MAJOR PROJECT REPORT

on

IMAGE ENCRYPTION USING DNA ENCODING

Submitted by

Kathir Sudan(R134218080)

Ankur Vijayvargiya(R134218023)

Anish Bansal(R134218020)

Abhay Jhouta(R134218002)

Under the guidance of

Dr. Manoj Kumar

UPES

UNIVERSITY WITH A PURPOSE

SCHOOL OF COMPUTER SCIENCE

UNIVERSITY OF PETROLEUM & ENERGY STUDIES

Bidholi Campus, Energy Acres, Dehradun – 248007.

Jan May-2022

Abstract

Cryptography algorithms play an important role in data security that works on complex concepts as well statistical processes to improve and develop good and efficient encryption algorithm. It can also be described as an art that allows people to hide their details in the electronic world. Using encryption, one can protect their personal information by converting the information in some form that can only be decrypted by the owner of the information. Nowadays exchange of data using images is rising. So, there is a need to encrypt the image to protect it from hackers. Our work mainly focuses on the cipher algorithm which is based on the arithmetic operations and biological process DNA. Image will be encrypted using DNA process and XOR arithmetic operation. In this, we have also used chaotic sequence to encrypt the image. With the combination of these 2 techniques, one can securely transfer and save the image in encrypted format.

Introduction

In today's world, Internet has become a very common tool for transmitting information. Communications between people have had a revolutionary change, and the transmission of digital images over the Internet has become more and more popular. A large number of videos and images are transmitted on the internet every day. However, the openness and sharing of networks bring attention to the security of digital images from major threats while transmission. These security issues become more important in fields like military affairs, commerce, and medical treatment. In these fields, digital images should have the highest level of protection. There are various protection methods, but the image encryption technique is one of the most efficient and common methods for the protection of image information.
In the case of data protection, other encryption techniques are used. In some cases, images are saved and transferred as data. In these cases, image processing techniques are widely used to reduce image sizes. Images are encrypted to protect them from unauthorized access and thus to store, communicate, and transmit information confidential. There are symmetric and asymmetric techniques for data encryption. But in image encryption, asymmetric encryption techniques may become more complex due to the

public-private key combination. Therefore, symmetric encryption techniques are more commonly used in image encryption. This method required a random sequence of key and the chaotic function is very useful for generating the random sequence. These functions take an initial value and then it generates the pseudo-random sequence which is very hard to analyze. Chaotic functions are very sensitive to initial conditions. So chaotic system provides a good level of encryption to the image. The advantage of chaotic encryption include that it is easy to implement, it is fast and it is highly secure. The major steps of a chaotic system are scrambling and diffusion. Scrambling is used for the reallocation of the pixels which reduces the correlation between pixels. It can be done with the help of a chaotic sequence. Scrambling has a better encryption effect but without changing pixels value, the histogram of the original image and cipher image would be the same therefore its security may be threatened by the statistical analysis. In diffusion, the pixel values of the image change with the help of a chaotic sequence. Diffusion provides higher security in comparison to shuffling. Therefore, in order to improve the encryption effect, researchers have combined scrambling with diffusion.

The encryption methods of chaotic maps are divided into one-dimensional chaotic encryption and multi-dimensional chaotic encryption. The one-dimensional chaotic map is easy to implement, has a simple structure, and the time required to generate a chaotic sequence is less. However, the authors concluded that there is a drawback in low-dimensional sequence like 1D chaotic map, because it has a short password cycle, low accuracy which creates difficulty in the security of image encryption, and it is easily breakable. Higher-dimensional chaotic maps provide a rich amount of randomness and a high level of security to the image. Wang et al. proposed an efficient algorithm for image encryption. First, the plain image was divided into a specified number of blocks. Then, the blocks were shuffled and pseudorandom numbers were also generated with the help of spatiotemporal chaos. Wang et al. also showed how the proposed algorithm is faster and more secure.

In the recent scenario, the DNA-based image encryption approach has gained so much popularity and it is proven to be more efficient and accurate. In this approach, the plain image converts to the DNA sequence, and subsequently, the encrypted image is produced using DNA rules. In order to improve the results of cipher image in this paper, we use the simple theory of the DNA sequence operation to encrypt image information and combined the chaotic maps and DNA sequence XOR operation to implement image encryption. Wei et al. proposed a DNA-based image encryption method that used a hyper-chaotic system for color images. The encryption process started by splitting the color image into three matrices of red green and blue and then these matrices

converted to the DNA sequence. After that element of DNA sequence matrices were shuffled using chaotic sequence. Wei et al. proved that the proposed method is resistant against exhaustive attacks, statistical attacks, and differential attacks. Zhang et al. also used a hyper-chaotic system as well as DNA sequence operation and image fusion. Simulation results proved the robust encryption effects and the ability to resist both exhaustive attacks and statistical attacks.

