KANTIPUR ENGINEERING COLLEGE

(Affiliated to Tribhuvan University)

Dhapakhel, Lalitpur

[Subject Code: CT 755]

A MAJOR PROJECT FINAL REPORT ON

IMAGE ENCRYPTION AND DECRYPTION USING CHAOS AND DNA CODING

Submitted by:

Abiral Chaulagain	**[38653]**
Anish Lal Joshi	**[38657]**
Grishan Pradhan	**[38676]**
Prajal Govinda Vaidya	**[38688]**

A MAJOR PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR IN COMPUTER ENGINEERING

Submitted to:

Department of Computer and Electronics Engineering

April, 2023

IMAGE ENCRYPTION AND DECRYPTION USING CHAOS AND DNA CODING

Submitted by:

Abiral Chaulagain	**[38653]**
Anish Lal Joshi	**[38657]**
Grishan Pradhan	**[38676]**
Prajal Govinda Vaidya	**[38688]**

Supervised by:

Dr. Babu Ram Dawadi

Asst. Professor, IOE Pulchowk Campus,

Tribhuvan University (TU)

A MAJOR PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR IN COMPUTER ENGINEERING

Submitted to:

Department of Computer and Electronics Engineering

Kantipur Engineering College

Dhapakhel, Lalitpur

April, 2023

ACKNOWLEDGMENT

We would like to express our deep and sincere gratitude to our project supervisor, Asst. Prof. Dr. Babu Ram Dawadi, Assistant Professor at IOE Pulchowk Campus, for providing continuous guidance. It was a great honour and pleasure to work under is supervision.

Moving on, we would like to acknowledge Er. Sahaj Shakya for his benevolent support and the effort made by Er. Bishal Thapa for guiding us the proper path to achieve goals that was once only an imagination. We can also not forget the platform and opportunity Kantipur Engineering College has provided us for our intellectual development.

Abiral Chaulagain	[38653]
Anish Lal Joshi	[38657]
Grishan Pradhan	[38676]
Prajal Govinda Vaidya	[38688]

ABSTRACT

In this digital world, encryption plays an important role in various domains. Securing the information is the main goal when storage and transfer of information takes place. Image encryption is a very important part of this as it applies to various domains like medical, multimedia, defence etc. In order to effectively protect digital image information, to improve the efficiency and security of the encrypted image, many effective and feasible image encryption algorithms have been proposed. Based on the study of digital image encryption technology, this paper proposes the color image encryption algorithm based on chaotic map system and DNA encoding. Chaos is used to shift the image's pixel positions, to achieve confusion and DNA encoding is used for changing the pixel values, to achieve diffusion. Chaotic systems show that it has greater resistance towards differential attacks, plain text attacks and noise and data loss attacks than the traditional block encryption approaches, such as AES, IDEA and DES.

Keywords – **Image encryption, Color image, Chaotic map, DNA coding, AES, IDEA, DES**

TABLE OF CONTENTS

Acknowledgment **i**

Abstract **ii**

List of Figures **vi**

List of Tables **vii**

List of Abbreviations **vii**

1 Introduction **1**

1.1 Background 1

1.2 Problem Statement 2

1.3 Objectives 2

1.4 Area of Application 3

1.4.1 Security 3

1.5 Project Features 4

1.6 Feasibility Study 4

1.6.1 Economic Feasibility 4

1.6.2 Schedule Feasibility 5

1.6.3 Operational Feasibility 5

1.6.4 Technical Feasibility 5

1.7 Project Requirements for Deployment 5

1.7.1 Hardware Specifications 5

1.7.2 Software Requirements 6

2 Theoretical Background and Literature Review **7**

2.1 DNA Coding 7

2.2 Chaos Theory 8

2.2.1 TD-ERCS 9

2.3 Related Works 11

3 Methodology **15**

3.1 Block Diagram 15

3.2 Algorithm 16

3.2.1 Key Generation 16

3.2.2 Image Encryption 17

3.2.3 Image Decryption 19
3.3 Use Case Diagram 21
3.4 Software Development Model 22
3.5 Activity Diagram 23
4 Result and Analysis **24**
4.1 Result 24
4.1.1 Original Image 24
4.1.2 Encrypted Image 25
4.1.3 Decrypted Image 26
4.2 Histogram Analysis 27
4.3 Mean Square Error and Peak Signal-To-Noise Ratio 28
4.3.1 MSE and PSNR between Original and Encrypted Images 28
4.3.2 MSE and PSNR between Original and Decrypted Images 29
4.4 Correlation Coefficient Analysis 29
4.5 Image Encryption and Decryption Time 32
4.6 Work Schedule 37
5 Conclusion and Future Enhancement **38**
5.1 Conclusion 38
5.2 Drawback 38
5.3 Future Enhancement 38
References **38**
Appendix **41**

LIST OF FIGURES

3.1 Block Diagram for encryption and decryption of an image 15
3.2 Use Case Diagram for the image encryption and decryption system . . . 21
3.3 Incremental Model for Software Development
[src: https://cs.ccsu.edu/ stan/classes/CS410/] 22
3.4 Activity Diagram for image encryption and decryption system 23
4.1 Original Image(Baboon) and its Histogram of RGB channels 24
4.2 Original Image(captured using smartphone) and it's Histogram of RGB channels 24
4.3 Original Chemical Plant(Far View) and its Histogram of RGB channels 25
4.4 Encrypted Image(Baboon) and Histogram of RGB channels of encrypted Baboon image 25
4.5 Encrypted Image(captured using smartphone) and it's Histogram of RGB channels 25
4.6 Encrypted Image(captured using smartphone) and it's Histogram of RGB channels 26
4.7 Decrypted Image(Baboon) and Histogram of RGB channels of decrypted Baboon image 26
4.8 Decrypted Image(captured using smartphone) and it's Histogram of RGB channels 26
4.9 Decrypted Image(Chemical Plant) and it's Histogram of RGB channels 27
4.10 Distribution of RGB values of two vertically adjacent pixels plain image (Baboon Image) 30
4.11 Distribution of RGB values of two vertically adjacent pixels cipher image (encrypted Baboon Image) 30
4.12 Distribution of RGB values of two horizontally adjacent pixels plain image (Baboon Image) 31
4.13 Distribution of RGB values of two horizontally adjacent pixels cipher image (encrypted Baboon Image) 31
4.14 Best fitted curve for number of pixels vs encryption time 33
4.15 Best fitted curve for number of pixels vs decryption time 34

4.16 Gantt Chart 37
5.1 Sample 1 (256x256) 41
5.2 Sample 2 (256x256) 41
5.3 Sample 3 (512x512) 42
5.4 Sample 4 (512x512) 42
5.5 Sample 5 (768x768) 43
5.6 Sample 6 (768x768) 43
5.7 Sample 7 (1024x1024) 44
5.8 Sample 8 (1024x1024) 45
5.9 Sample 9 (2250x2250) 46
5.10 Sample 10 (4032x3024) 47

LIST OF TABLES

1.1 Hardware Specifications 5

1.2 Software Requirements 6

2.1 DNA encoding rules 8

2.2 DNA XOR operations 8

4.1 MSE and PSNR values between Original and Encrypted Images 28

4.2 MSE and PSNR values between original and decrypted image. 29

4.3 Correlation Coefficients of horizontally adjacent pixels in original and cipher image 32

4.4 Correlation Coefficients of vertically adjacent pixels in original and cipher image 32

4.5 Encryption time(observed) 34

4.6 Encryption time (using equation 4.4) 35

4.7 Decryption time(observed) 35

4.8 Decryption time (using equation 4.5) 36

LIST OF ABBREVIATIONS

AES Advanced Encryption Standard

DES Data Encryption Standard

DNA Deoxyribonucleic Acid

IDEA International Data Encryption Algorithm

MSE Mean Square Error

NPCR Number of Pixels Change Rate

PRNG Pseudo Random Number Generator

PSNR Peak Signal to Noise Ratio

TD-ERCS Tangential Delay Ellipse Reflecting Cavity-map System

UACI Unified Average Change Intensity

CHAPTER 1
INTRODUCTION

1.1 Background

As technology is getting advanced day-by-day, an image is vastly used in all the domains. Hence, a powerful and an efficient cryptographic method is required for image storage and transmission. The existing techniques like DES and AES for encryption does not fulfil the requirements of image encryption as they have low performance measures in terms of efficiency and security.[1]

Digital images play a large role in multimedia communication, and thus, it is critical that appropriate mechanisms are in place to protect this information. Traditional block encryption approaches, such as AES, IDEA, and DES, can encrypt images effectively and ensure their safety to an extent. However, they cannot protect against transmission noise, which can be introduced to a digital image during the transmission process. [1]

With the rapid development of multimedia technology and network technology, digital image processing has been widely applied to all aspects of human life, such as remote sensing, industrial inspection, medical field, meteorology, communications, reconnaissance, and intelligent robots. As a result, increasing attention has been paid to image information. Additionally, it is more important to protect the security of image data, especially in military, commercial, and medical fields.[2]

It is almost impossible to completely avoid the eavesdropping while broadcasting data over internet and satellite communication. Hence, data needs to be secure especially multimedia data which originated from sensitive organizations like military, medical etc. Commonly, two ways of image encryption are found: optical and digital encryption. The former adopts optical instrument to build physical systems for image encryption which commonly relies on optics to randomize the frequency components in an image. The later one commonly takes advantage of a digital image and encrypts it either by an encryption algorithm in the form of software or a physical electronic device in the form of hardware. Most commonly employed encryption and security schemes

are digital in nature since present communication systems are becoming entirely digital. Thus among various digital image encryption techniques, the chaos-based image encryption method is believed to be good candidate for encryption purposes as chaotic systems are characterized by ergodicity, sensitive dependent on initial conditions and random like behaviors.

1.2 Problem Statement

Information security is an application-level technology used to protect any transmitted message against unnecessary revelation as well as secure it from unlawful modification while in transit over internet or storage. Image encryption is the transformation of an original image into an unrecognizable form of cipher-image.

With evolution of latest computer technology, the security of digital image data is seriously compromised. To meet the challenges of security, many traditional encryption algorithms has been proposed so far. But a large number of well-matured algorithms have been designed for text encryption only and it is not suitable for digital image encryption. Several distinctive properties such as bulk data volumes and high correlation among pixel values make text encryption algorithms impractical for an image data.

Traditional block encryption approaches, such as AES, IDEA, and DES, can encrypt images effectively and ensure their safety to an extent. However, they cannot protect against transmission noise, which can be introduced to a digital image during the transmission process. A range of chaos-based innovations have emerged including novel permutation techniques, new diffusion approaches, hyperchaotic map systems, simultaneous image encryption have been deciphered without knowing the secret keys. Logistic map has some drawbacks for information encryption, such as small key space, in-sufficient complexity, and low security, which allow easy decryption of the encryption schemes.[1]

1.3 Objectives

The objective of this project is: -

I. To create a system that secures the information content of images for storage

using chaos theory.

1.4 Area of Application

Cryptography is used for three broad purposes—to protect the confidentiality of information, to protect the integrity of information, and to authenticate the originator or sender of information. Applications that require the secrecy of large volumes of information use cryptography. Companies and organizations uses cryptography to enable the secure distribution of regular software updates, including security patches, over a network and is used to verify the identity of individuals and organizations. The increased availability and use of encryption—most notably to protect access to data stored on smart devices and to keep Internet messages, user passwords, IDs confidential. It is increasingly encountered in investigations by law enforcement and intelligence agencies.

1.4.1 Security

In multimedia sector, image encryption can be used to store or send or receive confidential data, maintaining integrity of the data or protecting image from unauthorized access. It also mainly protects the data from different types of attack that can affect the integrity of the information and maintains its validation such as:

Plaintext Attack

Plaintext attack is an attack performed on the plaintext(decrypted data) and the ciphertext(encrypted data) which the attacker may have received from a breach in security. After the attacker has received the plaintext, they may perform attacks such as Known Plaintext Attack and Chosen Plaintext attack, both of which might result in the attacker figuring out the key for the encryption.

Differential Attack

Differential attack is a sort of cryptanalysis that may be used to decrypt both block and stream ciphers, as well as cryptographic hash functions. It refers to a collection of

strategies for tracking differences across a network of transformations, finding where the cipher displays non-random behavior, and using such attributes to recover the secret key (cryptography key).

Noise and Data Loss Attacks

Noise attacks are those types of attacks that has no goals: it just distorts the quality of the image. Salt-and-pepper noise and Gaussian noise are the two types of image-noise attack. A data loss attack is an incident where part or whole information of the data is stolen or taken from a system without the knowledge or authorization of the system's owner.

1.5 Project Features

The features of this project are as follows: -

I. Very low inter-pixel redundancy
II. Greatly sensitive to initial condition due to use of chaotic map
III. Immune against statistical attacks

1.6 Feasibility Study

On the basis of outcome of initial investigation, feasibility takes place. The main goal of feasibility study is not to resolve the problem but to accomplish the scope. In the process of feasibility study, the cost and the benefits are estimated with greater accuracy.

1.6.1 Economic Feasibility

This project is economically feasible as the encryption algorithm can be implemented in any moderately powerful device without adding any extra hardware.

1.6.2 Schedule Feasibility

From our time analysis, we found out the system we are developing can be completed in scheduled time. We have almost 5-6 months to complete this project. And we four members have divided our work.

1.6.3 Operational Feasibility

The image encryption platform can be used by any user, even those who don't have the knowledge about image encryption at all. However, an engineer maybe required in case to setup the platform.

1.6.4 Technical Feasibility

This project is technically feasible as it provides a fast, low computational power demanding, low space consuming and efficient image encryption algorithm.

