Implementation of digital image encryption algorithm using logistic function and DNA encoding

Suryadi MT¹, Yudi Satria², Muhammad Fauzi³
¹,²,³ Department of Mathematics, Universitas Indonesia, Depok, 16424, Indonesia

E-mail: muhammad.fauzi22@sci.ui.ac.id

Abstract. Cryptography is a method to secure information that might be in form of digital image. Based on past research, in order to increase security level of chaos based encryption algorithm and DNA based encryption algorithm, encryption algorithm using logistic function and DNA encoding was proposed. Digital image encryption algorithm using logistic function and DNA encoding use DNA encoding to scramble the pixel values into DNA base and scramble it in DNA addition, DNA complement, and XOR operation. The logistic function in this algorithm used as random number generator needed in DNA complement and XOR operation. The result of the test show that the PSNR values of cipher images are 7.98-7.99 bits, the entropy values are close to 8, the histogram of cipher images are uniformly distributed and the correlation coefficient of cipher images are near 0. Thus, the cipher image can be decrypted perfectly and the encryption algorithm has good resistance to entropy attack and statistical attack.

1. Introduction

With the vast development of computer technology, digital image usage as image information storage used in the society tend to increase. Sometimes, the digital image need to be concealed because it could inflict detriments to the owner if somebody know or could manipulate those digital image. Thus, digital image cryptography emerges as one of the method to overcome this problem.

Chaos is a random or uncertain movement in a particular system. Chaos system has certain properties including aperiodic, non-convergent, and initial value sensitive [1-2]. Therefore, chaos system could be used in the cryptography field. Research on chaos based cryptography already performed [2-11] before on digital image. But, according to their research [8-10] a lot of encryption algorithms [12-14] which use a single chaos function are vulnerable to be interpreted.

Deoxyribonukleat acid (DNA) cryptography is a study which use DNA as information carrier and uses biological technology to perform encryption where it’s commonly uses common dogma in molecular biology to implement encryption. But, DNA encryption has some demerit points, it needs expensive equipment, complex operation, and difficult to grasp its biotechnology [2].

Based on the reasons that DNA encryption is hard to be implemented and the usage of single chaos function is vulnerable to be interpreted, Liu, Zhang & Wei [2] proposed encryption algorithm based on
chaos logistic function and DNA encoding. This algorithm split the channel contained in the RGB digital image, and transformed its pixel values into binary values in order to change it into DNA components according to the DNA map rule already defined. Then, DNA addition operation and DNA complement operation are performed with the help of chaos logistic function to scramble the DNA information. Finally, it changes the DNA information into binary values to be used in the XOR operation against values created with the chaos logistic function, the output values of the XOR operation will represent the cipher image’s pixels.

2. Experimental Methods
Logistic function is a recursive function which used to be referenced as an example of chaos properties complexity which may occur in the simple dynamic non-linear equation. This function is defined as follows:

\[ x_{n+1} = rx_n(1-x_n) \]  

where \( r \in (0,4) \), \( x_n \in (0,1) \), \( n = 0,1,2,... \), both \( r \) and \( x_0 \) are the initial value of the logistic function and \( x_n \) is the logistic function value with \( n \) element of the whole number [1].

The chaos logistic function values generated in this algorithm will be transformed into binary value with threshold function \( f(x) \) in the equation (2). Special function \( g(x) \) shown in equation (3) will be used to transform the chaos logistic function values into pixel values.

\[ f(x) = \begin{cases} 
0, & 0 < x_n \leq 0.5 \\
1, & 0.5 < x_n < 1 
\end{cases} \]  

\[ g(x) = \left\lfloor 100000 \times x \right\rfloor \mod 256 \]  

Adenin (A), Cytosine (C), Guanine (G), and Thymine (T) are DNA components which create a single strand of DNA, where A and T are complement to each other, so are C and G. In binary system theory, 0 and 1 are complement to each other, so 00 and 01, 10 and 11 are also complement to each other. DNA encoding in this algorithm transform binary value into 4 DNA components with 2 bit binary value to represent a DNA component, and DNA decoding transform DNA components into binary value. If 00, 01, 10, and 11 are used to represent 4 DNA components, there will be 24 possible DNA map combinations. But because there is a complement relation between DNA component and binary value, there is simply 8 kind of DNA map which satisfy complement relation from 24 possible DNA combinations [2]. This can be seen at Table 1.

