# Advancing Medical Image Security: A Hybrid DNA-AES Based Cryptosystem

by

Chisthia Khan
23341027
Tasfia Ayesha
20101478
Maiesha Fahomida
20101515
Samiha Jubaida Alam
20301458
Nushraq Nawer Hossain
20101242

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
School of Data and Sciences
Brac University
January 2024

# Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

**Student's Full Name & Signature:**

Chisthia Khan
23341027

Tasfia Ayesha
20101478

Maiesha Fahomida
20101515

Samiha Jubaida Alam
20301458

Nushraq Nawer Hossain
20101242

# Approval

The thesis/project titled "Advancing Medical Image Security: A Hybrid DNA-AES Based Cryptosystem" submitted by

1. Chisthia Khan (23341027)
2. Tasfia Ayesha (20101478)
3. Maiesha Fahomida (20101515)
4. Samiha Jubaida Alam (20301458)
5. Nushraq Nawer Hossain (20101242)

Of Fall 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 9, 2024.

**Examining Committee:**

Supervisor:
(Member)

Dr. Muhammad Iqbal Hossain
Associate Professor
Department of Computer Science and Engineering
School of Data and Sciences
Brac University

Co-Supervisor:
(Member)

Md. Faisal Ahmed
Lecturer
Department of Computer Science and Engineering
School of Data and Sciences
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
School of Data and Sciences
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi
Chairperson
Department of Computer Science and Engineering
School of Data and Sciences
Brac University

# Abstract

In today's technological age, it is simple to obtain someone's personal information via data transmission, especially for those who don't use sufficient data security throughout the process. Data breaches are now a significant security concern since private information is constantly transmitted through the internet around the world. Healthcare organizations experience data breaches and attacks more frequently than businesses in other sectors. Therefore, medical information needs to be transferred securely during the transmission of medical data to protect patients' sensitive details. Cryptography defines a secure transmission and communication process that results from mathematical ideas and rule-based calculation algorithms. This is how cryptography provides an efficient way for transmitting data in a secure manner that can protect it from third parties for medical data transmission. In the field of cryptography, DNA cryptography is a more secure technique because it uses the biological structure of DNA. It is a technique for hiding data that converts the alphabet into various combinations using the four bases of human deoxyribonucleic acid. Although there are many operations in DNA cryptography, the XOR operation is more versatile. DNA XOR techniques have configurations that are responsive and distinctive, in contrast to the current binary representation of 0s and 1s. The suggested encryption and decryption techniques for protecting information and storage in the sequences of DNA are particularly safe since the sequences of genes are used as the key parameters for the DNA XOR procedure. Additionally, to further strengthen security and resolve possible drawbacks of the conventional DNA-based method, the AES-256-GCM algorithm has been integrated. Comprehensive performance testing is conducted to assess the hybrid DNA-AES approach's reliability and efficiency. This research proposes a novel approach for securing medical data transmission using the DNA-AES cryptosystem.

**Keywords:** Medical data security, DNA cryptography, XOR operation, Encryption, Decryption, AES-256 GCM.

# Acknowledgement

To begin with, we are grateful to the Almighty for allowing us to finish our thesis without any significant obstacles.
Secondly, thanks to our supervisor, Dr.Muhammad Iqbal Hossain sir, for his thoughtful guidance and assistance with our work.
Third, we thank our co-supervisor, Md. Faisal Ahmed sir, and the whole judging panel of BRAC University's Information Security, Cloud Computing, and Networking.
And lastly, to our parents: it might not be possible without their help. We are nearing graduation, thanks to their sincere help and prayers.

# Table of Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Introduction

Technology is advancing at a rapid pace in the world today, and everything is now open source. You can see someone's personal information and confidential medical data, such as medical images, on websites without their consent. This occurs because it is now very easy to obtain sensitive information about someone during data transmission, especially for those who lack adequate data security while transmitting it. As confidential information is frequently exchanged all over the world through the internet, data breaches are now a serious security issue. Due to the continuous flow of information, third parties can violate the data of nearly any person or company they select from anywhere in the virtual world. So, it has become a major concern to secure the data while transmitting it. Cryptography is a technique that allows for the secure transmission of any confidential information.

The area of studying encrypted communication is referred to as cryptography. Its main function is to convert plain text messages into illegible text by encrypting them. Security attacks, including brute-force, statistical, and differential attacks, are all targets of encryption. Security breaches that aim to extract the private medical data of "elite" people or competing medical organizations may target medical images [1].To ensure the secure transmission of these data, cryptography is utilized in the encryption and decryption processes. However, conventional cryptography faced several limitations and challenges regarding this issue, which is when DNA cryptography was first introduced.

DNA cryptography was created as an outcome of recent work on DNA computing. Due to their high parallelism, incredible information density, and exceptional energy efficiency, DNA sequences are now employed for exposed cryptography, computing, and data storage.

Medical data, such as medical images, serve as essential parts of medical diagnostic techniques. For analyzing anatomical cross-sections of internal organs, they offer non-invasive methods. In the field of information security, medical image data is one of the most sensitive and significant [2]. It takes secrecy, dependability, and confirmation to transmit medical images safely. Unauthorized usage of these images may affect how successfully the patient's information is protected. To address confiden-

tiality issues surrounding telemedicine applications, medical image encryption plays a vital role. A robust encryption algorithm is necessary to prevent cryptographic attacks while transmitting medical image data. By using different encryption methods, medical images become scrambled and encrypted, which prevents the use of medical data without authorization [2].