1.7 Project Requirements for Deployment

Following are the hardware and software requirements for our project:-

1.7.1 Hardware Specifications

Hardware requirements for the operation of this application are as follows:-

Table 1.1: Hardware Specifications

Minimum Requirement:	Intel Core 2 Duo Processor with 4 GB RAM
Recommended:	Intel i-series Processor with 8 GB RAM

1.7.2 Software Requirements

Below are the software requirements for the development and operation of this proposed system:-

Table 1.2: Software Requirements

Operating System:	Windows 8 or above, Linux based OS, Mac OS, Android, iOS
Programming Language :	Python

CHAPTER 2
THEORETICAL BACKGROUND AND LITERATURE REVIEW

2.1 DNA Coding

DNA is a biological material present in nearly all living organisms. Parents' DNA is transmitted to a child, thus causing the child to inherit the parents' features. In humans, DNA is shaped like a twisted ladder and contains four nucleobases: adenine (A), cytosine (C), guanine (G), and thymine (T). Watson and Crick published an article in 1953 [3], that presented the DNA complementary rule. This rule states that adenine(A) bonds with thymine(T) and that guanine(G) bonds with cytosine(C).

DNA coding uses the double helix structure of DNA. The basic element of DNA is nucleotide; for the different chemical structure it is divided into four parts: Adenine (A), Guanine (G), Cytosine (C) and Thymine (T).The DNA sequence is based on these four bases A, C, G and T where A and T are complement to each other and so are C and G. In DNA coding, information is expressed by this DNA sequence. Here, each base is represented by a two digit binary number. As in binary number system, 0 and 1 are complement to each other, we can use 00 and its complement 11 and 10 and 11 to express the 4 bases. Thus the total number of coding combinations is 4!=24. But out of these 24 combinations, only 8 combinations can be used because of the complementary relationship between the bases. Table 2.1 shows the 8 possible combinations of DNA-Binary Coding.[4]

Due to the rapid development of DNA computing, several biological and algebraic operations have been proposed for DNA sequences, such as the XOR operation. This operation is executed for a DNA sequence in the same way as a traditional XOR operation is executed in a binary sequence. Because there are eight DNA encoding rules, there are also eight DNA XOR rules. Table 2.2 presents the DNA XOR operations.[1]

Table 2.1: DNA encoding rules

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

Table 2.2: DNA XOR operations

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

2.2 Chaos Theory

Chaos theory is a method of qualitative and quantitative analysis to investigate the behavior of dynamic systems that cannot be explained and predicted by single data relationships, but must be explained and predicted by whole, continuous data relationships.

Chaos theory is the study of deterministic difference (differential) equations that display sensitive dependence upon initial conditions (SDIC) in such a way as to generate time paths that look random. Even though a small error in measurement of the system state leads to degradation of predictability at an exponential rate, and thus prediction by any method is futile in the long term, nonlinear prediction methods can do a good job on short term prediction when chaos or other non-linearity is present.[5]

Chaos theory is concerned with unpredictable courses of events. The irregular and unpredictable time evolution of many nonlinear and complex linear systems has been named chaos. Chaos is best illustrated by Lorentz' famous butterfly effect: the notion that a butterfly stirring the air in Hong Kong today can transform storm systems in New York next month. The definition of deterministic chaos implies that our prediction in the form of a model, for instance, is very sensitive to the initial conditions. The difference between predictions with slightly different initial conditions grows exponentially [6]

Chaotic systems are really disordered and chaos theory tries to find the underlying order

in sequence of random data. Nowadays, many researchers are observing several interesting relationships between chaotic behavior and Random Number Generators. As a matter of fact, many properties of chaotic systems like their sensitivity to initial conditions can be considered as a fundamental part of secret key generator in cryptographic algorithms[2].

The chaotic map used in this paper is:-

2.2.1 TD-ERCS

In 2004, researchers reported the pseudo-randomness of a two-dimensional chaotic system on a physical model of ellipse reflecting cavity, known as Tangent Delay Ellipse Reflecting Cavity Map System (TD-ERCS). TD-ERCS is two dimensional map and it can be used in the image permutation process i.e., row and column permutation respectively. Td-ERCS chaotic map fulfills certain criteria such as zero correlation in total field and equi-probability. As a result, chaotic random number generated through TD-ERCS exhibits higher security such as Sensitive to Initial Conditions (SDIC), ergodicity, deterministic Pseudo-randomness, and structural complexity.[7]

$$\begin{cases} x_n = -\frac{2k_{n-1}y_{n-1}+x_{n-1}(\mu^2-k_{n-1}^2)}{\mu^2+k_{n-1}^2} \\ k_n = \frac{2k'_{n-m}-k_{n-1}+k_{n-1}k'^2_{n-m}}{1+2k_{n-1}k'_{n-m}-k'^2_{n-m}} \qquad n = 1, 2, 3 \ldots \end{cases} \tag{2.1}$$

where

$$k'_{n-m} = \begin{cases} -\frac{x_{n-1}}{yn-1}\mu^2 & n < m \\ -\frac{x_{n-m}}{y_{n-m}}\mu^2 & n \geq m \end{cases} \tag{2.2}$$

$$y_n = k_{n-1}(x_n - x_{n-1}) + y_{n-1} \tag{2.3}$$

$$y_0 = \mu\sqrt{1 - x_0^2} \tag{2.4}$$

$$k'_0 = -\frac{x_o}{y_o}\mu \tag{2.5}$$

$$k_0 = -\frac{\tan\alpha + k'_0}{1 - k'_0 \tan\alpha} \tag{2.6}$$

where, μ, x_0, α and m are TD-ERCS seed parameters and $0 < \mu \leq 1$, $-1 \leq x_0 \leq 1$, and $0 < \alpha < \pi$.

The TD-ERCS system satisfies numerous criteria like zero correlation in total field, equiprobability distribution and positive Lyapunov exponent.The sequence generated using TD-ERCS system meet security characteristics like Sensitive Dependence on Initial Conditions (SDIC), ergodicity, deterministic preudo randomness and structure complexity due to which it is used extensively in encryption algorithms [8]

Due to advancement in distributed computer networks, storage devices, and imaging tools, the digital images have been extensively used in various fields. With the rapid advancement of communication technology, nowadays more and more text, image, audio, video and other multimedia files are transmitted over the internet. However, any type of data transmitted over a public channel such as the internet can be intercepted by an eavesdropper and hence become subject to security threats. Thus, means of secure transmission of data has become a necessity. One of the most important security issues is data confidentiality which can be addressed using a practical and secure encryption scheme [9]. The topic of cryptography has been studied extensively in the past years. Some famous ciphers such as DES, IDEA. AES and others were designed for the encryption of text or genreal binary sequences. They are not optimal ciphers for image encryption. This is because the large amount of image data leads to a long running time. The strong correlation among image pixels may lead to security problems in block ciphers. For enhancement of security level, many encryption algorithms were designed specially for image. Among them, chaos-based algorithms show pretty good performance because of the properties of ergodicity, quasi-randomness, sensitivity dependence on initial conditions, and system parameters. [10].

Sensitivity to initial conditions means that when a chaotic map is iteratively applied to two extremely close initial points, the iterates quickly diverge, and become uncorrelated in the long term. This character is especially useful in image encryption, since two adjacent pixels have high correlation but after several rounds of iterations they will be highly uncorrelated. For quite a long time, many now-classic schemes like DES

and IDEA emphasize more on substitution than on permutation, aiming at the key issue of security alone. Actually, permutation plus diffusion can also compose very good encryption schemes with not only high security but also fast speed. In fact, this observation led to some excellent encryption schemes based on two-dimensional chaotic maps, which were essentially motivated by this observation [11].

A range of chaos-based innovations have emerged in the field of chaos-based encryption, including novel permutation techniques, new diffusion approaches, hyper-chaotic map systems, a simultaneous encryption-compression scheme and novel transform domains. However, many of these schemes to encrypt image have been deciphered without knowing the secret keys [1]. At present, the discrete time chaotic systems, commonly used in information encryption, include Logistic map, Tent map, Henon Map and so on. However there are still some shortcomings in those chaotic systems for information encryption, such as small key space, insufficient complexity and low security, which causes the related encryption algorithms to be deciphered easily. [12]

2.3 Related Works

I. **A Color-Image Encryption Scheme Using a 2D Chaotic System and DNA Coding 2019**

To overcome the problem such as deciphering without knowing the secret keys, weak against known plaintext attacks, a new image-encryption approach that employs secure image sharing based on DNA-sequence operations and tangent-delay ellipse reflecting cavity-map system (TD-ERCS) maps is proposed. The TD-ERCS system is used as chaotic map for its complexity and has proved to be immune against differential attacks. The preliminary values for the TD-ERCS system are produced by calculating SHA-256 from a user's password. The encryption approach employs diffusion and confusion. The generated TD-ERCS system is used to permute the positions of the pixels in the original image, thereby confusing it. Subsequently, the permuted image is DNA encoded before the TD-ERCS system and DNA encoding are employed to generate a mask image. Next, DNA XOR is performed between the mask image and permuted image to gen-

erate the scrambled image. Finally, DNA decoding is applied to the scrambled image to generate an encrypted image and, in doing so, achieve diffusion. [1]

II. **A colour image encryption based on dna coding and chaotic sequences 2013**
In the proposed algorithm, the plain image is first scrambled using generalized Arnold Cat Map to achieve confusion. The scrambled image pixel are converted to DNA codes and again reconverted to integers where the choice of DNA coding rule is made pseudorandom based on the binary sequences generated by chaos based pseudorandom binary number generator. The integers thus obtained are diffused by performing exclusive OR operation with the integer sequences generated by 1D Logistic map producing the cipher image. The experimental results depicts that the proposed algorithm can successfully encrypt and decrypt RGB color images with secret keys. The simulation analysis also exhibit that the proposed method is loss-less, secure and efficient measured in terms of statistical tests(like histogram analysis, correlation coefficient analysis, measures of central tendency and dispersion), key sensitivity test, key space analysis, information entropy test, encryption quality by MSE, PSNR, NPCR and UACI.[4]

III. **TD-ERCS map-based confusion and diffusion of autocorrelated data 2017**
A new scheme is proposed to encrypt highly autocorrelated image pixel data. In existing literature, single substitution was used to break autocorrelation in images. To get better results, instead of single substitution box, some researchers are minimizing the autocorrelation via utilizing multiple substitution boxes. However, it is found that multiple substitution boxes cannot solve the problem of autocorrelation independently. Therefore, chaotic confusion and diffusion is added to the existing substitution scheme in order to remove the correlation completely. Initially, the autocorrelation of data is broken by permuting the pixels via TD-ERCS chaotic map. Bitwise XOR is carried out for diffusion using logistic map followed by the existing substitution scheme. Experimental analysis shows that the proposed scheme has greater resistance to brute force attack and statistical attack. Further analyses such as statistical analysis, peak signal-to-noise ratio, contrast analysis, key space analysis, entropy analysis, correlation analysis, key

sensitivity analysis, and encryption quality analysis proved the high robustness of the proposed scheme.[13]

IV. **Color Image Encryption Algorithm Based on Tangent-Delay Ellipse Reflecting Cavity Map System 2012**

Chaotic encryption has presently become the focus of the information security field. The article presents the main theory of TD-ERCS, and proposed a new algorithm to improve the security of image encryption based on TD-ERCS. According to the sort permutation and the binary XOR operation from the TD-ERCS chaotic series, it can achieve image encryption. This encryption algorithm is a reversible algorithm, and it can achieve the original image in the rule inverse process of the encryption algorithm. Finally, through computer simulation, the experiment results show that the new chaotic encryption algorithm based on TD-ERCS is valid and has higher security.