Basic Theory of Proposed Algorithm

DNA Encoding and Complementary Rule

DNA sequencing is the process of determining the sequence of nucleic acids or the sequence of nucleotides in DNA. Adenine, Guanine, Cytosine and thymine are four bases of DNA. Basically, they are building blocks of DNA. Adenine binds with Thymine and Guanine bind with Cytosine. Every image in the digital world can be represented by 8-bit binary numbers. So, in binary, 0 and 1 complement each other, so 00 and 11 are complementary to each other, 01 and 10 are also complementary to each other. Here, we represent A, T, G and C with 00, 11, 01 and 10 respectively. This is done so that each pixel can be represented into a string of nucleotides. For example, if pixel value is 223 then its binary representation is 11100001. So according to the rules mentioned above, the string of nucleotide for this pixel value will be “TCAG”. There are 24 types of combinations of these four nucleotides. However, only eight coding combinations are applicable for the principle of complementarity. These rules are shown in Table 1.

Table 1: DNA decoding and encoding rules

RULE	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
00	A	A	T	T	G	G	C	C
01	G	C	G	C	A	T	A	T
10	C	G	C	G	T	A	T	A
11	T	T	A	A	C	C	G	G

During the encoding, nucleotides will be substituted using the DNA complementary rule. Also, we can select any one rule randomly thereby increasing strength of encryption. In this paper we have selected rule 2 for encoding.

DNA operations

We have to change the pixel value so that the cipher image's histogram is far different from the original image. For doing this we have to do some kind of operation between the pixel values. And these values are encoded in DNA sequence so we have to perform DNA operations. There can be addition, subtraction, and XOR operations among the DNA nucleotides. These operations produce the same results as those produced in their binary equivalent. In this paper, we have used DNA XOR operation for the encryption. The xor value of all possible combinations are shown below-:

Table 4: DNA XOR operation

XOR	**A**	**G**	**C**	**T**
A	A	G	C	T
G	G	A	T	C
C	C	T	A	G
T	T	C	G	A

Chaotic Map

Chaotic Maps are used to generate the pseudo-random sequence so that it can be used to confuse the pixels. We have used 3-D Lorenz Logistic Map to generate the chaotic sequence. This map is defined by non-linear system of ordinary differential equation:

$x_{dot} = a(y - x)$

$y_{dot} = c * x - y - x * z$

$z_{dot} = x * y - b * z$

, where a, b, c are the control parameters while x, y, z are the state variables. x_{dot}, y_{dot} and z_{dot} are the time derivatives of the state variables x, y and z respectively.

For the proposed image encryption algorithm, we have fixed value of a = 10, b = 2.667
, c= 28. We will also generate initial variables using the image. This should be kept secret because if someone knows these then encryption becomes weak.

These initial conditions are very sensitive to changes. Any small change in the initial conditions will cause the trajectories to change so much, making it difficult to predict any outcome without knowing the initial conditions of the system and thereby making it difficult to decrypt. So, it is necessary that our initial condition is only known by the owner of the image. Combining DNA encoding with chaos encryption method increases the level of encryption. If we combine DNA encoding, even the histogram of the image will change as we will do XOR operation with the generated Secret Key.

Proposed Algorithm

Step 1: Generation of Secret Key:

The inputted image is a 3d matrix which consists of red, green, blue 2d matrixes. Let this is image is represented by the matrix A of size (M, N, O). So,

A= A(i,j,k) where

i=1,2,……M and j=1,2,….. N and k=1,2,…….O

Here A(i, j,k) represents the value of the image pixel at (i,j,k). We will generate a secret key by first condensing the given 3d matrix into a 1-D array and then generating the 256 hash of the array using the SHA-256 algorithm which converts it into 256 bits code.

Step 2: Now use this generated secret key to generate the initial values x_o, y_o, and z_o for the chaotic map.

Step 3: Split the original image into three different matrices of red, green, and blue. and convert these matrices into the binary matrix.

