# Design and Implementation of Chaotic Image Encryption System Based on DNA Coding

Mengqi Wu

# Contents

# 1 Introduction

Human communication has become faster because to the advent of the Internet, and the network is also used to transfer and store information and data from a variety of areas. Nevertheless, there is also a risk of malicious assaults, leaks, and tampering when it comes to some sensitive data and individual information[1] The security of digital image storage and transmission is receiving increasing attention as one of the most vital Internet communication methods. As a result, the question of how to quickly and efficiently encrypt digital photographs has gained popularity as a research area.

The first picture encryption utilizing a DNA encryption technique[2] was implemented in 2003 by Gehani et al. In 2010, Zhang introduced a new image encryption scheme based on DNA addition operation and a chaotic system image encryption scheme[3]. In 2012, Liu and Zhang et al. suggested an RGB image encryption technique based on DNA coding and chaotic systems[4]. Furthermore, Hongkai Li et al. (2016) integrated a two-dimensional logical chaotic system with Chen's hyperchaotic system to achieve image encryption using addition and subtraction algebraic operations on DNA sequences[5]. These algorithms can not only produce effective encryption but also withstand a number of destructive actions including differential and statistical assaults. However, they only utilize one and a simple chaotic system model, and the DNA encoding rules present a single and fixed challenge. Our image encryption is pointless because the algorithm's overly simplistic

and unified design is simple enough for criminals to decrypt. Additionally, only a small number of researchers have been able to resolve the issue of the algorithm's weakness against some plaintext assaults, which implies that the security of the data is not guaranteed.

So, based on the current encryption techniques, this research suggests certain enhancements. In order to achieve better encryption through the integration of several models, in this study we will build and implement a color encryption algorithm based on the combination of Logistic mapping, Chen hyperchaotic picture, and DNA encoding operation. In the meantime, this study will evaluate the security of the encryption technique by examining the numerical indicators of the ciphertext image using typical cryptographic attacks.

# 2 Methods

The algorithm processes the image by RGB three color channels separately, and uses the pixel information of the plaintext image as the initial value to generate the pseudo-random sequence by iteration of Logistic mapping; then, according to the four chaotic sequences generated by Chen hyperchaotic system, the plaintext image and the pseudo-random sequence are encrypted and operated in blocks by using DNA encoding rules and operation rules; finally, all the sub-blocks are merged and disrupted to obtain the ciphertext image.

1) Read the color digital image I of size $length \times width$, which is divided into 3 two-dimensional matrices $I_1,\ I_2,\ and\ I_3$ according to R, G, and B channels.

$$I_1 = I(:,:,0)$$
$$I_2 = I(:,:,1) \quad (3.1)$$
$$I_3 = I(:,:,2)$$

2) Set the chunk size to complement the number of rows and columns of the matrix with zeroes so that all 3 two-dimensional matrices satisfy the equation (3.2).

$$\begin{cases} mod(length, size) = 0 \\ mod(width, size) = 0 \end{cases} \quad (3.2)$$

Where $mod$ —— means to take the balance.

The dimensions of the new matrices are reassigned to $length$ and $width$, so the two-dimensional matrices $I_1$, $I_2$, $and\ I_3$ are divided into $(length \times width)/size^2$ subblocks.

3) Set the initial value $x_0$ and the branching parameter $\mu$, and iterate the loop for the logistic mapping according to equation (2.1). To obtain better randomness, the first 1000 values are removed to obtain a chaotic sequence $p$ of $length \times width$. where the branching parameter $\mu$ is 3.9999 and the initial value $x_0$ is obtained from equation (3.3), and $x_0$ is also used as part of the key.

$$x_0 = \frac{sum(I_1) + sum(I_2)}{255 \times length \times width \times 2} \quad (3.3)$$

Where $sum(I_1)$, $sum(I_2)$ —— the sum of all values of the matrix $I_1$ and $I_2$.

4）According to equation (3.4), the chaotic sequence $p$ is transformed into a two-dimensional matrix $H$ of size $length \times width$, and each element value in the matrix is an integer between 0 and 255.

$$\begin{cases} p = mod(round(p \times 10^4), 256) \\ H = p.reshape(length, width) \end{cases} \quad (3.4)$$

Where $round$ —— denotes rounding to the nearest integer;

$p.reshape(length, width)$ —— denotes transforming the sequence $p$ into a matrix of$length \times width$.

