on Circuits and Systems,* Seoul, pp. 18-25, 2005

[8] F. Maleki, A. Mohades, S. Mehdi Hashemi, and M.E. Shiri, "An image encryption system by cellular automata with memory", *The 3rd International Conference on Availability, Reliability and Security,* Vienna, pp. 102-108, 2008.

[9] R.J. Chen, and J.L. Lai, "Image security system using recursive cellular automata substitution", *Pattern Recognition*, vol. 40, pp. 1621-1631, 2007.

[10] S. Wolfram, "Statistical mechanics of cellular automata", *Review Modern Physics,* vol. 55, pp. 601-607, 2003.

[11] D. Xiao, X. Liao, and K.W. Wong, "An efficient entire chaos-based scheme for deniable authentication", *Chaos, Solutions and Fractals*, vol. 23, pp. 1327-1331, 2005.

Received: December 15, 2014 Revised: January 04, 2015 Accepted: February 25, 2015