| DNA Map Rules | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---------------|---|---|---|---|---|---|---|---|
| 00-A          | 00-A| 00-C| 00-C| 00-G| 00-G| 00-T| 00-T|   |
| 01-C          | 01-G| 01-A| 01-T| 01-A| 01-T| 01-C| 01-G|   |
| 10-G          | 10-C| 10-T| 10-A| 10-T| 10-A| 10-G| 10-C|   |
| 11-T          | 11-T| 11-G| 11-G| 11-C| 11-C| 11-A| 11-A|   |

Also, this algorithm using DNA addition and subtraction rules as shown at Table 2.

| DNA Addition and Subtraction Rule | +   | A   | T   | C   | G   |
|----------------------------------|-----|-----|-----|-----|-----|
| A                                | G   | C   | T   | A   |
| T                                | C   | G   | A   | T   |
| C                                | T   | A   | G   | C   |
| G                                | A   | T   | C   | G   |
3. Results and Analysis

The experiment will be performed in a computer with processor specification Intel(R) Core(TM)2 Duo CPU T6500 ~ 2.1 Ghz, 2.00 GB, Windows 7 Ultimate 32-bit Operating System. Digital images used in this experiment are 8-bit Baboon.bmp and Lena.bmp shown in Figure 1 (a) and Figure 2 (a) with 3 sizes in pixel shown in Table 3 and Table 4, and the cipher image shown in Figure 1 (b) and Figure 2 (b). The encryption and decryption keys used through all the experiment are $x_0^{(1)} = 0.25$, $r^{(1)} = 3.94$, $x_0^{(2)} = 0.52$, $r^{(2)} = 3.78$, $key1 = 7$, $key2 = 4$.

Figure 1. Display of Baboon.bmp (a) Plain Image (b) Cipher Image

Figure 2. Display of Lena.bmp (a) Plain Image (b) Cipher Image

| Data Sample | Size (pixel) | Size (bytes) |
|-------------|--------------|--------------|
| 1           | $128 \times 128$ | 49.206     |
| 2           | $256 \times 256$ | 196.664    |
| 3           | $512 \times 512$ | 786.488    |

Table 3. Baboon.bmp Data Sample Sizes.

| Data Sample | Size (pixel) | Size (bytes) |
|-------------|--------------|--------------|
| 1           | $128 \times 128$ | 49.206     |
| 2           | $256 \times 256$ | 196.662    |
| 3           | $512 \times 512$ | 786.486    |

Table 4. Lena.bmp Data Sample Sizes.

3.1. Histogram and Goodness of Fit Analysis

Table 5. Histograms of Baboon.bmp.
Table 6. Histograms of Lena.bmp.

| Data Sample | R Channel (plain) | G Channel (plain) | B Channel (plain) | R Channel (cipher) | G Channel (cipher) | B Channel (cipher) |
|-------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| 1           |                   |                   |                   |                   |                   |                   |
| 2           |                   |                   |                   |                   |                   |                   |
| 3           |                   |                   |                   |                   |                   |                   |

Table 5 and Table 6 shows the histogram of the data sample in Table 3 and Table 4. It can be seen that the histograms of the plain image data samples tend to fluctuate but the histograms of the cipher image data samples tend to have a flat shape. In order to prove the uniformity of the cipher image data samples histograms, Goodness of Fit test are performed.

Table 7. Statistical Value of Baboon.bmp Cipher Images.

| Data Sample | R Channel | G Channel | B Channel |
|-------------|-----------|-----------|-----------|
| 1           | 274.68750 | 234.40625 | 254.15625 |
| 2           | 238.55468 | 273.79687 | 241.14062 |
| 3           | 274.26757 | 257.26757 | 256.21093 |

Table 8. Statistical Value of Lena.bmp Cipher Images.

| Data Sample | R Channel | G Channel | B Channel |
|-------------|-----------|-----------|-----------|
| 1           | 244.93750 | 225.12500 | 246.84375 |
| 2           | 268.67968 | 218.30468 | 245.10937 |
| 3           | 244.36914 | 205.78710 | 231.98046 |

Table 7 and Table 8 shows the statistical test value. Because the degree of freedom is 255 and with 1% level of significance, the critical value in this Goodness of Fit test are 310.4574. It can be seen from Table 7 and Table 8 that all of the data sample statistical test values are smaller than 310.4574 which can be implied that the distribution of the cipher image pixel values are uniformly distributed. Therefore, it shows that the encryption algorithm has a good resistance against statistical attack.