5) Set the parameters and initial values of Chen's hyperchaotic system $X(0)$, $Y(0)$, $Z(0)$, $W(0)$, and solve the equations according to equation (2.2) using Python's odeint() function. In order to obtain better randomness, the first 3000 values are removed to obtain four pseudo-random sequences of length $(length \times width)/size^2$ for $X$, $Y$, $Z, and\ W$. The initial value $X(0)$, $Y(0)$, $Z(0)$, $W(0)$ of the Chen hyperchaotic system are calculated from equation (3.5), and these four values are also part of the key.

$$\begin{cases} X(0) = \dfrac{sum(sum(bitand(I_1, 17)))}{17 \times length \times width} \\ Y(0) = \dfrac{sum(sum(bitand(I_2, 34)))}{34 \times length \times width} \\ Z(0) = \dfrac{sum(sum(bitand(I_3, 68)))}{68 \times length \times width} \\ W(0) = \dfrac{sum(sum(bitand(I_1, 136)))}{136 \times length \times width} \end{cases} \tag{3.5}$$

Where, bitand —— indicates the operation by bitwise AND

Because the values in the three matrices $I_1, I_2, and\ I_3$ split from the original image are all integers from 0 to 255, namely each value can be represented by an 8-bit binary number, then the whole matrix can be regarded as 8 bit-planes. If the matrix $I_1$ and the binary number 00010001 do the by-bit and operation, that is 17, you can get the average of the first and fifth bit-plane of matrix $I_1$. Therefore, the four initial values obtained above are determined by the average of the first and fifth bit-planes of

$I_1$, the second and sixth bit-planes of $I_2$, the third and seventh bit-planes of $I_3$ and the fourth and eighth bit-planes of $I_1$, respectively. This satisfies that each image has a different initial value, and also maintains a certain correlation with the original image.

6) The pseudo-random sequences $X$ and $Y$ determine the DNA encoding laws for the two-dimensional matrices $I_1, I_2, and\ I_3$ and the chaotic matrix $H$, respectively. there are eight DNA encoding laws, so all elements in sequences $X$ and $Y$ are converted to integers from 1 to 8. Similarly, sequence $Z$ determines the DNA algorithm between the corresponding sub-blocks of two-dimensional matrices $I_1, I_2, and\ I_3$ and chaos matrix $H$. There are four DNA algorithms, so all elements in sequence $Z$ are converted to integers from 1 to 4. The sequence $W$ determines the decoding method after DNA operation, so all the elements in the sequence $W$ are converted to integers from 1 to 8. So the four sequences $X,\ Y,\ Z, and\ W$ are processed as follows.

$$\begin{cases} X = mod(round(X \times 10^4), 8) + 1 \\ Y = mod(round(Y \times 10^4), 8) + 1 \\ Z = mod(round(Z \times 10^4), 4) + 1 \\ W = mod(round(W \times 10^4), 8) + 1 \end{cases} \quad (3.6)$$

Meanwhile, the two-dimensional matrices $I_1, I_2, and\ I_3$ and the ith sub-block of the chaotic matrix H are operated using the operation rule $Z_i$, which stipulates that for $Z_i = 1$, the addition operation rule is used; for $Z_i = 2$, the subtraction operation rule is used; for $Z_i = 3$, the dissimilar or operation rule is used; and for $Z_i = 4$, the same or operation rule is used. It should be noted that for better diffusion effect, all the sub-blocks in the matrix of $I_1, I_2, and\ I_3$ except the first sub-block perform DNA operation

and then perform another DNA operation with the previous sub-block, and the operation rule is still determined by $Z_i$. DNA decoding is the inverse process of DNA coding, that is, the conversion from DNA sequence to decimal number, and the i-th sub-block is decoded according to the decoding rule $W_i$ decoding.

7) After DNA encoding, DNA operation and DNA decoding, the three encryption matrices obtained still need row and column permutation, and the basis of permutation is determined by the other two chaotic sequences of Logistic mapping. After setting the branching parameter μ and the initial values $x_1$ and $x_2$, the iterative loop still follows equation (2.1) to remove the first 1000 values and obtain two chaotic sequences $K_x$ and $K_y$ with $length$ and $width$ respectively. $\mu$ remains 3.9999 and the initial values $x_1$ and $x_2$ are calculated by the following equation, and these two initial values are also used as part of the key.

$$\begin{cases} x_1 = \dfrac{sum(I_1) + sum(I_3)}{255 \times length \times width \times 2} \\ x_2 = \dfrac{sum(I_2) + sum(3)}{255 \times length \times width \times 2} \end{cases} \tag{3.7}$$

8) Arrange the two obtained sequences $K_x$ and $K_y$ in descending order, and record the position of each value in the original order using the index sequences $U_x$ and $U_y$. The index sequences $U_x$ and $U_y$ are obtained using the following formula.

$$\begin{cases} U_x = sort(-K_x) \\ U_y = sort(-K_y) \end{cases} \tag{3.8}$$

where $sort(-K_x)$, $sort(-K_y)$ —— indicates that the sequence $K_x$ $and$ $K_y$ are arranged in descending order