Tangent-Delay Ellipse Reflecting Cavity-map System (TD-ERCS) have many special properties such as the maximum Lyapunov exponent is over zero, unchangeable equiprobability distribution and zero correlation in the total field, and there is a square chaotic attractor when tangent delays one unit, and become ergodic state when tangent delays more units than one. The discrete chaotic systems can generate two independent pseudo-random sequences together. The behavior complexity of TD-ERCS system is high, and the complexity value changes a little with the change of the parameters of TD-ERCS system. TDERCS system is a discrete chaotic system with the steady complexity, and the pseudo-random sequences generated by TD-ERCS are suitable for use in information encryption.[14]

V. **Image Encryption Algorithm Based on DNA Encoding and Chaotic Maps**

DNA sequencing is the process used to map the nucleotide sequence forming a strand of DNA. Four bases, adenine (A), thymine (T), guanine (G), and cytosine (C) form the building blocks of genetic code. “A” binds with “T” and “G” binds with “C”. If we use the four deoxynucleotides “A,”,“T,” “G,” and “C” to represent the binary numbers “00,” “11,” “01,” and “10,” respectively, then each pixel can

be encoded into a string of nucleotides. Using only DNA encoding to encrypt images is not secure. Therefore, this paper combine chaos encryption technology and image encryption based on DNA computing to solve the hidden insecurity problems existing when images are confused using the chaotic encryption technology. In the paper the digital image pixels are confused using the chaotic encryption technology. Then the confused pixels are diffused using DNA encoding. The diffusion process is also applied to the chaotic encryption technology and, finally, the encryption result is obtained.[15]

CHAPTER 3
METHODOLOGY

3.1 Block Diagram

The working mechanism of system is shown below:-

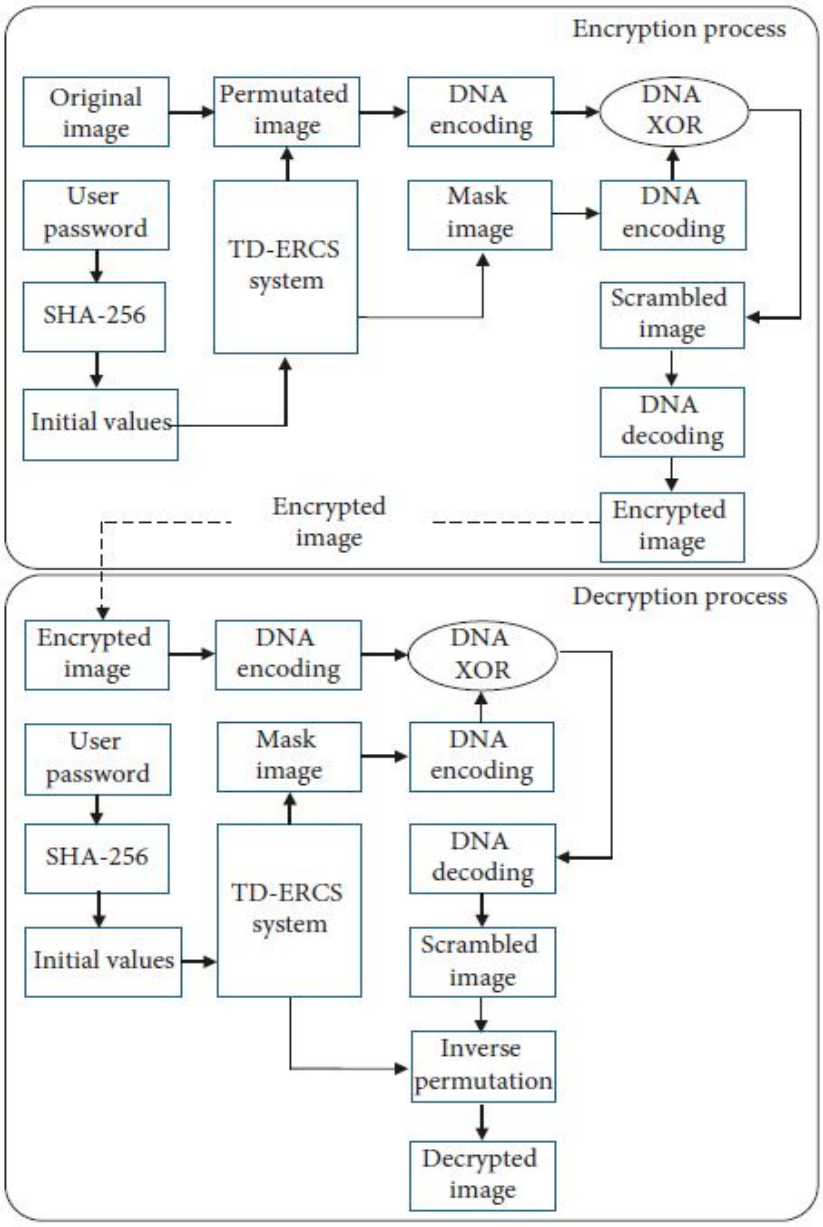


Figure 3.1: Block Diagram for encryption and decryption of an image
[src: https://www.researchgate.net/publication/337542764]

3.2 Algorithm

The algorithm for encryption and decryption involves generation of chaotic sequences using TD-ERCS. The chaotic sequences are used for pixel-position scrambling and creation of mask images which is DNA X-ORed with the pixel scrambled image for pixel substitution.

3.2.1 Key Generation

The proposed encryption scheme employs a user's password to generate secret keys. SHA-256 is used to generate a message digest of 256-bits for the password. The 64 hash values of the message digest are then divided into eight hexadecimal groups: k1, k2, . . ., k8. Each group contains eight hexadecimal values and is converted into a floating decimal number via the following equation:

$$d_j = \frac{hex2dec(k_1, k_2, ..., k_8)}{2^{10}} \tag{3.1}$$

where, j = 1, 2, 3, ... 8. The first set of the TD-ERCS initial values, (μ, x_0, α and m) for the TD-ERCS system is calculated by:

$$x_0 = sign(d_1 - d_2) \times d_1 \, mod \, 1, \tag{3.2}$$

$$\mu = d_2 \, mod \, 1, \tag{3.3}$$

$$\alpha = d_3 \, mod \, \pi, \tag{3.4}$$

$$m = d_4 \, mod \, 10 \tag{3.5}$$

The second set of initial values (μ, x_0,α and m) for the TD-ERCS system is calculated by:

$$x_0 = sign(d_1 - d_2) \times d_5 \; mod \; 1, \tag{3.6}$$

$$\mu = d_6 \; mod \; 1, \tag{3.7}$$

$$\alpha = d_7 \; mod \; \pi, \tag{3.8}$$

$$m = d_8 \; mod \; 10 \tag{3.9}$$

3.2.2 Image Encryption

The process of encryption involves the following steps:

1. Conversion of the original color image to three matrices: R(m, n), G(m, n), and B(m,n).
2. Check if R, G and B matrices are same or not by comparing corresponding values of each position of R, G and B matrices.

   If (R,G and B matrices are identical to each other):

   The image is grayscale.

   Else:

   The image is color image.
3. Generation of a pair of chaotic sequences, $x_{new} = (x_1, x_2, ..., x_m)$ and $y_{new} = (y_1, y_2, ..., y_n)$, via a TD-ERCS chaotic map. This uses the starting values μ, x_0, α and m as detailed in 3.2.1
4. Arrange the sequences x_{new} and y_{new} in ascending order and record their locations:-

   $$Index\ 1 = \{i_1, i_2, ..., i_m\}$$

   $$Index\ 2 = \{j_1, j_2, ..., j_n\}$$

   The objective of this step is to find the index of the smallest number from the sequence with size m and then store it in i_1, second smallest to i_2. Similarly, we can obtain index array Index 2 by arranging the sequence y_{new} in ascending order and taking its index.[15]

5. Using the Index 1 we shuffle the pixels row-wise and column-wise pixel shuffling is done using Index 2. This is done for all the matrices R,G and B from which we get R_{new}, G_{new} and B_{new}
6. Conversion of R_{new} , G_{new} , $and\ B_{new}$ to binary matrices. DNA encoding is then employed to encode these binary matrices, as detailed in table 2.1.
7. Generation of three chaotic sequences $x_n = (x_1, x_2, ..., x_{mn})$ and $y_n = (y_1, y_2, ..., y_{mn})$, and $z_n = (z_1, z_2, ..., z_{mn})$ via a TD-ERCS chaotic map. This uses the starting values of the second set,as discussed in 3.2.1(μ, x_0,α and m).Then, these generated values are converted into a range from 0 to 256 according to:

$$x_e(k) = round(abs(x_i(k)) \ \times \ 1000 \ mod \ 256) \ ,$$

$$y_e(k) = round(abs(y_i(k)) \ \times \ 500 \ \ mod \ 256) \ ,$$

$$z_e(k) = round(abs(z_i(k)) \ \times \ 1000 \ \ mod \ 256) \ ,$$

   where k ranges from 0 to M × N. Thus, the mask image is generated.
8. Conversion of $x_e,\ y_e\ and\ z_e$ to binary matrices. DNA encoding is then employed to encode these matrices. This results in three remodeled coding matrices: $x_{dna},\ y_{dna},\ and\ z_{dna}$ sized (m, n×4).
9. (a) For color image:
   Execution of the DNA XOR operation among ($R_{dna}\ and\ x_{dna}$) , ($G_{dna}\ and\ y_{dna}$) , and ($B_{dna}\ and\ z_{dna}$) . This results in three encrypted matrices: R_c, G_c, B_c.
   (b) For grayscale image:
   Execution of the DNA XOR operation among ($R_{dna}\ and\ x_{dna}$) , ($G_{dna}\ and\ x_{dna}$) , and ($B_{dna}\ and\ x_{dna}$) . This results in three encrypted matrices: R_c, G_c, B_c.
10. Generation of the final encrypted image by performing a DNA decoding operation for $R_c,\ G_c,\ and\ B_c$. This results in three new matrices with values ranging from 0 to 255: $R_{enc},\ G_{enc},\ and\ B_{enc}$.

The three matrices $R_{enc},\ G_{enc},\ and\ B_{enc}$ are combined to get the encrypted image.

3.2.3 Image Decryption

The process of decryption involves the following steps:

1. Conversion of the encrypted color image to three matrices: R(m, n), G(m, n), and B(m,n).
2. Check if R, G and B matrices are same or not by comparing corresponding values of each position of R, G and B matrices.

   If (R,G and B matrices are identical to each other):

   The image is grayscale.

   Else:

   The image is color image.
3. Conversion of R, G, and B to binary matrices. DNA encoding is then employed to encode these binary matrices, as detailed in table 2.1. This results in three remodeled coding matrices: $R_{dna},\ G_{dna},\ and\ B_{dna}$ sized (m,n$\times 4$).

4. Generation of three chaotic sequences $x_i = (x_1, x_2, ..., x_{mn})$ and $y_i = (y_1, y_2, ..., y_{mn})$, and $z_i = (z_1, z_2, ..., z_{mn})$ via a TD-ERCS chaotic map. This employs the starting values of the second set, as discussed in 3.2.1μ, x_0,α and m. The generated values are then converted into range of 0 to 256 according to:

$$x_e(k) = round(abs(x_i(k)) \times 1000 \mod 256),$$
$$y_e(k) = round(abs(y_i(k)) \times 500 \mod 256),$$
$$z_e(k) = round(abs(z_i(k)) \times 1000 \mod 256),$$

   where k ranges from 0 to M × N. Thus, the mask image is generated.
5. Conversion of $x_e,\ y_e\ and\ z_e$ to binary matrices. Then, DNA encoding is employed to encode these matrices. This results in three remodeled coding matrices: $x_{dna},\ y_{dna},\ and\ z_{dna}$ sized (m,n$\times 4$)
6. (a) For color image:

   Execution of the DNA XOR operation among ($R_{dna}\ and\ x_{dna}$), ($G_{dna}\ and\ y_{dna}$), and ($B_{dna}\ and\ z_{dna}$). This results in three encrypted matrices: R_d, G_d, B_d.

   (b) For grayscale image:

Execution of the DNA XOR operation among (R$_{dna}$ and x_{dna}) , (G$_{dna}$ and x_{dna}) , and (B$_{dna}$ and x_{dna}) . This results in three encrypted matrices: R$_d$, G_d, B_d.

7. Generation of a pair of chaotic sequences, x$_{new}$ $= (x_1, x_2, ..., x_m$) and y$_{new}$ $= (y_1, y_2, ..., y_n$), via a TD-ERCS chaotic map. This uses the starting values μ, x_0,α and m as detailed in 3.2.1
8. Arrange the sequences x_{new} and y_{new} in ascending order and record their locations[15]:-

$$Index\ 1 = \{i_1, i_2, ..., i_m\}$$
$$Index\ 2 = \{j_1, j_2, ..., j_n\}$$

   The index of the smallest number from the sequence with size m is stored in i_1, second smallest to i_2 and so on. Similarly, we can obtain index array Index 2 by arranging the sequence y_{new} in ascending order and taking its index.
9. Using the Index 1 we unshuffle the pixels row-wise and column-wise pixel unshuffling is done using Index 2. This is done for all the matrices R_d,G_d and B_d, from which we get R_{dec}, G_{dec} and B_{dec}
10. The matrices R_{dec}, G_{dec} and B_{dec} are DNA decoded from which we get R,G and B matrices.

Then, the final decrypted image is constructed by recombining these three matrices: R, G and B into a color image.

3.3 Use Case Diagram

The user can use the system for image encryption and decryption. To do so they need to select an image and enter the password. After the encryption or decryption process completes, the output is displayed to the user. The Use case Diagram figure 3.2 show the high-level functions and scope of the system.