3.2. Coefficient Correlation Analysis

Correlation coefficient test are also performed by using 2000 random pairs of adjacent pixels horizontally, vertically, and diagonally from the plain image and the cipher image to compute its correlation coefficient value.

Table 9. Horizontal Correlation Coefficient Values of Baboon.bmp.

| Data Sample | R Channel (plain) | G Channel (plain) | B Channel (plain) | R Channel (cipher) | G Channel (cipher) | B Channel (cipher) |
|-------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| 1           | 0.88327           | 0.74551           | 0.88117           | 0.04630           | 0.04001           | 0.01118           |
| 2           | 0.90367           | 0.81081           | 0.90927           | -0.00910          | 0.00831           | 0.01123           |
| 3           | 0.88639           | 0.75959           | 0.89902           | -0.00595          | 0.02003           | -0.00521          |
Table 3.3. PSNR and Entropy Analysis

The algorithm has a high security level against pixel correlation since the horizontal, vertical, and diagonal correlation coefficient values of the plain image are close to 1 which implied that the plain image pixels have a strong line correlation. However, the horizontal, vertical, and diagonal correlation coefficient values of the cipher image are close to 0 which implied that the linear correlation in their corresponding plain image were lost. Thus, it can be said that the encryption algorithm has a high security level against pixel correlation statistical attack.

3.3. PSNR and Entropy Analysis

Table 10 and Table 11 show the PSNR values between the plain image and the cipher image of Baboon.bmp and Lena.bmp. The horizontal, vertical, and diagonal correlation coefficient values of the plain image are close to 1 which implied that the plain image pixels have a strong linear correlation, but the horizontal, vertical and diagonal correlation coefficient values of the cipher image are close to 0 which implied that the linear correlation in their corresponding plain image were lost. Thus, it can be said that the encryption algorithm has a high security level against pixel correlation statistical attack.

Table 10. Vertical Correlation Coefficient Values of Baboon.bmp.

| Data Sample | R Channel (plain) | G Channel (plain) | B Channel (plain) | R Channel (cipher) | G Channel (cipher) | B Channel (cipher) |
|-------------|------------------|------------------|------------------|-------------------|-------------------|-------------------|
| 1           | 0.89311          | 0.73341          | 0.82691          | -0.01210          | 0.02432           | -0.01319          |
| 2           | 0.93051          | 0.89391          | 0.90991          | 0.01190           | 0.03859           | 0.03328           |
| 3           | 0.91277          | 0.88798          | 0.89985          | 0.00905           | -0.01137          | -0.00139          |

Table 11. Diagonal Correlation Coefficient Values of Baboon.bmp.

| Data Sample | R Channel (plain) | G Channel (plain) | B Channel (plain) | R Channel (cipher) | G Channel (cipher) | B Channel (cipher) |
|-------------|------------------|------------------|------------------|-------------------|-------------------|-------------------|
| 1           | 0.83893          | 0.69898          | 0.70857          | -0.01137          | 0.02120           | 0.00531           |
| 2           | 0.92393          | 0.79898          | 0.80373          | -0.02636          | -0.04144          | 0.04011           |
| 3           | 0.85858          | 0.72212          | 0.88656          | 0.01955           | 0.01342           | 0.00910           |

Table 12. Horizontal Correlation Coefficient Values of Lena.bmp.

| Data Sample | R Channel (plain) | G Channel (plain) | B Channel (plain) | R Channel (cipher) | G Channel (cipher) | B Channel (cipher) |
|-------------|------------------|------------------|------------------|-------------------|-------------------|-------------------|
| 1           | 0.92771          | 0.91070          | 0.88345          | -0.01038          | 0.00890           | 0.04182           |
| 2           | 0.95805          | 0.93792          | 0.89743          | -0.02696          | -0.03394          | -0.02621          |
| 3           | 0.98885          | 0.98316          | 0.96021          | -0.02703          | -0.02576          | -0.02131          |

Table 13. Vertical Correlation Coefficient Values of Lena.bmp.