More or less, many cryptography techniques are being used to protect medical data from third parties. But compared to conventional cryptography techniques, DNA cryptography provides superior security, privacy preservation, resistance to manipulation, effective data storage, future-proofing, and regulatory compliance, making it a compelling option for the safe transfer of medical data.

Furthermore, DNA cryptography can also be used as a hybrid model combining with other models to provide more security to end-to-end users. Advanced Encryption Standard with Galois Counter Mode (AES-GCM) is competent for securing data and it can also be merged with DNA cryptography to serve the hybrid model. As it is a block cipher mode function, it can provide data integrity and great speed for authenticated encryption. At present, hackers can hack data quite easily which means that privacy protection nowadays is inadequate. Applying a hybrid DNA cryptography with AES-GCM can make a significant change in security and privacy protection. AES-GCM can perform well in ensuring the confidentiality and integrity of the binary bit stream. In research analysis, AES-GCM was found to perform better than other work [3].

## 1.2 Problem Statement

Data breaches have become an alarming issue all around the world. It has now become a significant security concern since private information is constantly transmitted through the internet. One of the largest companies in the United States, Anthem Inc., suffered from a data breach that affected 78.8 million people and occurred during data transmission in 2015. Attackers took all the personal information of the mentioned number of individuals, which is scary [4].

Today, medical data is at higher risk than any other sector. Again in the United States, data breaches in the healthcare industry have risen 42 percent since 2020, and for the last decade in a row, any healthcare industry had the highest number of average data breach costs [20]. Moreover, these breaches not only cost money and reputation but also affect the records of existing patients. In 2020, third-party breaches affected the records of around 29.5 million patients [5].

Therefore, the need to secure medical data is now more important than ever. Cryptography ensures a secure transmission and communication procedure that is based on mathematical concepts and rule-based calculation algorithms. But conventional cryptography techniques are not efficient enough to secure this vital medical data. Rather, the number of breaches is increasing each year, which causes a huge cost. Healthcare breaches rose by 55.1% between 2019 and 2020, according to a 2021

study. In 2020 alone, there occurred around 600 data breaches [6]. The cost of a data breach went up from $3.86 million to $4.24 million in 2021, which is the highest average total cost in the last 17 years [7].

Any medical data, such as medical images, that is being transferred must be encrypted and stored, which requires the creation of a strong, secure, and effective encryption method. Compared to conventional cryptography, DNA cryptography has several advantages for the secure transmission of medical data:

**Data security:** DNA-based methods add extra security for preserving important health information. The structure of DNA sequences means they're tricky to crack, even with advanced computing. This offers great protection. For example, sharing private medical data about patient files, genetic specifics, or clinical trial outcomes are important part of maintaining privacy safeguards. Assuring confidentiality for sharing sensitive health data needs a strong security protocol.

**Privacy Protection:** DNA-based encryption methods can maintain the confidentiality of medical data. Traditional encryption approaches mostly make weaknesses for encrypting data as they require frequent swapping of encryption keys. Despite that, DNA encryption can result in using openly accessible DNA sequences by eliminating the swapping decryption keys. To increase privacy protection, private data does not need to be transmitted during the encryption process, enhancing privacy protections. In addition, DNA-derived encryption techniques enable medical data to remain private while avoiding common encryption pitfalls that make it necessary to share secret keys.

**Data storage in DNA:** DNA has the capacity to store large amounts of data. In a small space, it is feasible to store tons of information. Transmitting, sharing, and storing a large amount of health data DNA encryption can be a secure choice. The potential of DNA can make it possible to protect image data, and store, and also transmit genetic patterns.

**Data Validity:** Having inherent stability, DNA particles are not easily changed. As a result, it increases the security of medical data that will be sent by them. By tampering with the data it can easily detect medical data. Keeping all data secure is essential for medical applications. Inadequate changes can lead to misdiagnosis, poor treatment, or flawed research.

**Future-proofing process:** Data encryption with DNA helps prevent data from being altered by emerging technologies, such as quantum computing. It must be considered as medical data frequently to be kept on file for extended periods. A great option for the long run is DNA cryptography. It guarantees the privacy of medical data.

**Regulations and Medical Data:** A lot of nations like those in the European Union under GDPR and the USA under HIPAA, have strict laws about patient information privacy. DNA cryptography can help hospitals follow these laws because they can make sharing sensitive medical info more secure. This helps to cut down

on the possible data losses and keeps patient data private.

The topic "DNA Cryptography" has been the focus of many studies so far. But there is a lot more that needs to be discovered. Therefore, there is a great opportunity to work with DNA-based methods to strengthen the security of medical image data.

Therefore, the question this research is trying to answer is:

**RQ1: How can DNA cryptography be improved even further with the help of one or more security layers to ensure the security of the medical data?**

**RQ2: Does the encryption affect the original image, which may cause the decrypted image to be different?**

In DNA algorithms, addition, subtraction, and XOR operations are done to encode and decode the data. Although DNA cryptography provides unbreakable algorithms, it has some limitations; computational complexity and experimental constraints.

In this paper, we will answer the above question by applying the XOR operation to the medical data.

## 1.3 Research Objectives

This research tries to develop a security system that will help the healthcare industry secure the transmission of medical data. It will provide security to the data during transmission through its un-decodable encoding, and it will secure the medical data from third parties. The main objectives of this research include:

**i. Awareness:** Showing enough information about data breaches and need for security in medical data.

**ii. Understanding:** Providing enough information about DNA and how DNA cryptography is used to improve data security using the same biological standards of DNA.

**iii. Security:** Providing security to the medical data. The goal is to secure patients' sensitive information during transmission more effectively than the existing methods.

**iv. User acceptance and practical use:** Exploring this field and providing another secure solution so that it can gain more user acceptance and be implemented more in the real world.