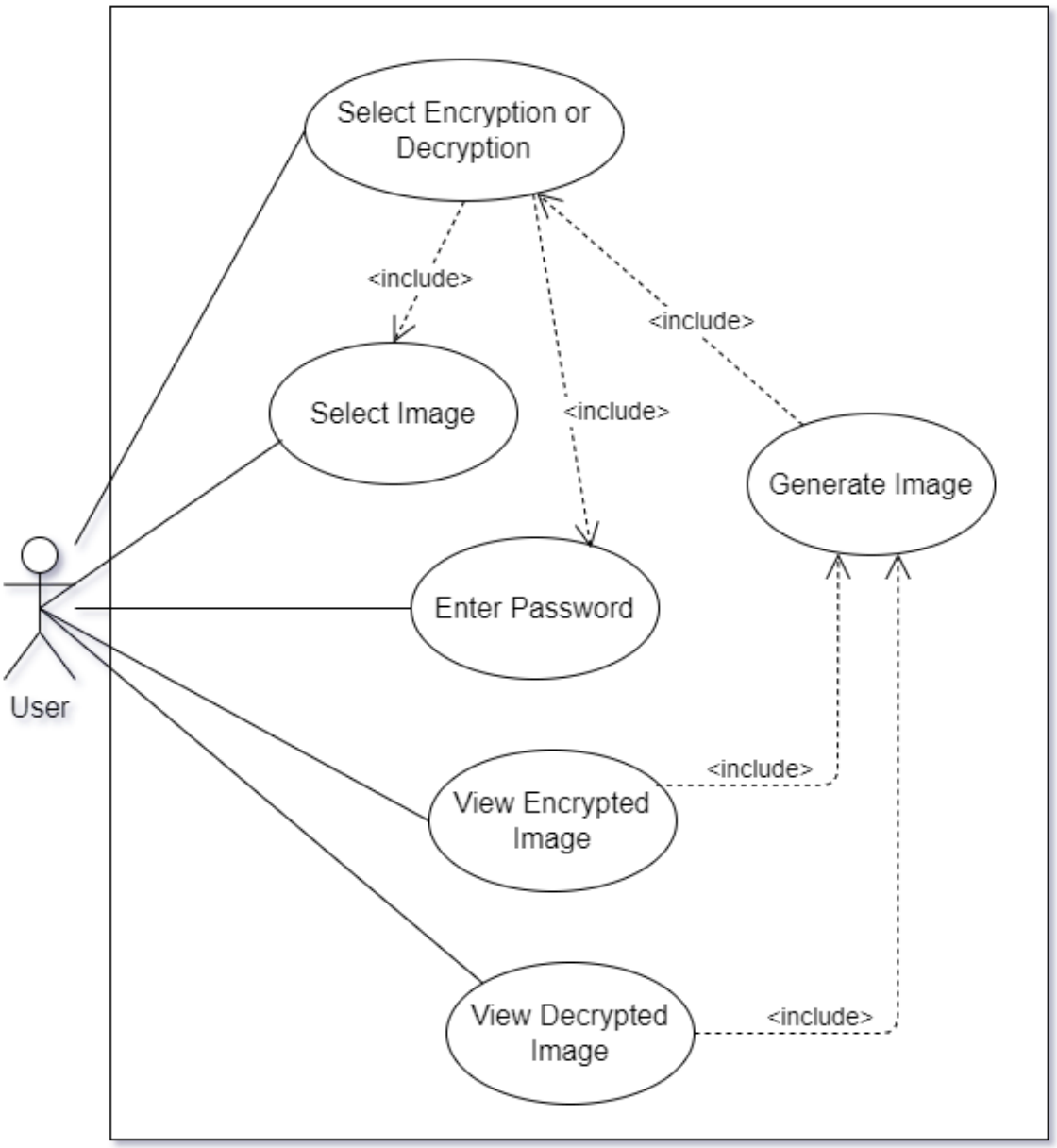


Figure 3.2: Use Case Diagram for the image encryption and decryption system

3.4 Software Development Model

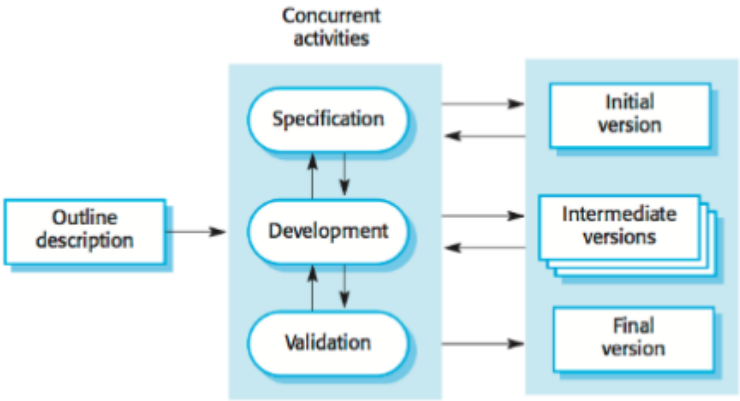


Figure 3.3: Incremental Model for Software Development
[src: https://cs.ccsu.edu/ stan/classes/CS410/]

Incremental development is based on the idea of developing an initial implementation, exposing this to user comment and evolving it through several versions until an adequate system has been developed. Specification, development, and validation activities are interleaved rather than separate, with rapid feedback across activities.

Incremental development reflects the way that we solve the problems. We rarely work out a complete problem solution in advance but move toward a solution in a series of steps, backtracking when we realize that we have made a mistake. By using incremental model, it is easier to make changes in the software as it is being developed.

3.5 Activity Diagram

The figure 3.4 is the activity diagram which represents the sequence of actions within the system for encryption and decryption.

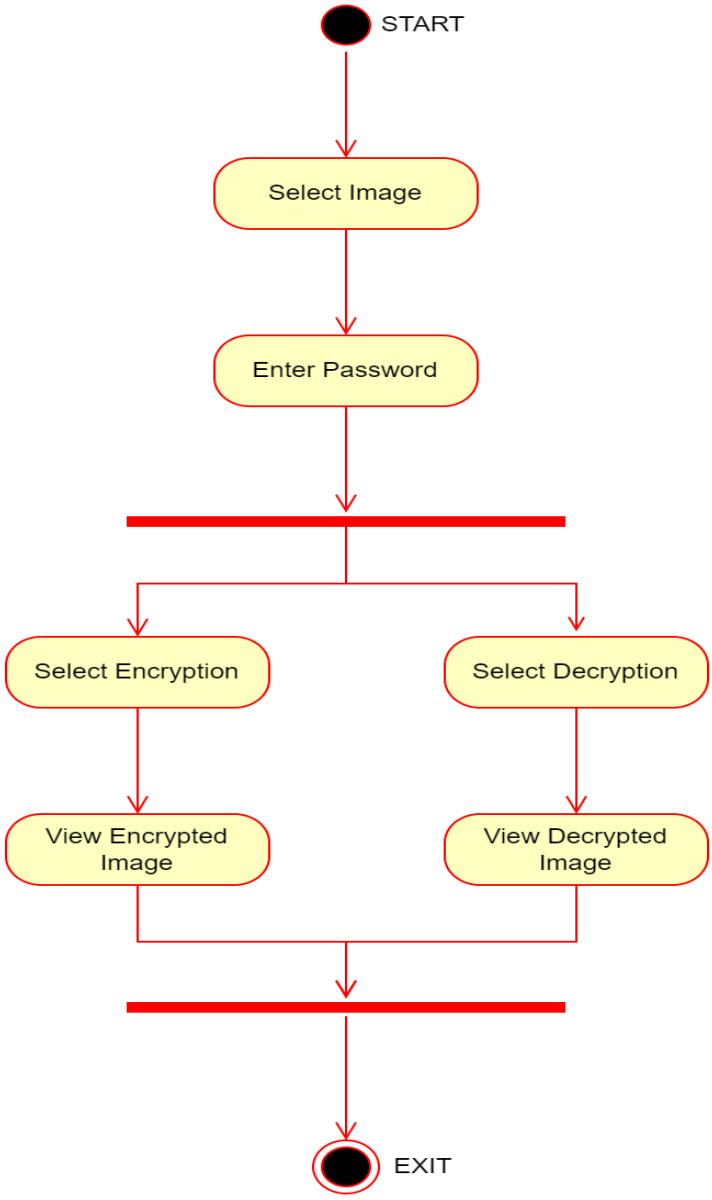


Figure 3.4: Activity Diagram for image encryption and decryption system

CHAPTER 4
RESULT AND ANALYSIS

Images from the dataset[16], which contains color and grayscale images and a image captured using smartphone are used for the analysis. The images are encrypted and decrypted using the algorithm explained in section 3.2 implemented to code in python programming language. The histograms of RGB channels and distribution of RGB values of adjacent pixels of images are plotted using matplotlib library.

Section 4.1 contains the results for the original, encrypted and decrypted images. And the following sections in this chapter contains the analysis of the results.

4.1 Result

4.1.1 Original Image

This is the original image, from USC-SIPI database[16].

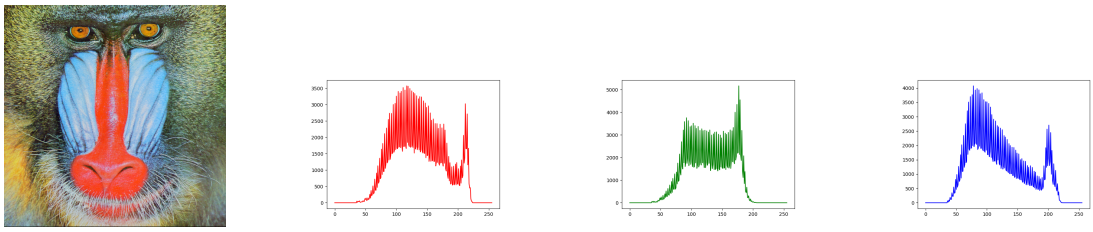


Figure 4.1: Original Image(Baboon) and its Histogram of RGB channels

This is the image captured using smartphone.

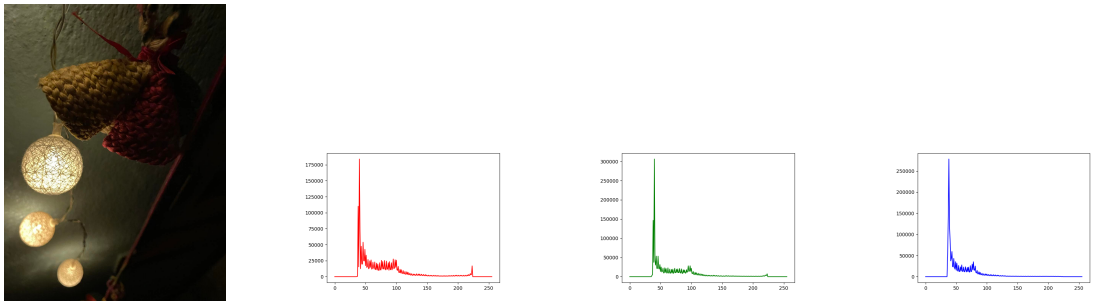


Figure 4.2: Original Image(captured using smartphone) and it's Histogram of RGB channels

This is the grayscale image from USC-SIPI database[16].

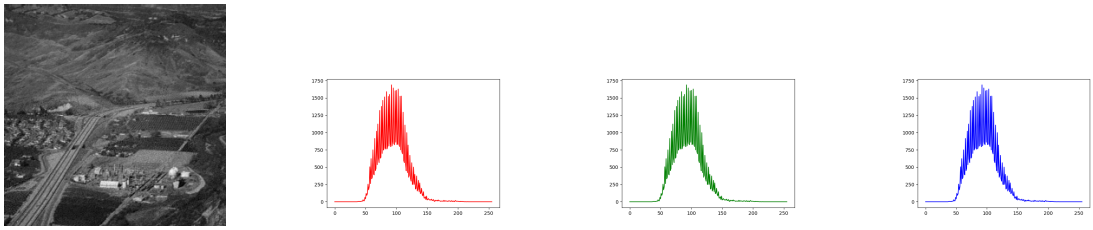


Figure 4.3: Original Chemical Plant(Far View) and its Histogram of RGB channels

4.1.2 Encrypted Image

After encryption of figure 4.1:-

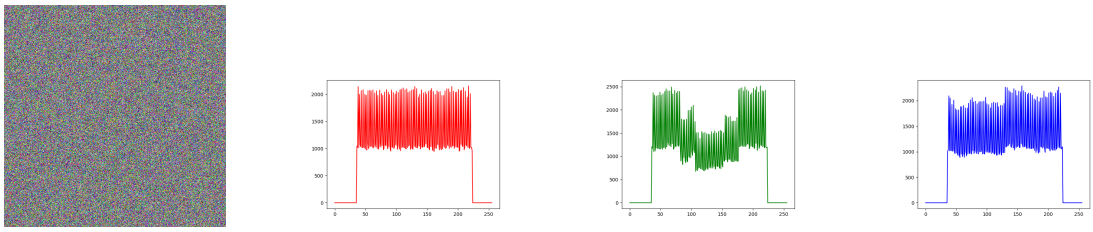


Figure 4.4: Encrypted Image(Baboon) and Histogram of RGB channels of encrypted Baboon image

After encryption of figure 4.2:-

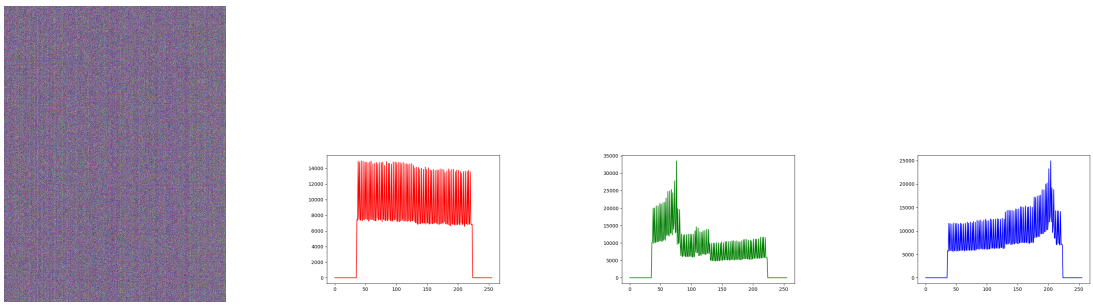


Figure 4.5: Encrypted Image(captured using smartphone) and it's Histogram of RGB channels

After encryption of figure 4.3:-

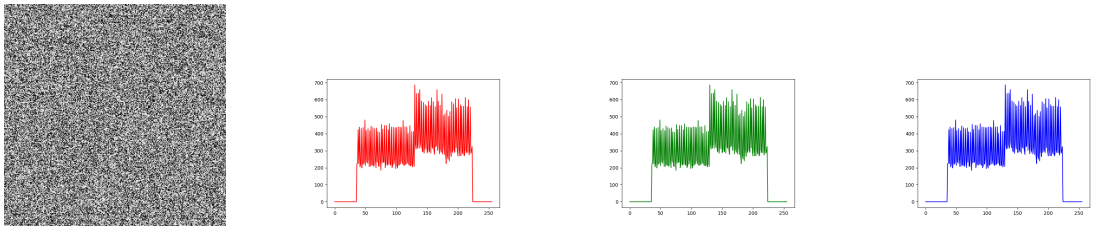


Figure 4.6: Encrypted Image(captured using smartphone) and it's Histogram of RGB channels