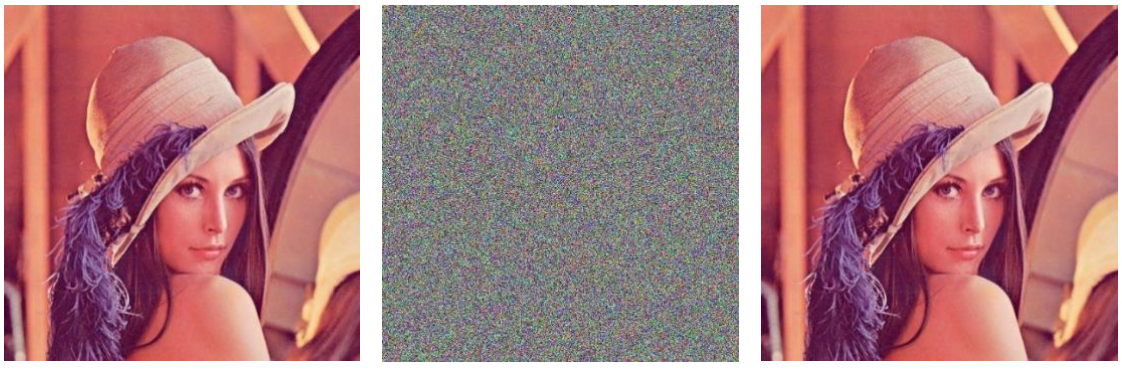
The values of $U_x$ and $U_y$ and their corresponding index values are used as the

exchange coordinates of rows and columns to perform row and column permutation of the encrypted matrix, so as to obtain a better permutation effect.

9) Merge the three 2D matrices obtained from the operation into a 3D matrix to get the ciphertext image.

# 3 Results

To verify the effectiveness of the image encryption algorithm in this paper, experiments are conducted on a standard test color image Lena of size 512×512×3. The algorithm is implemented in Python language and run under Windows 10 operating system environment with 8GB memory and 2.50GHz CPU. The key consists of $x_0, x_1, x_2, X(0), Y(0), Z(0), W(0)\ and\ size$ where the chunk size is 4. Figure 3.3(a)-Figure 3.3(c) shows the original image, the ciphertext image, and the decrypted image, respectively. Just from the naked eye, it can be seen that the ciphertext image shows snowflake noise and has no correlation with the original image.



（a）original image　　（b）ciphertext image　　（c）decrypted image

Figure 3.3 experimental result

# 4 Conclusion

The algorithm designed in this paper is tested for its encryption effect by using lena color image as an example, and the results show that the chaotic image encryption algorithm based on DNA coding designed in this paper successfully achieves the encryption of the image. Compared with other existing studies, the algorithm designed in this paper can resist well the chosen plaintext attack. Since the key of this algorithm is unique and random, it is difficult for an attacker to derive the key and crack the algorithm.

# 5 Recommendations

Although the implementation of the basic functions is completed, there are still some shortcomings and room for improvement.

1) The time consuming of the encryption algorithm is too long, which will cause the web page to receive data feedback very slowly and affect the users' usage. The computational process of key generation is considered to be simplified at a later stage to improve the running speed of the algorithm.

(2) The algorithm only allows encryption of one image at a time, which will cause network congestion or system crash in the case of multiple users calling this algorithm at the same time. Later, concurrent technology will be used to meet the needs of multiple images and multiple users.

# 6 References


[1] Zhang, Q., Zhu, J., & Ding, Q. (2020). OBBC: A blockchain-based data sharing scheme for open banking. In *CCF China Blockchain Conference* (pp. 1-16). Springer, Singapore.

[2] Gehani, A., LaBean, T., & Reif, J. (2003). DNA-based cryptography. In *Aspects of Molecular Computing* (pp. 167-188). Springer, Berlin, Heidelberg.

[3] Zhang, Q., Guo, L., & Wei, X. (2010). Image encryption using DNA addition combining with chaotic maps. *Mathematical and Computer Modelling*, *52*(11-12), 2028-2035.

[4] Liu, L., Zhang, Q., & Wei, X. (2012). A RGB image encryption algorithm based on DNA encoding and chaos map. *Computers & Electrical Engineering*, *38*(5), 1240-1248.

[5] Li, H.K., Qiu, G.Y., & Wang, T.. (2016). A DNA random encoding-based encryption algorithm for true color maps. Computer Application Research, 33(4), 1132-1136.

[6] May, R. M. (2004). Simple mathematical models with very complicated dynamics. In *The Theory of Chaotic Attractors* (pp. 85-93). Springer, New York, NY.

[7] Chen, G., & Ueta, T. (1999). Yet another chaotic attractor. *International Journal of Bifurcation and chaos*, *9*(07), 1465-1466.

[8] Guesmi, R., & Farah, M. A. (2021). A new efficient medical image cipher based on hybrid chaotic map and DNA code. *Multimedia tools and applications*, *80*(2),

1925-1944.