| Data Sample | R Channel (plain) | G Channel (plain) | B Channel (plain) | R Channel (cipher) | G Channel (cipher) | B Channel (cipher) |
|-------------|------------------|------------------|------------------|-------------------|-------------------|-------------------|
| 1           | 0.85383          | 0.82398          | 0.80136          | -0.01818          | -0.00568          | -0.03831          |
| 2           | 0.92341          | 0.88146          | 0.83511          | 0.02614           | 0.00641           | -0.01494          |
| 3           | 0.97940          | 0.97123          | 0.93368          | -0.01001          | -0.01758          | -0.02634          |

Table 14. Diagonal Correlation Coefficient Values of Lena.bmp.

| Data Sample | R Channel (plain) | G Channel (plain) | B Channel (plain) | R Channel (cipher) | G Channel (cipher) | B Channel (cipher) |
|-------------|------------------|------------------|------------------|-------------------|-------------------|-------------------|
| 1           | 0.81588          | 0.80449          | 0.78441          | 0.02290           | 0.02634           | 0.00999           |
| 2           | 0.89650          | 0.86048          | 0.82421          | -0.00847          | 0.00287           | 0.00020           |
| 3           | 0.96751          | 0.95926          | 0.92277          | -0.00770          | -0.00062          | 0.00126           |
Table 15. PSNR Value of Baboon.bmp *Cipher-Plain.*

| Data Sample | R Channel | G Channel | B Channel |
|-------------|-----------|-----------|-----------|
| 1           | 8.93554   | 9.41361   | 8.45741   |
| 2           | 8.88767   | 9.43164   | 8.52543   |
| 3           | 8.77369   | 9.23965   | 8.34853   |

Table 16. PSNR Value of Baboon.bmp *Plain-Decrypted.*

| Data Sample | R Channel | G Channel | B Channel |
|-------------|-----------|-----------|-----------|
| 1           | ∞         | ∞         | ∞         |
| 2           | ∞         | ∞         | ∞         |
| 3           | ∞         | ∞         | ∞         |

Table 17. PSNR Value of Lena.bmp *Cipher-Plain.*

| Data Sample | R Channel | G Channel | B Channel |
|-------------|-----------|-----------|-----------|
| 1           | 7.86522   | 8.49217   | 9.54076   |
| 2           | 7.87227   | 8.52729   | 9.55073   |
| 3           | 7.86159   | 8.54053   | 9.59230   |

Table 18. PSNR Value of Lena.bmp *Plain-Decrypted.*

| Data Sample | R Channel | G Channel | B Channel |
|-------------|-----------|-----------|-----------|
| 1           | ∞         | ∞         | ∞         |
| 2           | ∞         | ∞         | ∞         |
| 3           | ∞         | ∞         | ∞         |

It can be seen that the PSNR values are quite small which ranges from 7.86-9.59 dB. PSNR values ≤ 30 dB implied that the encryption algorithm gives good quality cipher image [15]. Table 16 and Table 18 shows PSNR values between the plain image and the decrypted image. It can be seen that all the values are infinite, thus it proves that there are not any pixel differences between the plain image and the decrypted image and the decryption algorithm returns the decrypted image perfectly.

Table 19. Entropy Value of Baboon.bmp *Cipher Image.*

| Data Sample | R Channel | G Channel | B Channel |
|-------------|-----------|-----------|-----------|
| 1           | 7.98793   | 7.98972   | 7.98891   |
| 2           | 7.99736   | 7.99697   | 7.99732   |
| 3           | 7.99924   | 7.99929   | 7.99929   |

Table 20. Entropy Value of Lena.bmp *Cipher Image.*

| Data Sample | R Channel | G Channel | B Channel |
|-------------|-----------|-----------|-----------|
| 1           | 7.98913   | 7.99004   | 7.98908   |
| 2           | 7.99702   | 7.99759   | 7.99729   |
| 3           | 7.99932   | 7.99910   | 7.99936   |

Table 19 and Table 20 shows the entropy values between the plain image and the cipher image of Baboon.bmp and Lena.bmp data samples. It shows that its entropy values ranges from 7.98-7.99 *bits*, close to 8 *bits*. Thus, it can be implied that the encryption algorithm has a good resistance against entropy attack[2].
3.4. Encryption and Decryption Time Analysis

| Table 21. Average Encryption and Decryption Time of Baboon.bmp. |
|-------------|----------------|----------------|
| Data Sample | Encryption (second) | Decryption (second) |
| 1           | 1.83520          | 1.89979         |
| 2           | 7.31240          | 7.52479         |
| 3           | 29.39357         | 30.50018        |

| Table 22. Average Encryption and Decryption Time of Lena.bmp. |
|-------------|----------------|----------------|
| Data Sample | Encryption (second) | Decryption (second) |
| 1           | 1.81410          | 1.86235         |
| 2           | 7.14926          | 7.34866         |
| 3           | 29.11883         | 30.48092        |