4.1.3 Decrypted Image

After decryption of figure 4.4:-

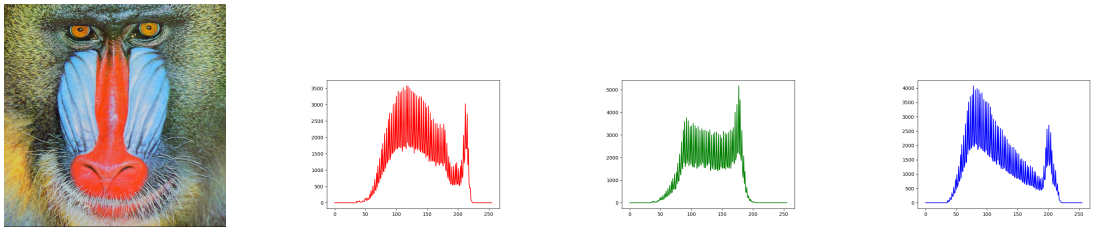


Figure 4.7: Decrypted Image(Baboon) and Histogram of RGB channels of decrypted Baboon image

After decryption of figure 4.5:-

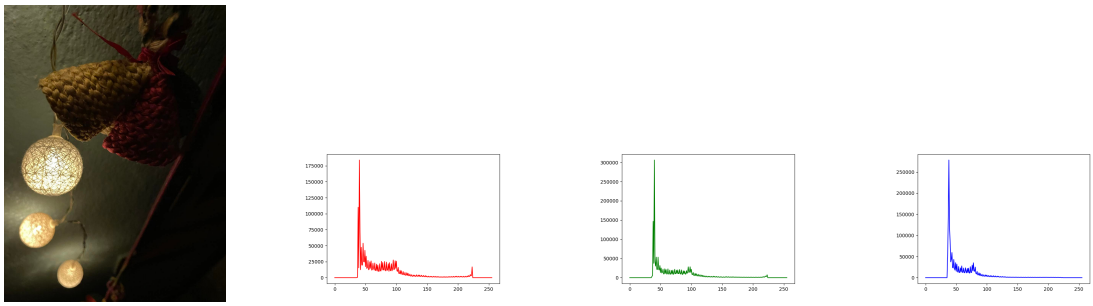


Figure 4.8: Decrypted Image(captured using smartphone) and it's Histogram of RGB channels

After decryption of figure 4.6:-

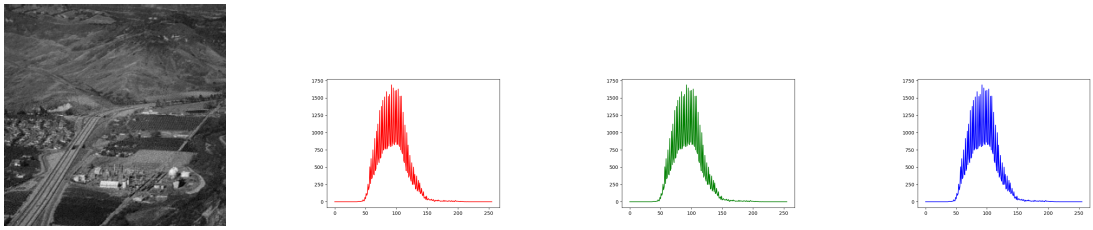


Figure 4.9: Decrypted Image(Chemical Plant) and it's Histogram of RGB channels

4.2 Histogram Analysis

Histograms of an image are used to represent the pixel distribution for each color intensity level. An ideal image-encryption algorithm produces a histogram of the encrypted image with a uniform distribution. The histograms of the original and encrypted image (Baboon) are displayed in figure 4.1 and figure 4.4 for the red, green, and blue components, respectively. It should be noted that the histograms of encrypted images are flat and thus differ from the original images. Same is the case for image captured using smartphone 4.2 and it's encrypted version 4.5.Therefore, the proposed encryption scheme can resist statistical attacks.

Figure 4.7 is the image decrypted from figure 4.4. It has the same histogram as that of the histogram of original image 4.1 and the histogram of image captured using smartphone is same for both original 4.2 and decrypted 4.8. For the photo captured using smart phone the histogram is changed and is more uniform for the encrypted image 4.5

From figure 4.3, we can see that the histograms of R,G and B are identical, which is the case for grayscale images, it is because gray values in corresponding pixel position of R,G and B matrices are same. Figure 4.6 is the result obtained after the encryption of 4.3. The R,G and B histograms are identical here too because of the identical operation performed in each of the R,G and B matrices.

It is observed that the histogram is uniform for the encrypted images which proves the encryption scheme can resist statistical attacks.

4.3 Mean Square Error and Peak Signal-To-Noise Ratio

The peak signal-to-noise ratio (PSNR) and mean-square error (MSE) are two error measures that are widely used for comparing image quality. The MSE refers to the cumulative square error between the original image and encrypted image and is calculated using equation 4.1. The higher the value of the MSE, the higher the probability that the encrypted images are distorted and noisy. The PSNR represents the noise ratio between the original image and the encrypted image and is measured in decibels (dB) calculated using equation 4.2 Low PSNR values indicate that the encryption scheme results in high degradation of the encrypted images.[1]

$$MSE = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} [x(i,j) - y(i,j)]^2 \tag{4.1}$$

$$PSNR = 10log_{10}[\frac{(I_{max})^2}{MSE}] \tag{4.2}$$

4.3.1 MSE and PSNR between Original and Encrypted Images

Table 4.1: MSE and PSNR values between Original and Encrypted Images

Image	MSE	PSNR
Baboon	8889.5857	8.6419
Female	9599.9875	8.3080
House	9364.1645	8.4161
Peppers	10337.0734	7.9868
Splash	11419.8928	7.5541
Chemical Plant	9417.4622	8.3914
Image captured from smartphone	14784.7078	6.4326

From the table 4.1, it can be noted that the MSE between encrypted and decrypted images are high whereas PSNR is low for all the images whether it is the image from dataset(color or grayscale) or image captured using smartphone. This shows the encryption scheme produces cipher image with high degradation.

4.3.2 MSE and PSNR between Original and Decrypted Images

Table 4.2: MSE and PSNR values between original and decrypted image.

Image	MSE	PSNR
Baboon	0.0	∞
Female	0.0	∞
House	0.0	∞
Peppers	0.0	∞
Splash	0.0	∞
Chemical Plant	0.0	∞
Image captured from smartphone	0.0	∞

From the table 4.2, it can be noted that the MSE between original and decrypted images are 0.0, which leads PSNR to inifinity for all the images whether it is the image from dataset(color or grayscale) or image captured using smartphone. This shows the decryption can recover the original image from the ecnrypted image without any change.

4.4 Correlation Coefficient Analysis

RGB distribution of Baboon image before and after encryption is given below. RGB distribution graph represents the intensity of different color channels in the image. The figure 4.10 shows RGB distribution of vertically adjacent pixels and figure 4.12 shows the RGB distribution of horizontally adjacent pixels of the Baboon image. These two graphs shows almost linear relation between pixel intensity and number of pixels and which shows high correlation among the pixels. Whereas, the figure 4.11 and figure 4.13 which are RGB distribution of encrypted Baboon image of vertically adjacent and horizontally adjacent pixels shows that the pixels are distributed randomly and are uncorrelated. The correlation coefficients are calculated using equation 4.3

$$\begin{cases} r_{xy} = \frac{|cov(x,y)|}{\sqrt{D(x)} \times \sqrt{D(y)}} \\ Cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \\ E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \\ D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2 \end{cases} \tag{4.3}$$

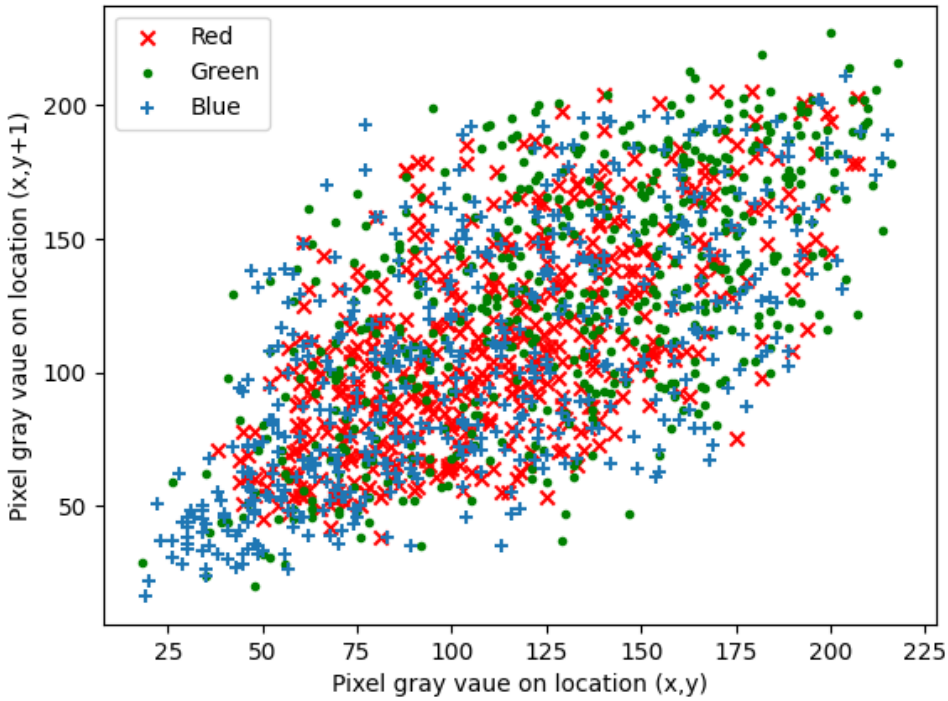


Figure 4.10: Distribution of RGB values of two vertically adjacent pixels plain image (Baboon Image)

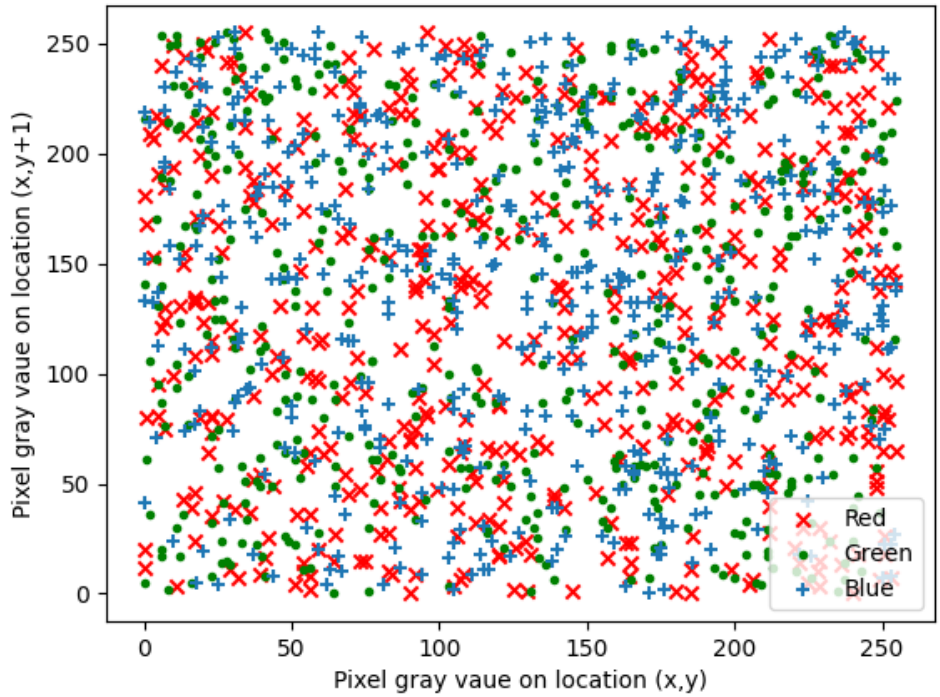


Figure 4.11: Distribution of RGB values of two vertically adjacent pixels cipher image (encrypted Baboon Image)

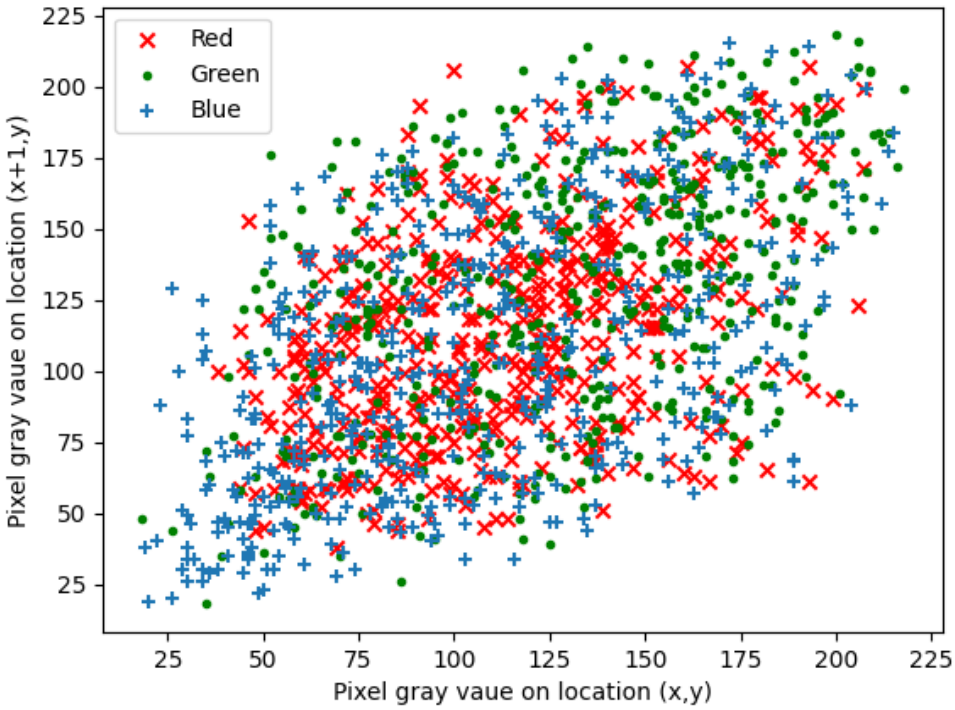


Figure 4.12: Distribution of RGB values of two horizontally adjacent pixels plain image (Baboon Image)