## 1.4 Thesis Structure

The remaining sections of this paper are structured as follows:

- The background and literature review for DNA Cryptography are presented in Section 2.
- The proposed algorithm is described and outlined in Section 3.
- Data analysis and comprehensive result analysis are discussed in Section 4
- The concluding remarks with prospective future research are mentioned in Section 5.

# Chapter 2

# Related Work

## 2.1 Background

Deoxyribonucleic acid, or DNA, is used to keep the genetic information or materials of all the living things on earth, including humans as well as tiny viruses. Additionally, it goes by the name "information carrier" and is constructed of a lengthy and complex polymer of nucleotides, which are tiny particles. Nucleotides are composed of three elements: a nitrogenous base, a five-carbon sugar, and a phosphate group. Adenine, Thymine, Cytosine, and Guanine (A, T, C, and G) are the four bases that make up the nitrogenous base, which is used for keeping all the intricate details about organisms. Watson-Crick mentioned a complementary rule that reads as follows: "An only join with T through double binding (A=T) and C only joins with G through triple binding (CG)" [8]. And DNA cryptography combines mathematical computation with this biological property. A, C, G, and T are the four elements that can be encoded as 00, 01, 10, and 11. Using these four elements, we can obtain 4!=24 different encoding methods and maintaining the paring sequence, we have eight types of DNA encoding rules (Rule 1, Rule 2, . . . and Rule8)[2]. These eight types of encoding rules are given in Table 2.2 [9] [10]. Table 2.1[10] shows the complement of nucleotides, and Table 2.3 [10] shows the XOR operation using DNA cryptography. Additionally, the Advanced Encryption Standard (AES) is the most widely used symmetric key cryptography in today's world. To meet specific security needs, key lengths can be chosen from the standard key lengths for AES which are 128, 192, and 256 bits but the input block size is fixed at 128 bits (Figure 2.1). The typical key length for most applications is 128-bit, but 256-bit keys provide a more secure layer of security for sensitive data. AES has different modes of operation that meet specific cryptographic needs. The Galois/Counter Mode (GCM) boosts the AES (Advanced Encryption Standard) method and in order to do that GCM offers authenticated encryption with associated data (AEAD). GCM ensures confidentiality of data as well as provides integrity and authentication; these are attained through a combination of AES encryption, counter mode and Galois field multiplication [11].

| DNA | Complement |
|---|---|
| C=00 | G=11 |
| T=01 | A=10 |
| A=10 | T=01 |
| G=11 | C=00 |

Table 2.1: DNA Complement

| Rules | Rule1 | Rule2 | Rule3 | Rule4 | Rule5 | Rule6 | Rule7 | Rule8 |
|---|---|---|---|---|---|---|---|---|
| 00 | A | A | G | G | T | T | C | C |
| 01 | C | G | A | T | C | G | A | T |
| 10 | G | C | T | A | G | C | T | A |
| 11 | T | T | C | C | A | A | G | G |

Table 2.2: Rules of DNA Coding

| XOR | C=00 | T=01 | A=10 | G=11 |
|---|---|---|---|---|
| C=00 | C | T | A | G |
| T=01 | T | C | G | A |
| A=10 | A | G | C | T |
| G=11 | G | A | T | C |

Table 2.3: DNA XOR Rule

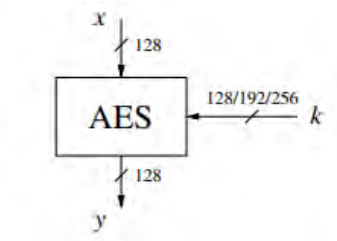



Figure 2.1: AES input/output parameters

## 2.2 Literature Review

DNA cryptography is now attracting a lot of interest around the world since it has hard-to-break algorithms. This review aims to give an overall overview of the published papers on DNA cryptography that are applied to medical data. It will also showcase their key findings and contributions.

A paper by Pushpa B. R. [12] proposed a method to implement data hiding using binary coding, random values, and a sample DNA sequence to encrypt the medical data. In this method, both the sender and the receiver share a common DNA sequence while encrypting and decrypting the data. Here, the plain text is converted into ASCII values, and then the DNA base and complementary rule are applied. After applying the complementary rule, a random key is generated. Lastly, the data has been encrypted using a DNA sequence. The method reverses during the decryption of the data. Data integrity and confidentiality are guaranteed throughout data transmission using this method of transmission.

In another paper, Abduljaleel and Khaleel [2] established a method for sending medical images that conceals them inside a speech signal file using three levels. The first level contains a proposed combination technique that utilizes a binary representation of the image data and the Zaslavsky map in order to make sure that the original medical image changes. The next level is the encryption of the distorted image by utilizing DNA coding and a series of keys created using a suggested hybrid method derived from Tinkerbelle Map and 3 Hénon that gave us a good picture distortion and made it challenging for any intruder to retrieve it. The last step involves using the LSB technique and the integer wavelet transform to conceal the contents of encrypted medical images in a speech signal and then utilizing the suggested approach depending on (the fuzzy c-means clustering algorithm and short-time energy) to pinpoint the speech areas to conceal in.

In Gollagi et al.'s [13] paper, they came up with a new picture encryption method based on a chaotic map and DNA encoding. The suggested approach is resistant to noise attacks and more secure. Chaotic maps are mathematical functions with chaotic behavior, which means they are extremely sensitive to initial conditions and can yield seemingly random and unpredictable sequences. This research intends to produce encryption algorithms that can withstand multiple attacks, such as noise attacks, by combining chaotic maps and DNA encoding.