Table 21 and Table 22 show the average of encryption and decryption time measured in seconds. This was taken from the average of 5 test with the same key. Table 21 and Table 22 show that the encryption and decryption time tend to increase as the image size increased.

4. Conclusion

According to the program implementation and results analysis on the data samples in Table 4 and Table 5 we can see that the cipher image’s histograms and pixels are uniformly distributed hence the encryption algorithm can resist statistical attack, all the data sample cipher image’s coefficient correlation values close to 0 hence the encryption algorithm can resist pixel correlation statistical attack, cipher-plain image’s PSNR values range around 7.8-9.5 dB under 30 dB hence the encryption algorithm has a good quality, plain-decrypted image’s PSNR values are infinite hence the decrypted images are equal plain images, cipher image’s entropy values range around 7.98-7.99 bits which is close to 8 bits hence the encryption algorithm can resist entropy attack, and the bigger the data sample image size the bigger the encryption and decryption time of the data sample.

Acknowledgment

The authors would like to thank Kemenristekdikti RI and Directorate of Research and Community Service (DRPM) Universitas Indonesia for the grant PTUPT 2017 who support us in this research.

References

[1] Kocarev L and Lian S 2011 *Chaos-Based Cryptography: Theory, Algorithm and Applications*, (Berlin: Springer-Verlag)
[2] Liu L, Zhang Q and Wei X 2012 A RGB image encryption algorithm based on DNA encoding and Chaos Map *Computers and Electrical Engineering design* **38** 1240-1248
[3] Eva N and Suryadi MT 2013 Chaos-Based Encryption Algorithm for Digital Image *Proceedings IICMA* 169-177
[4] Suryadi MT, Maria YTI, and Satria Y 2015 Encryption Algorithm using New Modified map for digital image *Proceedings of IICMA* 71-78
[5] Suryanto Y, Suryadi MT, and Ramli K 2016 A Secure and Robust Image Encryption Based on Chaotic Permutation Multiple Circular Shrinking and Expanding *Journal of Information Hiding and Multimedia Signal Processing* **7** 697-713
[6] Suryanto Y, Suryadi MT, and Ramli K 2017 A New Image Encryption using color scrambling based on chaotic permutation multiple circular shrinking and Expanding *Multimedia Tools and Applications* **76** 16831-16854
[7] Ahmad J and Ahmed F 2012 Efficiency Analysis and Security Evaluation of Image Encryption
Schemes *International Journal of Video & Image Processing and Network Security* **12** 18-31
[8] Laptyeva TV, Flach S and Kladko K 2011 The weak password problem: Chaos, criticality, and encrypted p-CAPTCHAs *EPL* **95** 981-981
[9] Shujun L and Xuan Z 2002 Cryptanalysis of a chaotic image encryption method *International Symposium on Circuits and System* **2** 87-91
[10] Qinan L 2011 Color image encryption algorithm and its decryption method protecting from shearing attack *Computer engineering and design* **32** 509-512
[11] Suryadi MT, Nurpeti E, Widya D 2014 Performance of Chaos-Based Encryption Algorithm for Digital Image *TELKOMNIKA* **12** 675-682
[12] Sun F, Liu S and Li Z 2008 A novel image encryption scheme based on spatial chaos map *Chaos Solitons Fractals* **38** 631-640
[13] Vinod P, Pareek NK, Purohita G and Suda KK 2011 A robust and secure chaotic standard map based pseudorandom permutation substitution scheme for image encryption *Optics Commun* **284** 4331-4339
[14] Yunpeng Z, Fei Z, Zhengjun Z and Cai X 2008 A new image encryption Algorithm based on Multiple Chaos System *International Symposium on Electronic Commerce and Security* **142** 347-350
[15] Pareek NK 2012 Design and Analysis of a Novel Digital Image Encryption Scheme *International Journal of Network Security & Its Applications* **4** 95-108