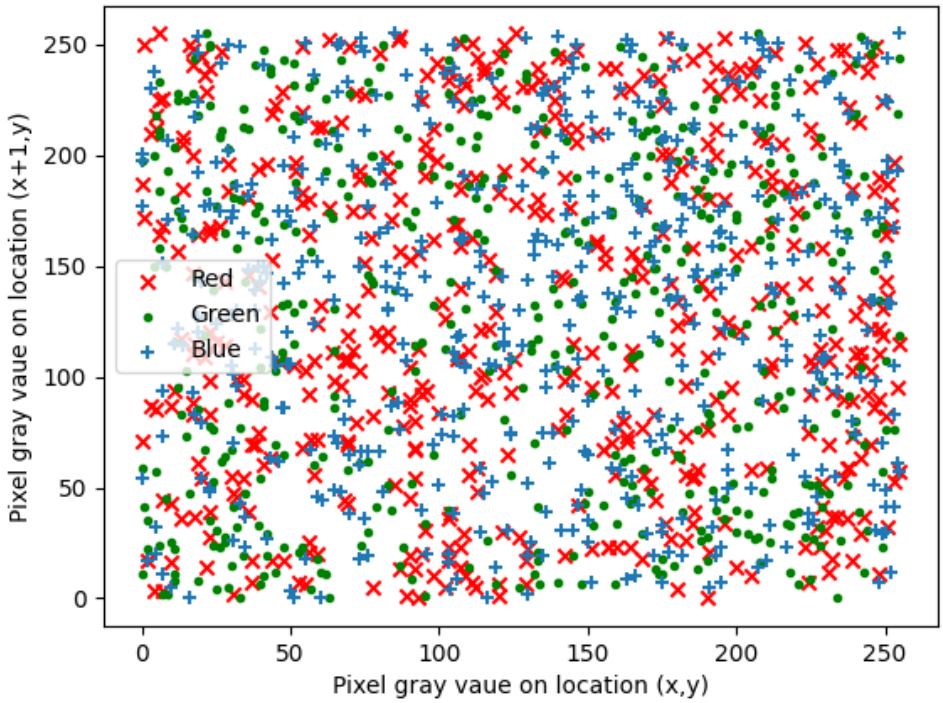


Figure 4.13: Distribution of RGB values of two horizontally adjacent pixels cipher image (encrypted Baboon Image)

Table 4.3: Correlation Coefficients of horizontally adjacent pixels in original and cipher image

Image Name	**Original Image**			**Cipher Image**		
	R	**G**	**B**	**R**	**G**	**B**
Baboon	*0.888*	*0.845*	*0.925*	*-0.044*	*0.077*	*0.008*
Female	*0.983*	*0.975*	*0.961*	*0.207*	*0.351*	*0.116*
House	*0.855*	*0.861*	*0.936*	*0.018*	*0.020*	*0.031*
Peppers	*0.966*	*0.981*	*0.963*	*-0.017*	*0.217*	*0.008*
Splash	*0.896*	*0.927*	*0.849*	*0.057*	*-0.026*	*-0.003*
Chemical Plant	*0.8305*	*0.8305*	*0.8305*	*-0.021*	*-0.021*	*-0.021*
Image captured using smartphone	*0.999*	*0.999*	*0.998*	*0.017*	*-0.046*	*0.012*

Table 4.4: Correlation Coefficients of vertically adjacent pixels in original and cipher image

Image Name	**Original Image**			**Cipher Image**		
	R	**G**	**B**	**R**	**G**	**B**
Baboon	*0.360*	*0.192*	*0.268*	*0.011*	*-0.007*	*0.062*
Female	*0.790*	*0.801*	*0.813*	*0.031*	*0.263*	*-0.034*
House	*0.725*	*0.878*	*0.926*	*0.030*	*-0.019*	*0.044*
Peppers	*0.934*	*0.947*	*0.951*	*0.038*	*-0.078*	*-0.101*
Splash	*0.955*	*0.920*	*0.924*	*-0.100*	*-0.038*	*-0.007*
Chemical Plant	*0.5557*	*0.5557*	*0.5557*	*-0.0084*	*-0.0084*	*-0.0084*
Image captured using smartphone	*0.994*	*0.994*	*0.991*	*0.026*	*0.052*	*-0.033*

The table 4.3 and 4.4 shows that correlation coefficient of R,G and B values of the original image is high but after the encryption it is close to zero or negative, which indicates no correlation between adjacent image pixels. Low to no correlation between pixels means the encrypted image is more random and highly unpredictable, which makes it more difficult analyze and decrypt the encrypted image.

4.5 Image Encryption and Decryption Time

The time it took to encrypt and decrypt images of different size(or number of pixels) was observed as in the table 4.5 and table 4.7. Regression equation was created for number of pixels as independent variable and time for encryption as dependent variable, same was done for decryption.

The regression equation for encryption is:-

$$y = 0.00000317647x + 0.07015 \tag{4.4}$$

where,

y=encryption time in minutes

x=number of pixels in the image

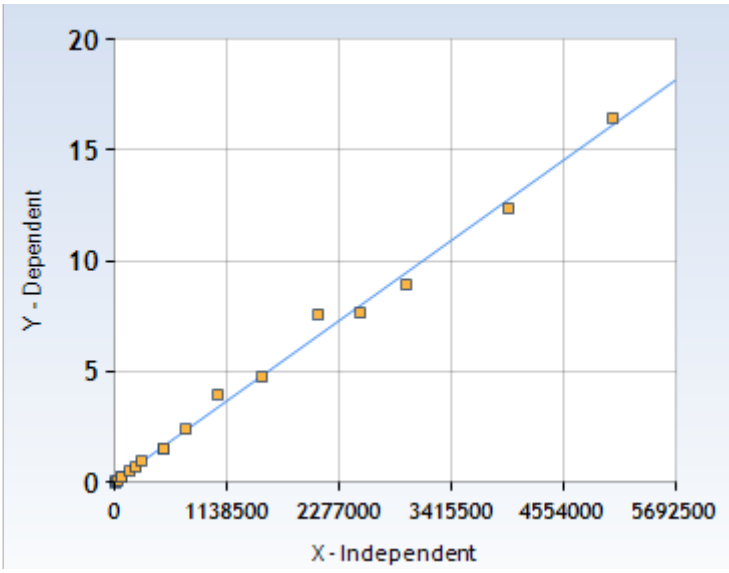


Figure 4.14: Best fitted curve for number of pixels vs encryption time

The regression equation for decryption is:-

$$y = 0.00000316398x + 0.0841 \tag{4.5}$$

where,

y=decryption time in minutes

x=number of pixels in the image

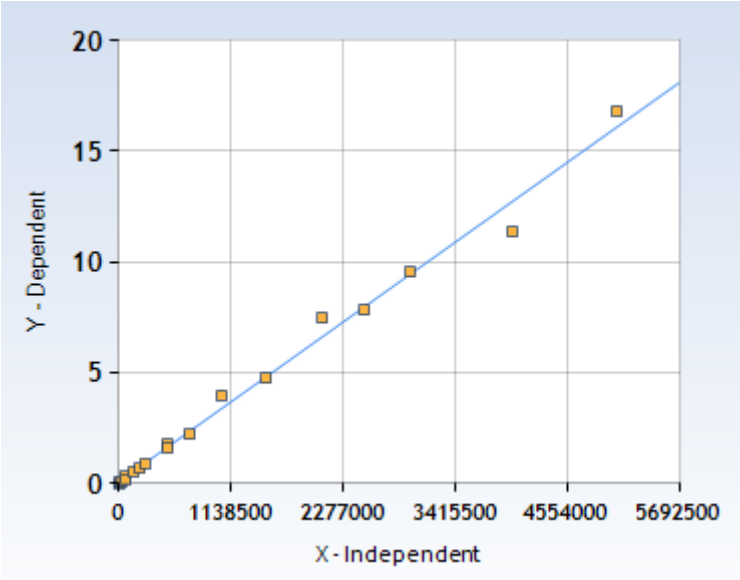


Figure 4.15: Best fitted curve for number of pixels vs decryption time

Table 4.5: Encryption time(observed)

Image Size	**Observed**	**Using Regression y=0.00000317647x+0.07015**	**Error**
Total pixels	**Encryption Time (min.)**	**Encryption Time (min.)**	**Encryption**
2500	0.016667	0.07809	-368.54%
10000	0.03333	0.10191	-205.77%
30000	0.1	0.16544	-65.44%
65536	0.233	0.27832	-19.45%
65536	0.216	0.27832	-28.85%
150000	0.51667	0.54662	-5.80%
220000	0.71667	0.76897	-7.30%
262144	0.95	0.90284	4.96%
490000	1.55	1.62662	-4.94%
500000	1.55	1.65839	-6.99%
722500	2.3833	2.36515	0.76%
1048576	4	3.40092	14.98%
1500000	4.75	4.83486	-1.79%
2073600	7.6	6.65688	12.41%
2500000	7.65	8.01133	-4.72%
2962440	8.95	9.48025	-5.92%
4000000	12.38333	12.77603	-3.17%
5062500	16.483	16.15103	2.01%

Using equation 4.4 encryption time of a couple of image was calculated. The regression equation calculated the time with the following accuracy as shown in table 4.6

Table 4.6: Encryption time (using equation 4.4)

Image Size	**Using Regression y=0.00000317647x+0.07015**	**Observed**	**Error**
Total pixels	**Encryption Time (min.)**	**Encryption Time (min.)**	**Encryption**
11000000	35.01132	34.98	-0.09%
12192768	38.80011	41.566	6.65%

From table 4.6, it can be observed that the error between predicted and observed value of time is low. Also the error in prediction of encryption time of images using which the regression equation was constructed, can be observed in table 4.5, which is low for most cases. Hence, the equation 4.4 it can predict the encryption time of image of any size.

Table 4.7: Decryption time(observed)

Image Size	**Observed**	**Using Regression y=0.00000316398x+0.0841**	**Error**
Total pixels	**Decryption Time (min.)**	**Decryption Time (min.)**	**Decryption**
2500	0	0.09201	-100%
10000	0.03333	0.11574	-71.20%
30000	0.1	0.17902	-44.14%
65536	0.3	0.29145	2.93%
65536	0.2	0.29145	-31.38%
150000	0.55	0.55870	-1.56%
220000	0.66667	0.78018	-14.55%
262144	0.9	0.0.91352	-1.48%
490000	1.78	1.63445	8.91%
500000	1.5833	1.66609	-4.97%
722500	2.2	2.37008	-7.18%
1048576	4	3.40177	17.59%
1500000	4.73333	4.83007	-2.00%
2073600	7.467	6.64493	12.37%
2500000	7.8333	7.99405	-2.01%
2962440	9.533	9.45720	0.80%
4000000	11.36667	12.74002	-10.78%
5062500	16.816	16.10175	4.44%

Using equation 4.5 decryption time of a couple of image was calculated. The regression equation calculated the time with the following accuracy as shown in table 4.8

Table 4.8: Decryption time (using equation 4.5)

Image Size	**Using Regression y=0.00000316398x+0.0841**	**Observed**	**Error**
Total pixels	**Decryption Time (min.)**	**Decryption Time (min.)**	**Decryption**
11000000	34.88788	34	-2.54%
12192768	38.66177	40.133	3.81%

From table 4.8, it can be observed that the error between predicted and observed value of time is low. Also the error in prediction of decryption time of images using which the regression equation was constructed, can be observed in table 4.7, which is low for most cases. Hence, the equation 4.5 it can predict the decryption time of image of any size.

4.6 Work Schedule

With the use of gantt chart 4.16, which provides a visual representation of the project timeline and tasks, the tasks are completed within the allotted time frame and there is no work remaining. The gantt chart provides a clear roadmap for the project.