Akkasaligar and Biradar [14] proposed a cryptosystem where a lossless discrete haar wavelet transform is utilized to compress the original medical image. The binary picture that was created from the compressed image is then broken up into four smaller images. Using a 4D Lorenz chaotic map, chaotic sequences are created that randomly rearrange the sub-image pixels. DNA coding instructions are used to create the four distinct DNA structures. The DNA XOR technique is used to combine the DNA structures, and after DNA decoding, the cipher image is obtained. The suggested cryptosystem is secure against differential, exhaustive, and statistical attacks, according to the cryptanalysis. Applications for telemedicine and e-health can use the proposed cryptosystem.

A paper by Nayak and Jayalakshmi [15] described an effective fusion cryptographic system that combines DNA and Elliptic curve cryptography with Fog computing capacity. This combination of DNA and Elliptic curve encryption produces an encrypted barrier that is difficult to breach. Initially, in order to deceive the attacker, medical images are stored in a fog of decoy images. The images are encrypted by first converting the odd and even pixel values into binary numbers, which are then translated into DNA sequences. The values are then represented by an elliptic curve with decimal points.

Pengfei et al. [16] proposed A neural network-integrated with a color image encryption technique based on hyperchaotic systems and DNA coding to provide an improved fusion random sequence as the key stream. It uses block encryption methods and DNA transformation algorithms for pixel-level image scrambling and diffusion encryption. The experimental findings show the suggested encryption algorithm's strong security capabilities and interesting application prospects.

The paper by Kumari and Nagaraju [17] represents effective encryption methods using DNA processes and chaotic maps. Utilizing the statistical approach and precise settings, create random numbers from a chaotic process. Then, using particular DNA encoding guidelines, convert the dispersed picture that results into DNA through DNA encryption. Also, the gray-code technique is carried out by shifting 8 bits to the left after the grayscale picture has been converted to a binary image. Besides, gaining access to the encrypted data, such a condensed procedure requires creating chaotic numbers, changing pixel locations, modifying pictures, and using encryption principles.

The paper by R. & Sivamangai [18] on DNA cryptography and the chaotic map indicates primary enhanced cryptography techniques that are the main subject of this review paper. Developing an understanding of both of these techniques and applying them to bring security to medical data. This study is aimed at presenting cutting-edge techniques like DNA cryptography and Chaotic Map as innovative approaches that may eventually assist in effectively encrypting digital images. The significant accomplishments highlighted in the encryption programs and the achievement of their indicators for quantitative as well as qualitative measurement are acknowledged in this evaluation. This paper significantly highlighted the achievements of encrypting programs and the achievements of their factors are also acknowledged in this evaluation.

Vaziri et al.'s [19] paper represents novel image encryption based on DNA encoding and chaos systems. As a starting point, they employ a secret key derived from the SHA-2 hash function to XOR color image pixels with DNA encoding. Secondly, they modify the chaotic system's beginning circumstances in order to produce permutation boxes that can be used to confuse the image. According to the experiment's outcomes, their encryption technique remains extremely secure and resistant to numerous attacks, including numerical, uneven, noise, and data loss attempts.

In R & Mathew's [1] paper, we found 13 papers were used in this paper's research

and comparison procedures. Traditional techniques like DES, 3DES, AES, RSA, ECC, and other encryption techniques are provided in the paper. However, the most widely used conventional encryption method is AES. The rise of chaotic systems and DNA cryptosystems are current advances in cryptography. Image encryption gains an entirely new level thanks to chaotic theory. It provides an instance of nonlinear functions and traits that are constantly shifting and uncertain. In spite of having no access to user keys, traditional encryption could be susceptible to deception. All conventional algorithms that take into account the current range of measurements were outperformed by DNA and Chaos together.

Using DNA coding, chaos, and DNA ADD-SUB operations, Amdouni et al. [20] suggested a hardware-based block cipher for medical picture encryption. The implemented algorithm showed significant complexity and randomness. The suggested method creates a secure and reliable cryptosystem by taking advantage of the properties of DNA and chaos. DNA-ADD was used to apply DNA encoding, while DNA-SUB was used to apply DNA decoding throughout the decryption process.

Srilatha and Murali [21] proposed a fast DNA cryptographic technique to provide security by using the three levels for encryption and description. Data is transformed into shift text and the l's complement in the first level. Perform the LBP procedure at the second level before converting it to a DNA sequence. The levels are applied in reverse order during decryption. The technique performed well for both encryption and decryption.

The three phases of the proposed paper by Ravichandran [22] are picture permutation, encoding, and diffusion. The permutation phase solely relies on the keys produced by various chaotic systems. The encoding phase of the suggested algorithm consists of DNA encoding and complementary operations, and the diffusion phase consists of DNA addition and XOR operations. The logistic-tent and logistic-sine maps are used to achieve coupling in the proposed medical picture encryption technique, which employs three one-dimensional chaotic maps.

In a paper, Partha, Tamal, and Abir [23] described a technique for a quantum key exchange algorithm employing Fermat numbers and DNA sequences. In order to safeguard the data transmission system, it also made use of the Watson-Crick theory of DNA sequence transformation and the randomness of Fermat numbers. First, convert the input string into the DNA sequence and use Watson-Crick's transformation to obtain the sequence. Again, convert it using a lookup table. For decryption, reverse the encryption process. A detailed explanation of the mathematics utilized in encryption and decoding is also provided. The security analysis of the algorithm is in-depth and takes into account the most common security concerns. QBER is also covered in the analysis section.