Step 4: encode these matrices into DNA sequence by using one of the rules from Table 1.

For example, by using rule 2

01 10 11 01		C G T C
10 11 11 01	=>	G T T C
00 01 10 00		A C G A
01 01 10 10		C C G G

Step 5: Similarly, convert the secret key generated int step 2 into 2-dimensional matrix and encode this matrix key into the DNA sequence.

Step 6: Now perform the XOR operation between the each DNA encoded matrices and DNA encoded matrix key by using the XOR operation shown in Table 4.

Step 7: Generate three chaotic sequence such as X = {x1, x2, x3, ..., xk}; Y = {y1, y2, y3, ... yk} and Z = {z1, z2,z3, ..., zk},by using Lorenz 3-d chaotic map .

Step 8: Now we will shuffle the values of all three R, G, B matrices. For this we will prepare 3 more sequence X_sort, Y_sort, Z_sort such that

X_sort = sort(X);

Y_sort = sort(Y);

Z_sort = sort(Z);

For the R matrix, we will shuffle its value by using the X and X_sort sequence. Since the R matrix is a 2-d array and X is the 1-d sequence so we will convert that matrix into a 1-d array and after shuffling we will convert back into a matrix. For example let X = {x1, x2, ⋯., xk}, we will find the position of x1 into X_sort and we will take a pixel from R matrix which position is the same as x1 in sequence X and put that pixel into a position which is equal to the position of x1 into sequence X_sort. And for the G and B matrix we will do the same by using sequences Y and Z respectively.

Step 9: Now convert all three DNA encode scrambled matrices it back into binary matrices and combine them for retrieve the encrypted image.

Step 10: Now for decryption of the image, the process is the inverse of the encryption process. The receiver will get the secret key from the sender which we have generated in step 1.