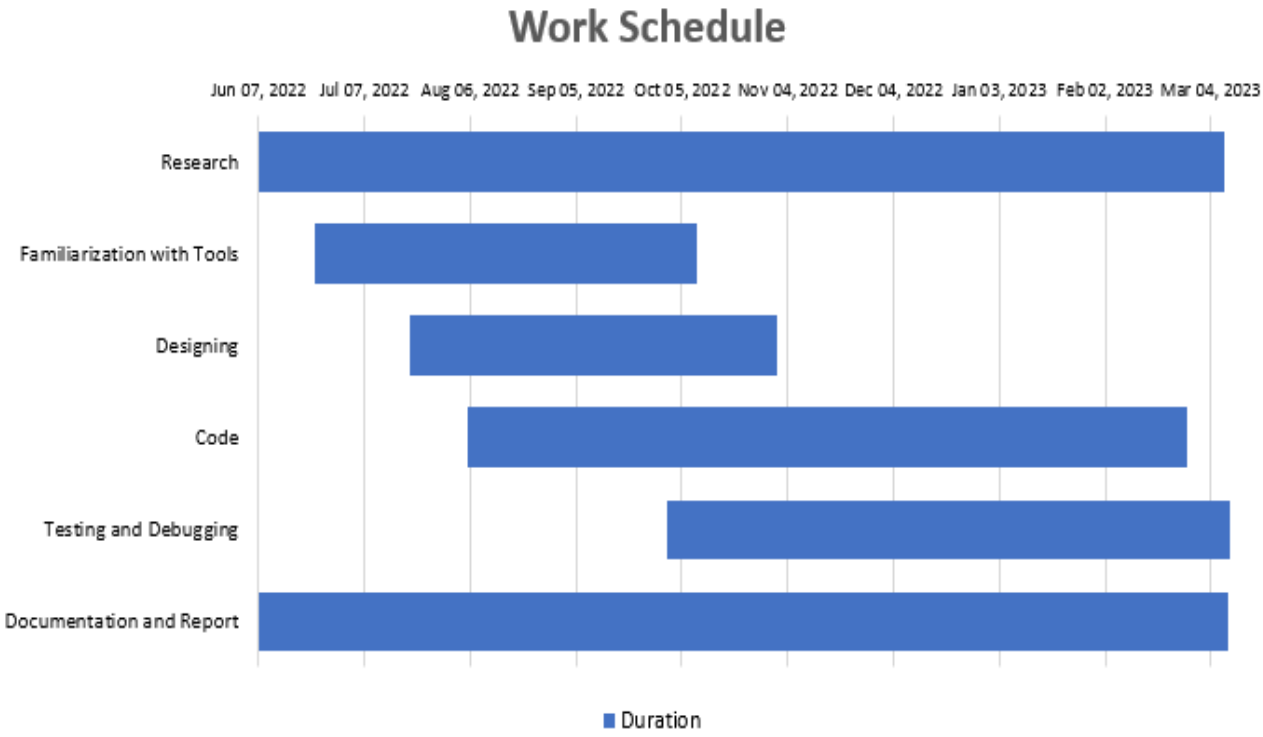


Figure 4.16: Gantt Chart

CHAPTER 5
CONCLUSION AND FUTURE ENHANCEMENT

5.1 Conclusion

The project is practical implementation and application of chaotic pseudo random number generator. DNA coding and TD-ERC System, a chaos based PRNG are used for image encryption and decryption. The image pixels are shuffled by the TD-ERCS system and pixel gray values are changed using DNA XOR operation. Upon analysis of experimental results, the algorithm employed effectively eliminates the pixels correlation, resists statistical attacks and the cipher images are highly degraded.

5.2 Drawback

The drawback of the project is that speed performance of the algorithm is not ideal. The time it takes to encrypt or decrypt an image increases with the number of pixels the image contains. The equations equation4.4 and equation4.5 shows the relation between encryption/decryption time and the number of pixels, are linear.

5.3 Future Enhancement

For enhancement, the code can be optimized, by using more efficient data structure, eliminating bottlenecks. Another approach to increase encryption speed can be using hybird encryption algorithm, combining multiple encryption algorithms, such as using chaos for initial algorithm and using another faster algorithm further in the process.

REFERENCES

[1] H. R. Shakir, "A color-image encryption scheme using a 2d chaotic system and dna coding," *Advances in Multimedia*, vol. 2019, 2019.

[2] M. Babaei, "A novel text and image encryption method based on chaos theory and dna computing," *Natural computing*, vol. 12, no. 1, pp. 101–107, 2013.

[3] J. D. Watson and F. H. Crick, "The structure of dna," in *Cold Spring Harbor symposia on quantitative biology*, vol. 18. Cold Spring Harbor Laboratory Press, 1953, pp. 123–131.

[4] S. Som, A. Kotal, A. Chatterjee, S. Dey, and S. Palit, "A colour image encryption based on dna coding and chaotic sequences," in *2013 1st International Conference on Emerging Trends and Applications in Computer Science*. IEEE, 2013, pp. 108–114.

[5] W. A. Brock, W. A. Brock, D. A. Hsieh, B. D. LeBaron, and W. E. Brock, *Nonlinear dynamics, chaos, and instability: statistical theory and economic evidence*. MIT press, 1991.

[6] S. Mandal, S. Ray, S. Roy, and S. E. Jørgensen, "Order to chaos and vice versa in an aquatic ecosystem," *ecological modelling*, vol. 197, no. 3-4, pp. 498–504, 2006.

[7] F. A. Khan, J. Ahmed, J. Ahmad, J. S. Khan, F. Ahmad, V. Stankovic, and H. Larijani, "A novel chaos-based partial image encryption scheme using lifting wavelet transform," in *The First International Nonlinear Dynamics Conference*, 2019.

[8] J. Ahmad and S. O. Hwang, "A secure image encryption scheme based on chaotic maps and affine transformation," *Multimedia Tools and Applications*, vol. 75, pp. 13 951–13 976, 2016.

[9] M. Ghebleh, A. Kanso, and H. Noura, "An image encryption scheme based on irregularly decimated chaotic maps," *Signal Processing: Image Communication*, vol. 29, no. 5, pp. 618–627, 2014.

[10] G. Ye, “A block image encryption algorithm based on wave transmission and chaotic systems,” *Nonlinear Dynamics*, vol. 75, no. 3, pp. 417–427, 2014.

[11] Y. Mao, G. Chen, and S. Lian, “A novel fast image encryption scheme based on 3d chaotic baker maps,” *International Journal of Bifurcation and chaos*, vol. 14, no. 10, pp. 3613–3624, 2004.

[12] C. Han, “An image encryption algorithm based on modified logistic chaotic map,” *Optik*, vol. 181, pp. 779–785, 2019.

[13] J. S. Khan, J. Ahmad, and M. A. Khan, “Td-ercs map-based confusion and diffusion of autocorrelated data,” *Nonlinear Dynamics*, vol. 87, no. 1, pp. 93–107, 2017.

[14] Z. Kun and F. Xing-an, “Color image encryption algorithm based on tangent-delay ellipse reflecting cavity map system,” in *2012 International Conference on Image Analysis and Signal Processing*. IEEE, 2012, pp. 1–4.

[15] J. Zhang, D. Fang, and H. Ren, “Image encryption algorithm based on dna encoding and chaotic maps,” *Mathematical Problems in Engineering*, vol. 2014, 2014.

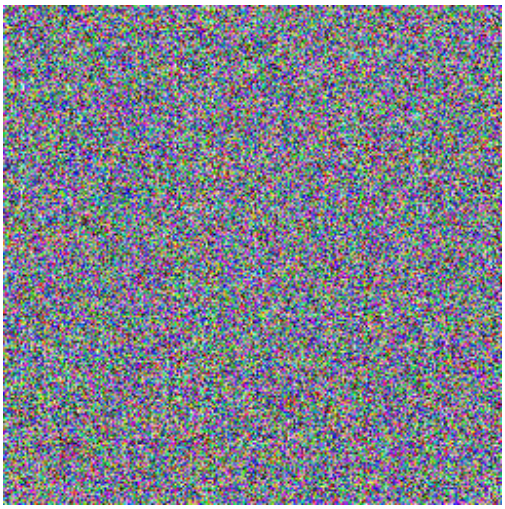
[16] A. G. Weber, “The usc-sipi image database: Version 6,” *https://sipi.usc.edu/database/*, 2018.

APPENDIX

Sample Outputs

Color Images

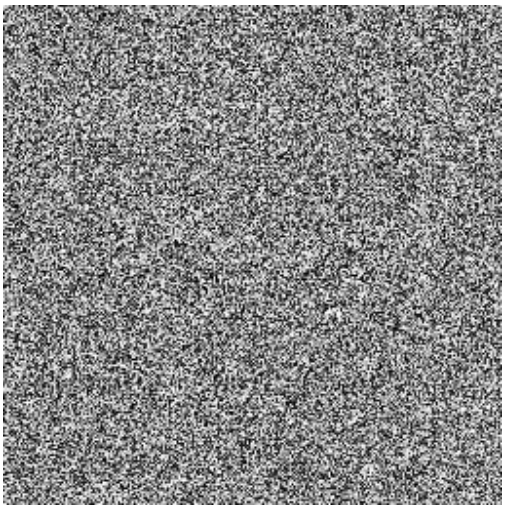
a(original)



b(encrypted)

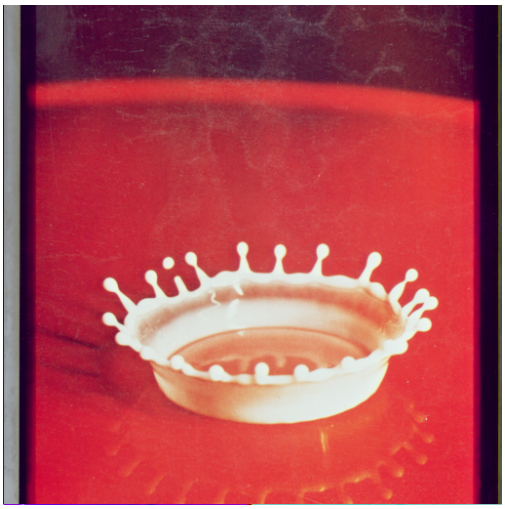
Figure 5.1: Sample 1 (256x256)

a(original)

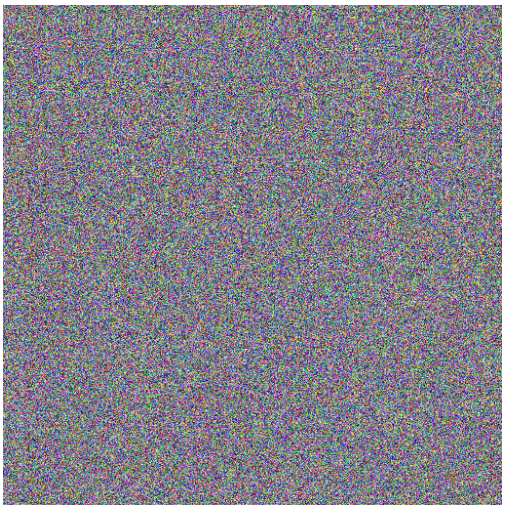


b(encrypted)

Figure 5.2: Sample 2 (256x256)



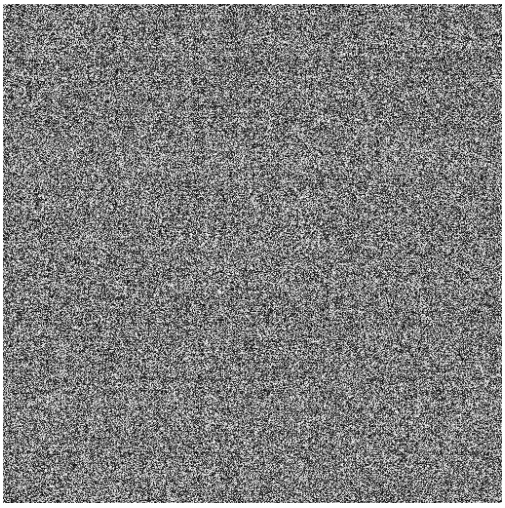
a(original)



b(encrypted)

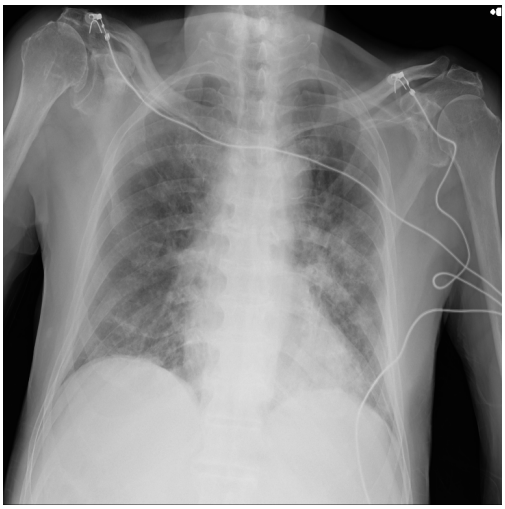
Figure 5.3: Sample 3 (512x512)

a(original)

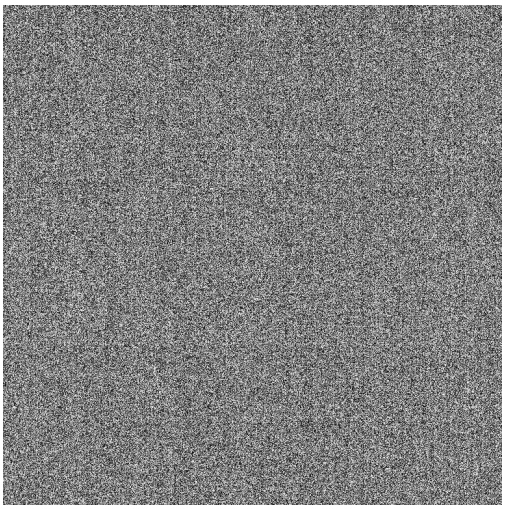


b(encrypted)

Figure 5.4: Sample 4 (512x512)

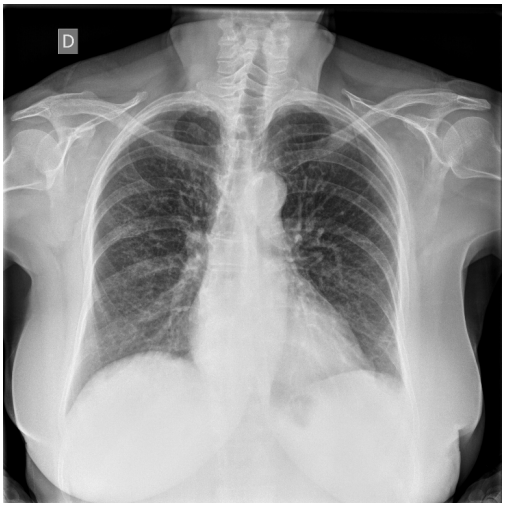


a(original)