In their most recent paper, Wu and Wan [24] demonstrated a comprehensive defense system. Two efficient modules are needed to guarantee the security of cipher images: a randomly generated DNA encoding module and a content-aware permutation and diffusion module. This system improved storage capacity, parallelism, calculation speed, and security.

Overall, the publications that have been reviewed show the potential of DNA cryptography for protecting medical data, especially in image encryption. In order to achieve high security and resistance against different attacks, they emphasize the value of incorporating DNA encoding, chaotic mapping, permutation, diffusion, and other techniques. The sequential literature evaluation not only highlights some gaps and potential future study fields but also offers insights into the approaches already in use.

# Chapter 3

# Proposed Algorithm

This research presents an innovative approach to image security, drawing inspiration from the remarkable properties inherent in DNA. It all begins by converting a standard image into a binary stream, which acts as a canvas for encoding DNA sequences. These sequences undergo a transformation process using randomly generated keys to create an unbreakable cipher text. Afterward, another security layer has been added, which is done by AES-256-GCM to strengthen the overall algorithm. To establish the whole algorithm, the working flow of AES-256-GCM will be mentioned first and then the whole algorithm will be described step by step in two phases. Finally, upon decryption, these reconstruct the original image with negligible differences.

## 3.1 Encryption

### AES-256-GCM Algorithm

**Encryption:**

- **Initialization**
  - **Key Generation:** A 256-bit symmetric key (K) is generated and securely shared using The Diffie-Hellman key exchange algorithm between the sender and receiver.
  - **Nonce Generation:** A unique 96-bit nonce (N) is randomly generated for each encryption to ensure ciphertexts are distinct, even with the same plaintext.
- **Encryption:**
  - **Input Preparation:** The plaintext (in this case, DNA ciphertext) is prepared for encryption.
  - **Generate Authentication Subkey (H):** Set the H value to the result of encrypting a zero block (128 bit) with the AES algorithm using the key K. So, $H = e_k(0)$

- **Counter Generation ($CTR_i$) :** A 128-bit counter block ($CTR_i$) is constructed by combining a unique nonce (96 bits) with a counter value (32 bits) that increments for each block of plaintext. Here, counter value $CTR_0$ from the unique nonce and compute $CTR_1 = CTR_0 + 1, CTR_2 = CTR_0 + 2 \ldots$.so on where i= block number)
- **Encryption process:**
  - The 128-bit $CTR_i$ is encrypted using the AES-256 algorithm with the 256-bit key (K). This produces a 128-bit encrypted output called the keystream block [11].
  - The keystream block is XORed with a 128-bit block of the plaintext. This bit-by-bit operation creates a 128-bit ciphertext block.
  - In mathematical terms, if ek ($CTR_i$) represents the encryption of the counter blocks using the AES algorithm with the key K, and P represents a plaintext block, then the ciphertext block $y_i$ is calculated as:

$$y_i = e_k(CTR_i) \oplus P_i, \ _{i \geqslant 1}$$

  - The process is repeated for each subsequent block of plaintext, using a new counter value to generate a distinct keystream block for each encryption. (let x be the plaintext consisting of the blocks $x_1, ..., x_n$)
  - This procedure is then repeated for every plaintext block to get the final ciphertext.
- **Authentication (Tag Generation):**
  - Compute Galois Field Multiplication ($g_0$): Compute $g_0$= AAD×H using Galois field multiplication (Here, let AAD be the additional authenticated data) [11]
  - Compute $g_i = (g_{i-1} \oplus y_i) \times$ H, $1 \leqslant i \leqslant n$ (Galois field multiplication)
  - Final authentication tag: $T = (g_n H) \oplus e_k(CTR_0)$ ( n= last ciphertext block number)
- **Output:** The final ciphertext is generated with the Authentication Tag.

- **Decryption:**

  - **Initialization:**
    - Same as encryption, using shared key and shared IV (shared using Diffie-Hellman key exchange algorithm).

  - **Decryption:**
    - The receiver of the packet [$(y_1, ..., y_n)$,T,ADD] decrypts the ciphertext by also applying the Counter mode.
    - The receiver applies the shared key and IV to reverse the encryption process, transforming the scrambled ciphertext back into the original plaintext.
  - **Authentication Tag Verification:**
    - The GHASH function recalculates an authentication tag based on:
      - The decrypted data
      - The counter values used during encryption
      - The IV
      - Any additional authenticated data (AAD)
  - **Tag Comparison**
    - The receiver computes an authentication tag T' using the received ciphertext and ADD as input.
    - The receiver compares the freshly calculated authentication tag with the tag that was received along with the ciphertext.
  - **Authentication Success:**
    - If the tags match perfectly, it confirms the integrity and authenticity of the message. This means the message hasn't been tampered with or forged during transmission.
    - The receiver can accept the decrypted content as genuine.