Flow chart

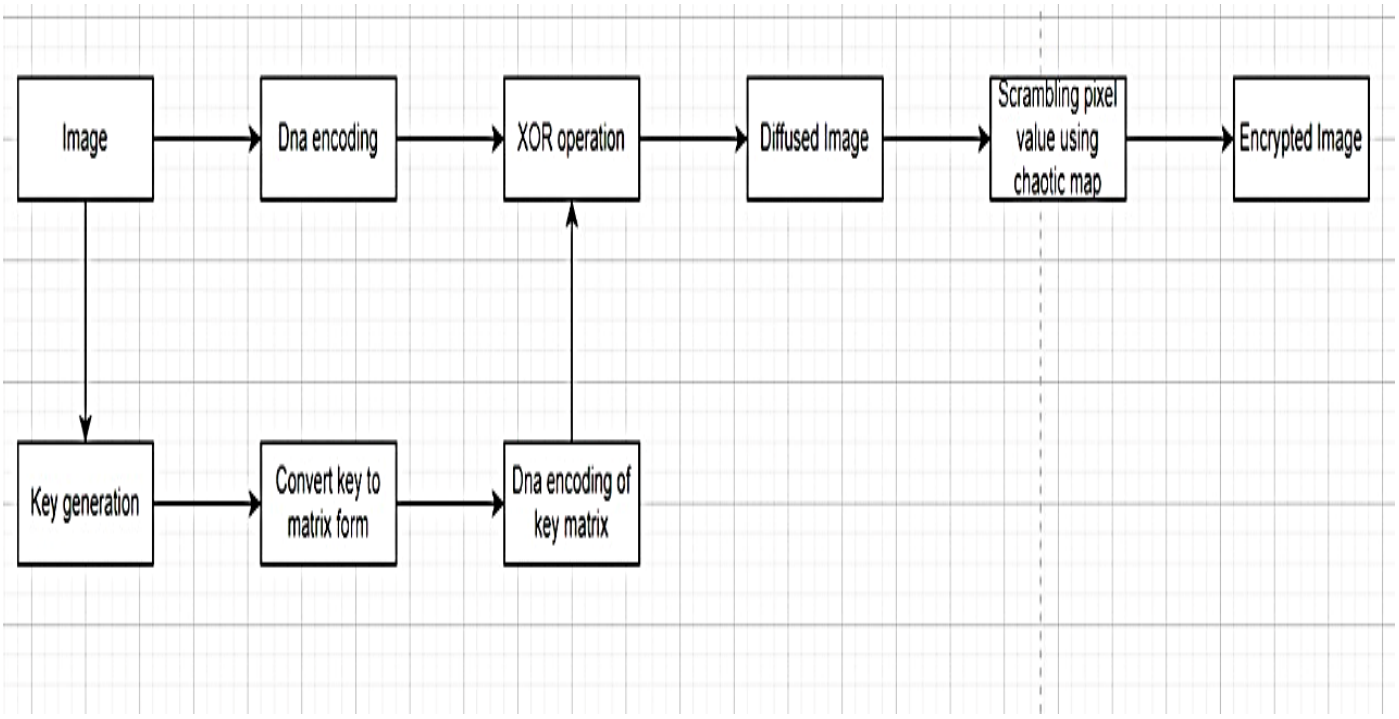


Image
Dna encoding
XOR operation
Diffused Image
Scrambling pixel value using chaotic map
Encrypted Image
Key generation
Convert key to matrix form
Dna encoding of key matrix

Experimental Results

To generate the sequence from Lorenz chaotic map, we have set initial parameters as a_o= 10, b_o =2.667 and c_o=28.The initial values x_0 , y_0 and z_o are generated dynamically using the secret key. The original image and encrypted images are shown below.

Original Image

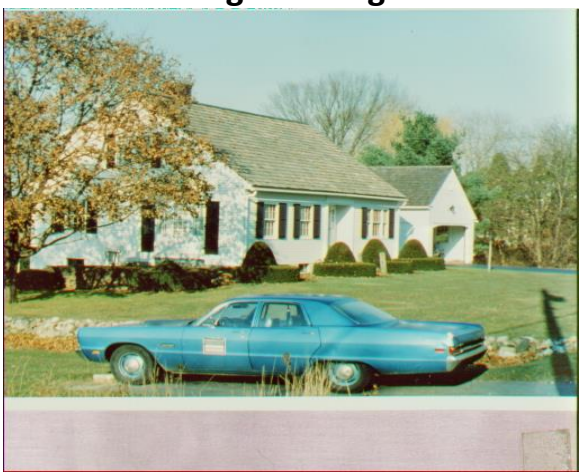


Fig 1.1

Fig 1.3

Encrypted Image

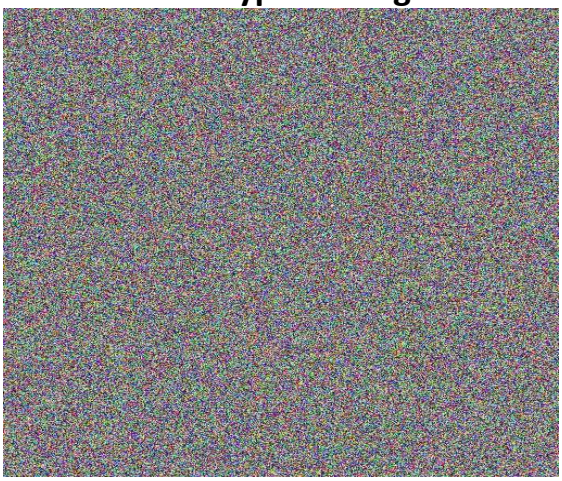


Fig 1.2

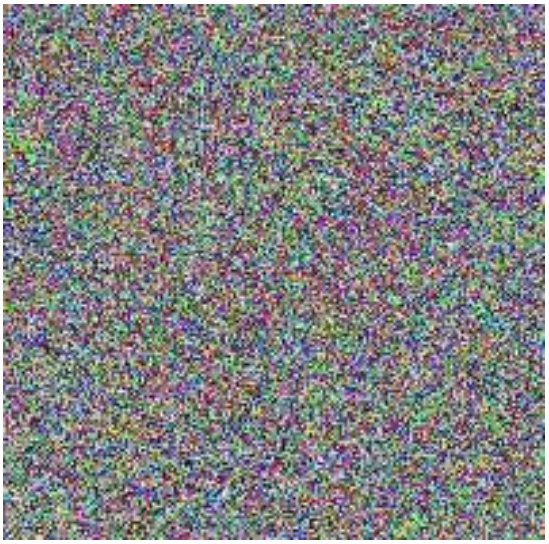
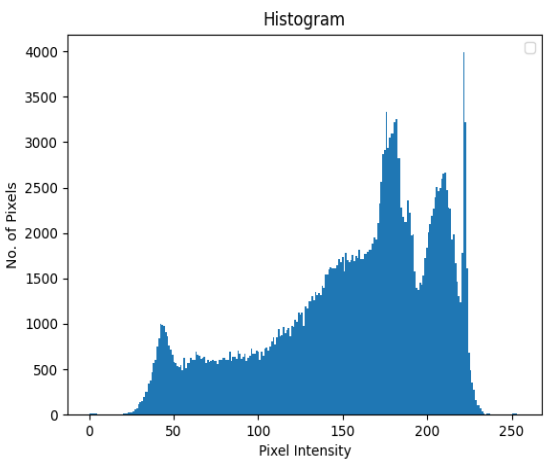