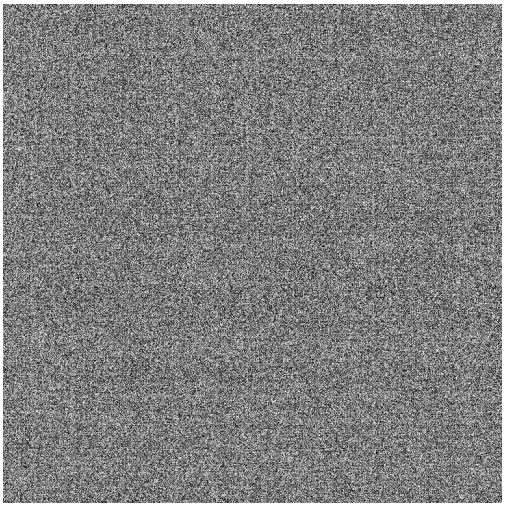


b(encrypted)

Figure 5.5: Sample 5 (768x768)



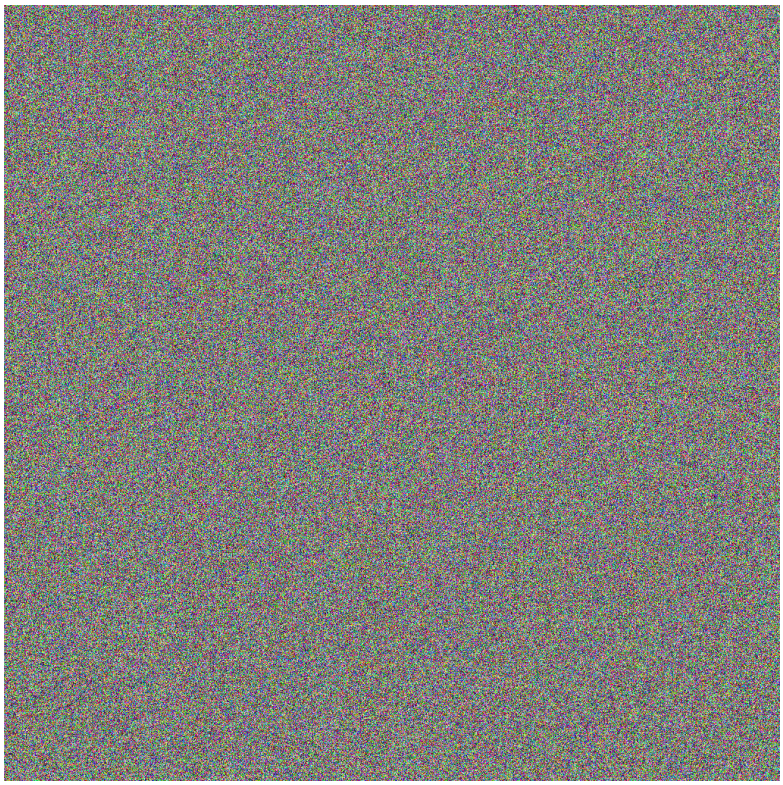
a(original)



b(encrypted)

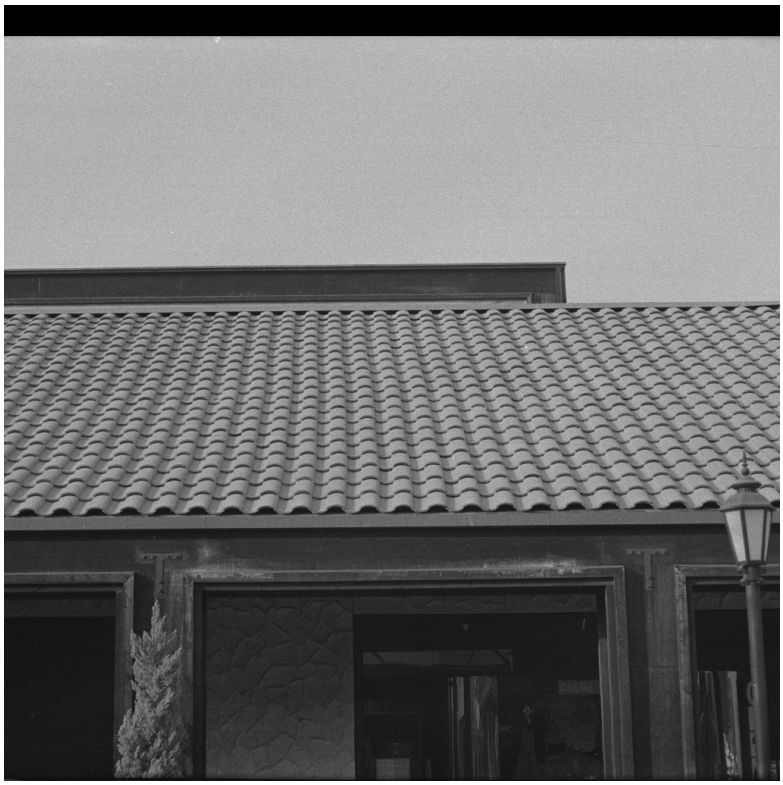
Figure 5.6: Sample 6 (768x768)

a(original)

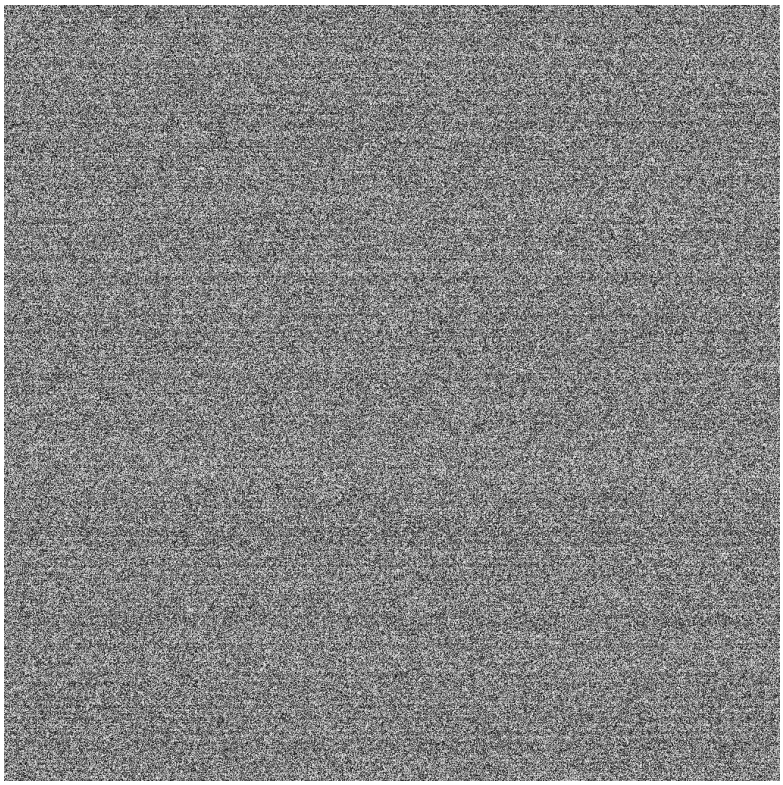


b(encrypted)

Figure 5.7: Sample 7 (1024x1024)

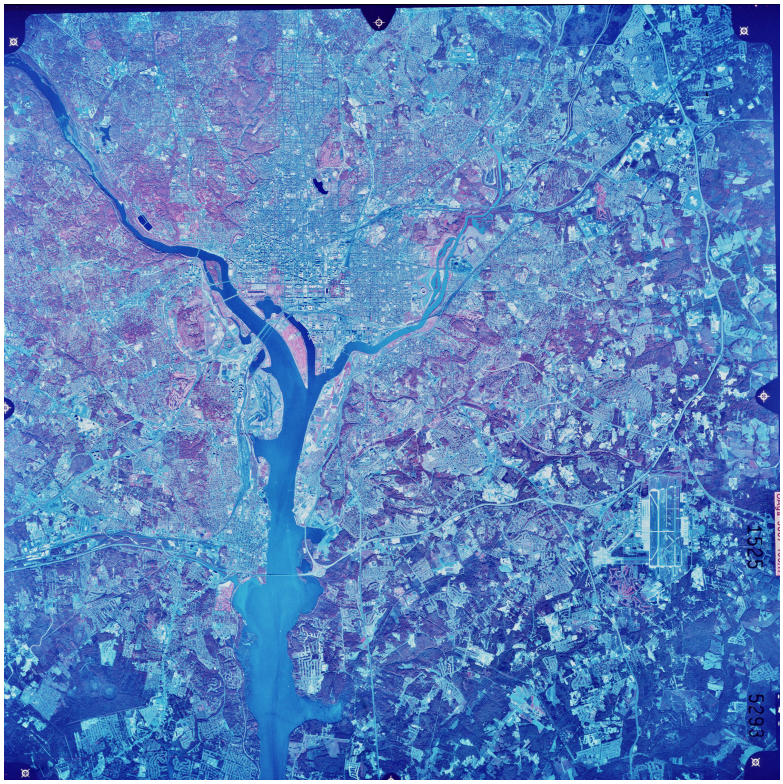


a(original)

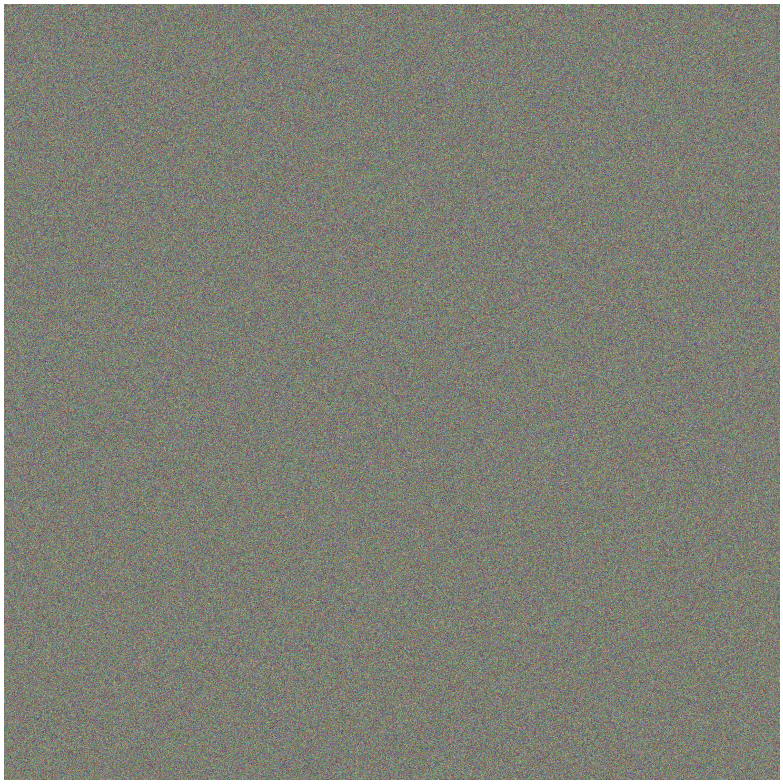


b(encrypted)

Figure 5.8: Sample 8 (1024x1024)



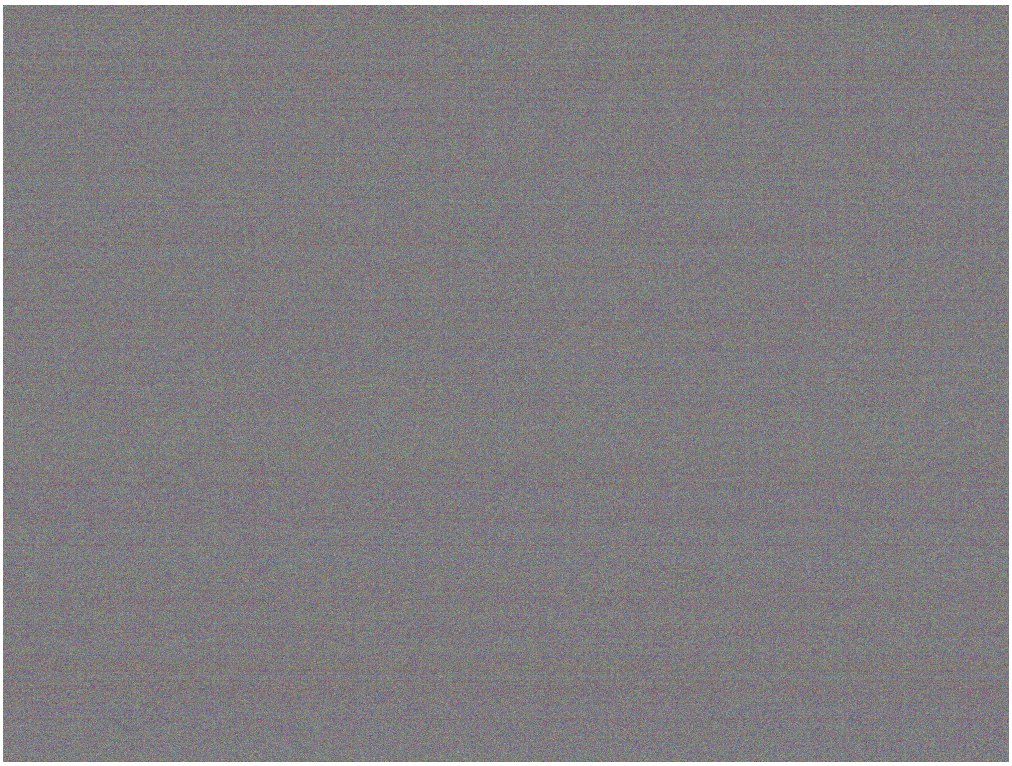
a(original)



b(encrypted)

Figure 5.9: Sample 9 (2250x2250)

a(original)



b(encrypted)

Figure 5.10: Sample 10 (4032x3024)