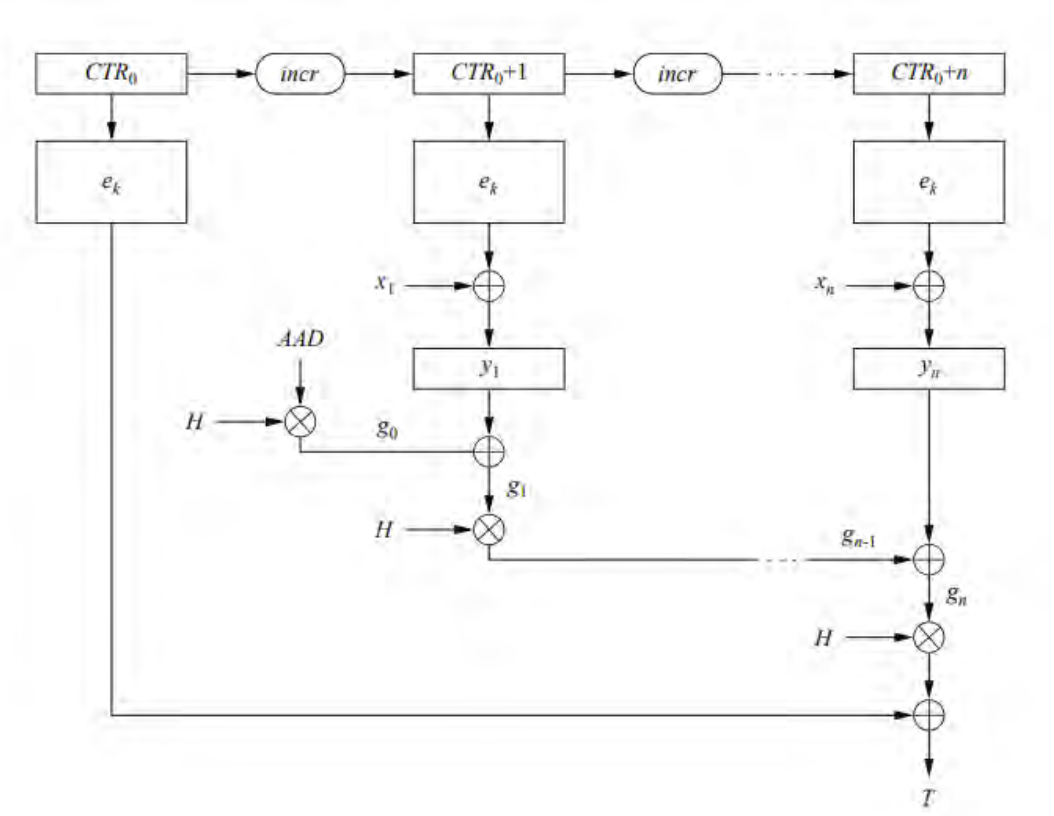
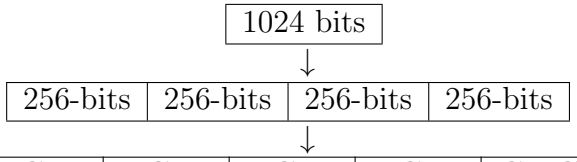



Figure 3.1: Structure of AES-256-GCM

**Phase-1: Random Key Generation**

First, a plain text DNA sequence of 1024 bits is generated randomly, which generates a sequence of A, G, C, and T of 1024 bits. The 1024-bit string is divided into four equal-sized blocks of 256 bits each. Afterward, these blocks are chosen randomly and converted into a 16x16 matrix, with each cell having 4-bit key values [25]. The key values will need to be read in a column-wise manner; four columns at a time (one column is 64 bits; therefore, four columns consist of 256 bits). The selection will be done in phase 2.




| C1 | C2 | C3 | C4 | C5 | C6-C16 | C15 |
|---|---|---|---|---|---|---|
| ATCG | GCTA | CTGA | CAGT | ATCG | ... | AGCT |
| TGAC | TGCA | AGCT | TGCA | CTAG | ... | AGTC |
| CGTA | CAGT | TGAC | CTGA | TAGC | ... | ACTG |
| TGCA | CAGT | CTCA | CGTA | CGAT | ... | TGAC |
| TAGC | CTGA | ATCG | CTGA | CTGA | ... | TACG |
| AGTC | GATC | CTGA | GTCA | GCTA | ... | TAGC |
| TCGA | GCTA | GCTA | ATCG | CGTA | ... | CGTA |
| AGTC | TGCA | CGTA | ATGC | TCGC | ... | GCTA |
| TGAC | TGCA | AGCT | TGCA | CTAG | ... | AGTC |
| CGTA | CAGT | TGAC | CTGA | TAGC | ... | ACTG |
| AGTC | GATC | CTGA | GTCA | GCTA | ... | TAGC |
| AGTC | GATC | CTGA | GTCA | GCTA | ... | TAGC |
| TCGA | GCTA | GCTA | ATCG | CGTA | ... | CGTA |
| CGTA | CAGT | TGAC | CTGA | TAGC | ... | ACTG |
| TGCA | CAGT | CTCA | CGTA | CGAT | ... | TGAC |
| TGAC | TGCA | AGCT | TGCA | CTAG | ... | AGTC |

**Phase-2: Image Encryption**

Step 1: An image (Figure 3.2) should be provided as input to the algorithm. The algorithm starts by reading the image using a buffer.


Figure 3.2: X-ray Image [26]

Step 2: With the use of the method ByteArrayoutputStream, convert the image to a binary stream. (01101000101111.....)

Step 3: To transform the binary stream data into a stream of DNA sequences, rule 4 will be applied using Table 2.2.


Figure 3.3: DNA Sequence using rule 4

Step 4: After that, the DNA complement rule (Figure 3.4) will be applied to the DNA sequence.


Figure 3.4: DNA Complement Rule

Step 5: Then that DNA sequence will be divided into 256-bit blocks.

| **256 bits** | **256 bits** | **256 bits** | **256 bits** | ..... | **256 bits** |
|---|---|---|---|---|---|

Step 6: By observing the $0^{th}$ index (A/C/T/G) of the first 256-bit block, four columns will be selected from the 16x16 matrix (phase-1) and these four columns will generate the mentioned 256-bit key. This first 256-bit block will be as it is so that, on the receiver's end, the image can be decrypted using the same key.

Step 7: The rest of the 256-bit blocks will be XOR-ed with the 256-bit key, and the randomly generated plain text of 1024 bits (Phase-1) will be concatenated with this, and thus the DNA ciphertext will be generated.