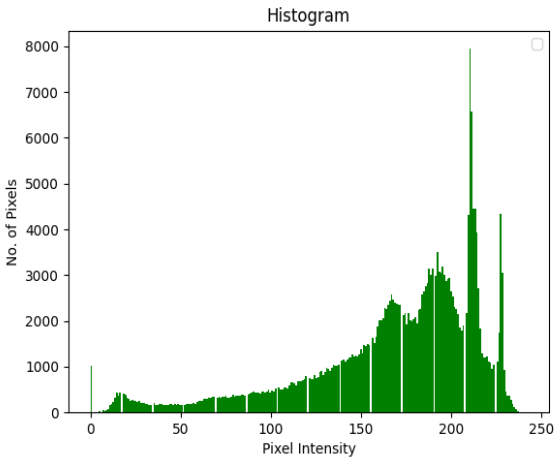
Fig 1.4

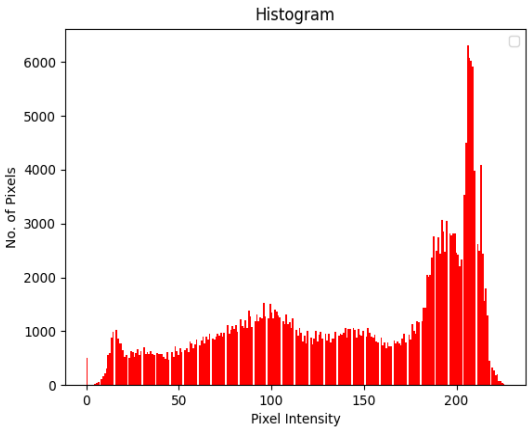
Algorithm Performance

Histogram Analysis

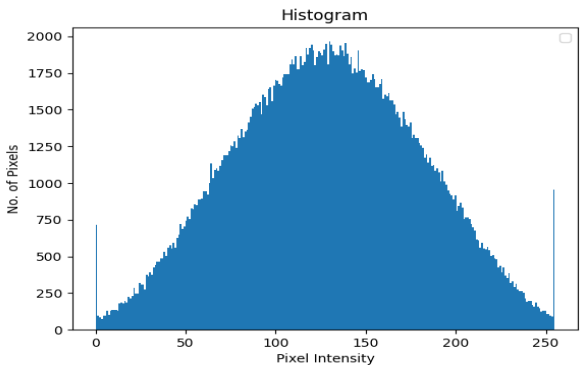
In this statistical analysis of the original and encrypted images, we construct the histogram of each colour of the above displayed images (both original and encrypted). This show that histogram of encrypted and decrypted images are quite different which makes it more difficult to decrypt.