Step 8: Then, AES-GCM encryption will be performed with a key length of 256-bit into the DNA ciphertext to generate the final ciphertext.

CCTATCTTGTCCGGGGTAGTCGTTGGGCTTATG.....CCACCCGGG

↓

cCuS2/fpeSvn9qi/JGMelVif00frBIp2b0nac01NB2VVWvXfPF+.....bv/EZNE7ca2ox

Step 9: Then this concatenated ciphertext will be sent to the receiver's end.

The workflow of the whole algorithm is shown using the flowchart below (Figure 3.5).

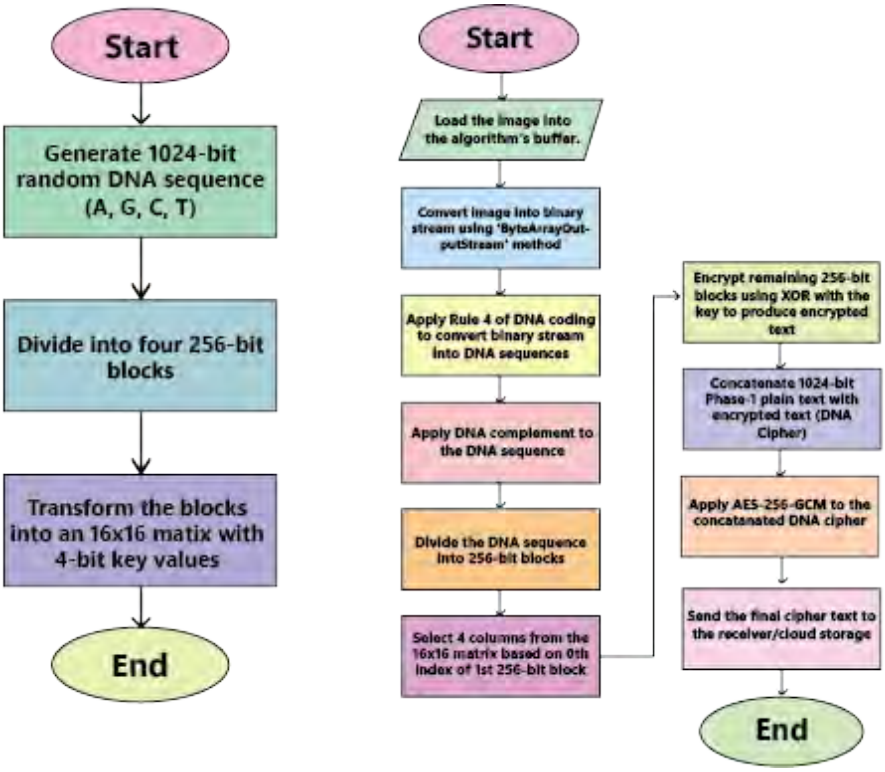



Figure 3.5: Phase-1 & 2 Flowchart

## 3.2 Decryption

Step 1: The concatenated ciphertext will be received on the receiver's side.

Step 2: Then the cipher text will be decrypted using the AES-256-GCM algorithm and this will generate the DNA ciphertext with a concatenated 1024-bit key sequence at the beginning along with XOR-ed text with the key and the first 256-bit block without being XOR-ed.

Step 3: Then the 1024-bit key sequence (for generating the 16x16 key matrix), which was concatenated with the DNA ciphertext, will be separated.

Step 4: Then the key matrix (16x16) will be generated in the same way as Phase-2.

Step 5 By observing the $0^{th}$ index of the first 256-bit block (the bit right after the concatenated 1024-bit sequence) the key will be generated from the matrix.

Step 6: Using this key, an XOR operation will be performed with all of the 256-bit blocks except the first 256-bit block, by doing this it will be converted into DNA complement sequence.

Step 7: After that, the DNA complement rule will be applied to the DNA sequence from Step 6 and it will generate a regular DNA sequence.

Step 8: Then the DNA sequence will be converted to a binary sequence.

Step 9: Finally, from this binary sequence, the desired image (Figure 3.6) will be generated.


Figure 3.6: Decrypted Image

Here, in the flow chart it shows that an image has been generated as an input. The DNA algorithm gets the image input and reads its buffer, and converts the image into binary sequence. After that, the binary sequences convert into DNA sequences. In that sequence, a DNA complement will be applied to the XOR operation. In the meantime, a key will be generated in the XOR operation. After that, It will be a DNA ciphertext. And on that text, AES-256 GCM will be implemented to get encrypted ciphertext.