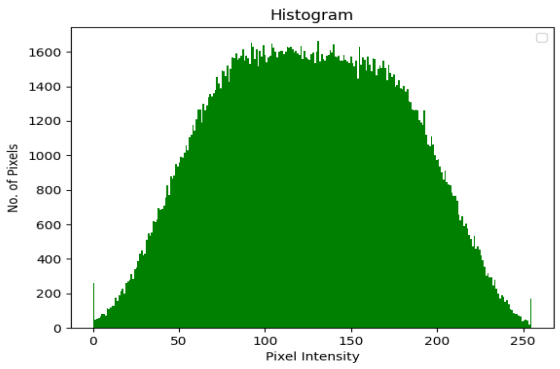


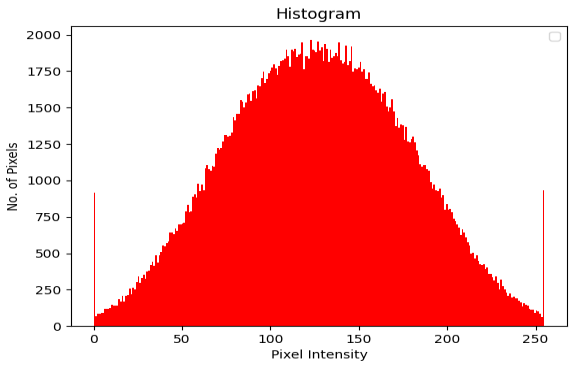




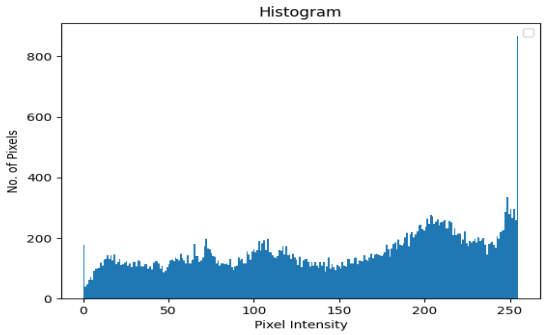
Histogram of Fig 1.1 (Original Image)

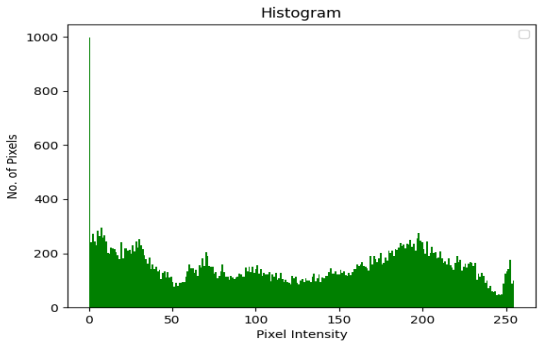


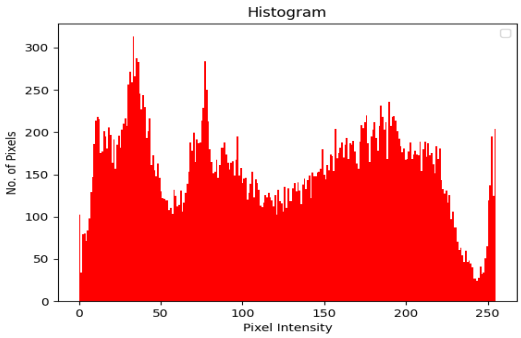




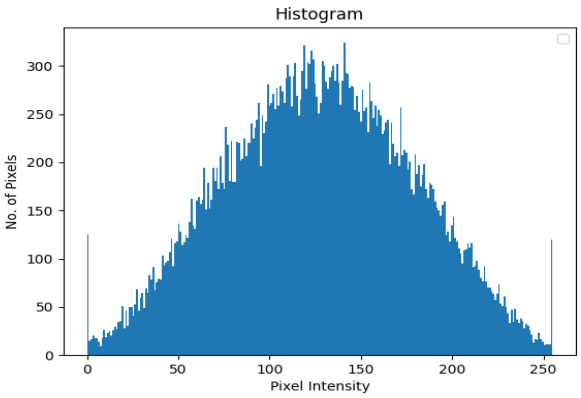
Histogram of Fig 1.2(Encrypted Image)

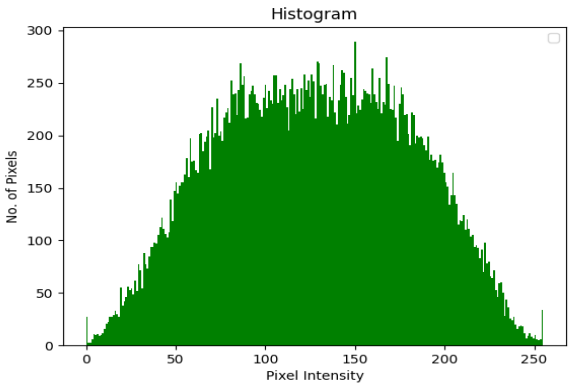


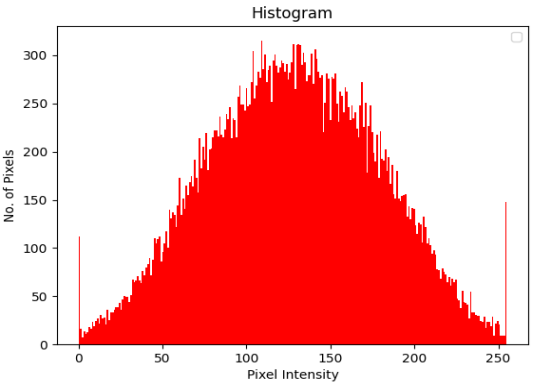




Histogram of Fig 1.3(Original Image)







Histogram of fig 1.4(Encrypted Image)

Correlation Coefficient

Correlation Coefficient is one of the ways that is used to find the degree of linear correlation between two random values. The value of the correlation coefficient r can vary from -1 to 1. r>0 indicated positive correlation while r<0 means negative correlation. Meaning for different values of r are given in the table 3. For finding correlation coefficient, we will take pairs of adjacent pixels in all the three directions (horizontal, vertical and diagonal) for both encrypted and original image. Table 4 shows the results of correlation coefficients of both the images. Then we calculate the correlation coefficient using formulas given below:

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)} * \sqrt{D(y)}}$$

$$cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y))$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}\left(x_i - E(x)\right)^2$$

Therefore rxy=

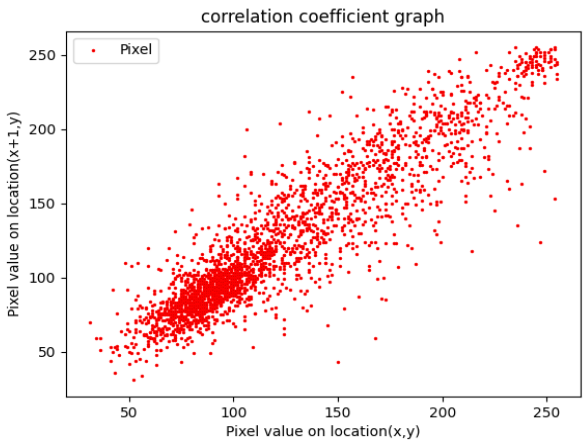
$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{\left[n\sum x^2 - (\sum x)^2\right]\left[n\Sigma y^2 - (\sum y)^2\right]}}$$

Value of r	*Indication*
+1	Complete positive correlation
+0.8	Strong positive correlation
+0.6	Moderate positive correlation
0	No correlation
-0.6	Moderate negative correlation
-0.8	Strong negative correlation
-1	Complete negative correlation

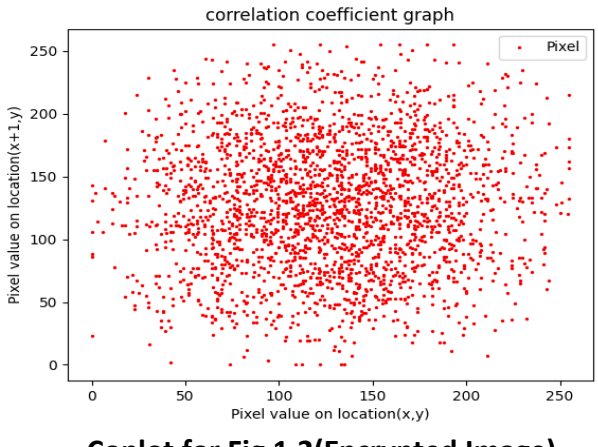
Table 3

Correlation	*Horizontal*	*Vertical*	*Diagonal*
Fig 1.1(original)	0.96774	0.62505	0.912253
Fig 1.2(encrypted)	0.09159	0.06559	0.10730
Fig 1.3(original)	0.95473	0.609207	0.916110
Fig 1.4(encrypted)	0.08226	0.060396	0.08016

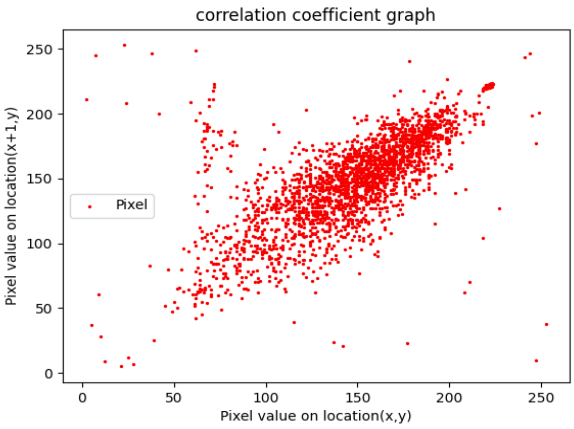
Table 4



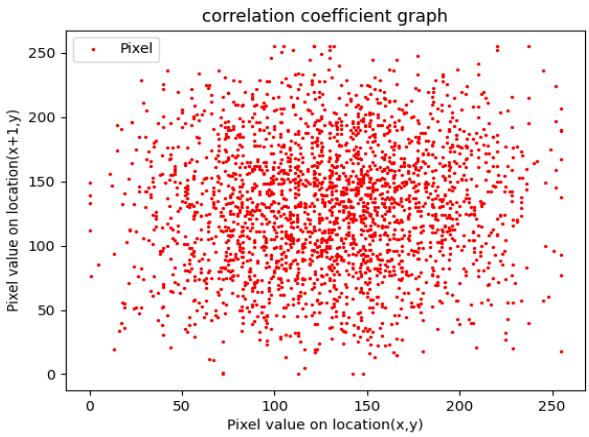
Coplot for Fig 1.1(Original Image)



Coplot for Fig 1.2(Encrypted Image)



Coplot for Fig 1.3(Original Image)



Coplot for Fig 1.4(Encrypted Image)

Differential Attack

The most important requirement for image encryption is that the image obtained after the encryption should be significantly different from the original image. This difference can be measured by two techniques namely NPCR and UACI. NPCR is number of changing pixel rate. Basically, it is the change rate of the encrypted image pixels when the image changes one pixel in the process of encryption. The larger the value of NPCR is, the stronger the resistance is of the algorithm to plaintext attacks. UACI is unified averaged changed frequency. Basically, it is the change rate of the average strength of the original image and the encrypted image. The larger the value UACI is, the stronger the resistance is of the algorithm to differential attacks. The formulas for calculating the values of NPCR and UACI are:

$$NPCR = \frac{\sum_{i,j} D(i,j)}{W * H} * 100\%$$

$$UACI = \frac{1}{W * H}\left[\sum_{i,j} \frac{|C(i,j) - C'(i,j)}{255}\right] * 100\%$$

Where H and W represents the height and width of the image, respectively. C and C$^{'}$ denote the encrypted images before and after one pixel of the image is changed. For the pixel at position (i,j), if C (i,j) !=C` (i,j), let D(i,j) = 1; else, let D(i,j) = 0. We calculated the NPCR, and UACI of the images, shown in table 5 and subsequently deduced that our algorithm is robust against differential attacks.

Image	**NPCR**	**UACI**
Fig 1 and Fig 2	99.59	30.46
Fig 3 and Fig 4	99.53	24.76

Table 5 – NPCR and UACI values

Conclusion

In this project, we have combined chaotic maps with DNA so that we can get better encryption. First, we generated a secret key by using the SHA-256 algorithm. Then, we shuffled the position of pixels by using chaotic maps. Then we encoded all pixels of the original image and the secret key into DNA sequences by converting all values into four DNA nucleotides. Next, we perform an XOR operation between DNA encoded pixels and DNA encoded secret key. Finally, we convert the resulted 2-d matrix back into binary form and get our encrypted image. Our experimental results show that our encryption is very good and very difficult to decode by any unauthorize person.

References

[1] Q. Zhang, et al., RGB color image encryption method based on Lorenz chaotic system and DNA computation, IETE Tech. Rev. 30 (5) (2013) 404–409.

[2] H. Liu and X. Wang, Image encryption using DNA complementary rule and chaotic maps, Appl. Soft Comput. 12 (2012), 1457–1466

[3] Huo Dongming, Zhou Ding-fu, Yuan Sheng, Yi Shaoliang, Zhang Luozhi, Zhou Xin, Image encryption using exclusive-OR with DNA complementary rules and double random phase encoding, Physics Letters A.

[4] Dania Saleem Malik, Tariq Shah, Color multiple image encryption scheme based on 3D-chaotic maps, Mathematics and Computers in Simulation 178 (2020) 646–666.

[5] A. Girdhar, et al., A RGB image encryption technique using Lorenz and Rossler chaotic system on DNA sequences, Multimedia Tools Appl. 77 (20) (2018) 27017–27039.

[6].Qiang Zhang, Ling Guo, Xiaopeng Wei, Image encryption using DNA addition combining with chaotic maps, Mathematical and Computer Modelling, Mathematic problems in Engineering Volume 2014.