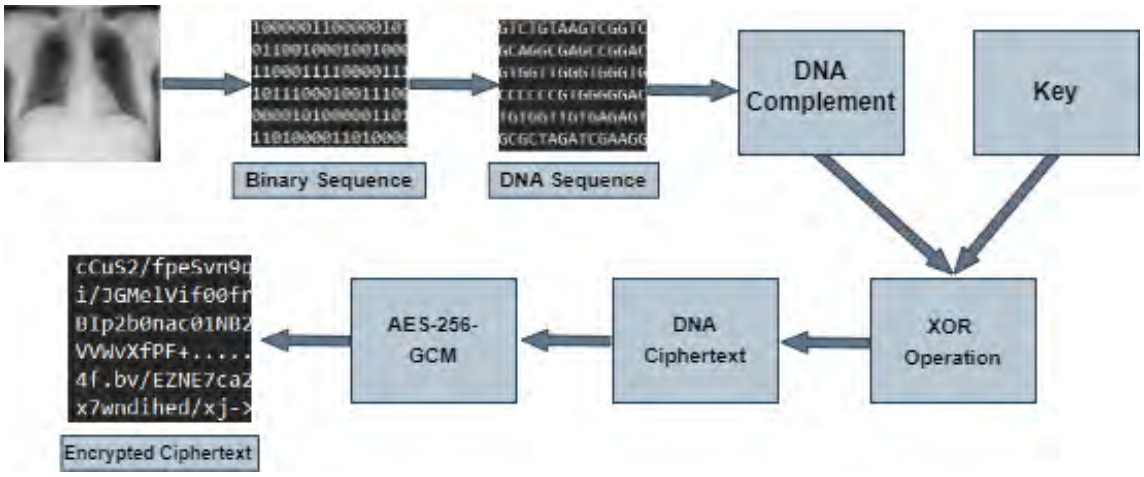



Figure 3.7: The encryption scheme

# Chapter 4

# Comprehensive Analysis

## 4.1 Data Analysis

### 4.1.1 Detailed Algorithm Analysis

**1. Image to Binary Conversion:**
At first, an image (Figure 4.1) is taken and read using a buffer and then converted to binary using ByteArrayOutputStream.

*BufferedImage image = ImageIO.read(new File(imagePath))*
*binaryString = convertImageToBinaryString(image)*


Figure 4.1: X-ray Image [26]

↓

1111111111011000111111111110000000000000000010000. . . . ...11001 (498190 bits)

**2. Binary to DNA consequence::**
Then, this binary to DNA sequence (00=G, 11=C, 01=T, and 10=A) is obtained, and the DNA complement rule is applied to this sequence.

*dnaCipherText= getDNACipherText(binaryString)*
*dnaComplimentRule4 = DNAComplimentRule4(dnaCipherText)*

Result (DNA to DNA complement conversion ) -
CCCCCTAGCCCCCAGGGGGGGTGGTGAATGTATGATTGTAGG.....CCTAT
(DNA seq.)
↓
GGGGGATCGGGGGTCCCCCCCCACCACTTACATACTAACATCC.....GGATA
(complement DNA seq.)

**3. DNA Encryption:**

*paddedDNASequence = convertTo256BitBinaryBlock(dnaComplimentRule4)*
*XORConvertedString = DNASequencetoXORConverstion(paddedDNASequence)*

A randomly generated key was used to apply DNA encryption ( step 5-7 of phase-2 from chapter 3) to the DNA sequence from the previous step. The cipher text after the encryption becomes-

CCCCCTAGCCCCCAGGGGGGGTGGTGAATGTATGATTGTAGG.....CCCTAT
(ciphertext)

**4. AES-256-GCM Encryption:**

Then AES-GCM encryption was performed on the previous step's DNA cipher text with a key length of 256-bit. The outcome of the encryption becomes-

*aesGCM = AESGCM(XORConvertedString)*

CCTATCTTGTCCGGGGTAGTCGTTGGGCTTATG.....CCACCCGGG
↓
cCuS2/fpeSvn9qi/JGMelVif00frBIp2b0nac01NB2VVWvXfPF+.....bv/EZNE7ca2ox

**5. Decryption and Image Construction:**

On the receiver side, the receiver gets this cipher text, and following all the above steps in reverse order, the following image is generated in the end.

**6. Analyzing the Image:**

Additionally, some analysis is to be performed on the output image to see if the image generates an actual image on the receiver side or not.

### 4.1.2 Hashing Techniques

Using the Average Hash (A-Hash) method, the images can be represented in binary format. It can calculate the Hamming distance between those codes. The lower the hamming distance result, the higher the similarity between those images. A similarity score is determined by the Hamming distance; if the similarity score is near 0, the pictures are regarded as being different. If the similarity score is near 1, it defines both images are similar. With the use of a bar chart, the sample presents these similarity scores in a clear and understandable manner. After analysis, the similarity result is 0.984375 (Figure 4.2), which is seen to be a good result and it also denotes that both of the images share a significant similarity and an adequate amount of visual information.

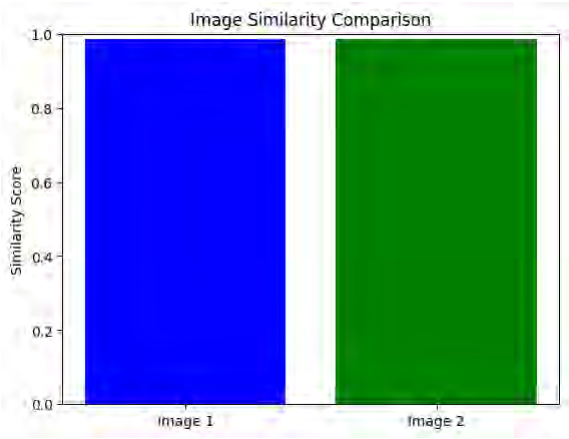



Figure 4.2: Hashing Techniques

### 4.1.3 Pixel-wise Comparison

In image processing and computer vision, a method called pixel-wise comparison, also known as pixel-level comparison, serves the purpose of evaluating the similarity or dissimilarity between two images by comparing the individual values of each pixel.

In this process, two images are loaded: the original image and the decrypted image. The dimensions of these images are then checked for compatibility to ensure they have the same size. Afterwards, a pixel-wise comparison is conducted to identify any differences between the corresponding pixels in the two images. This analysis generates a distinct difference that highlights these variations. By representing it as a heatmap, with cooler colors indicating smaller differences and warmer colors indicating larger differences, visualizing these disparities becomes more accessible. Overall, this visualization provides an efficient and intuitive way to assess and compare the original and decrypted images.

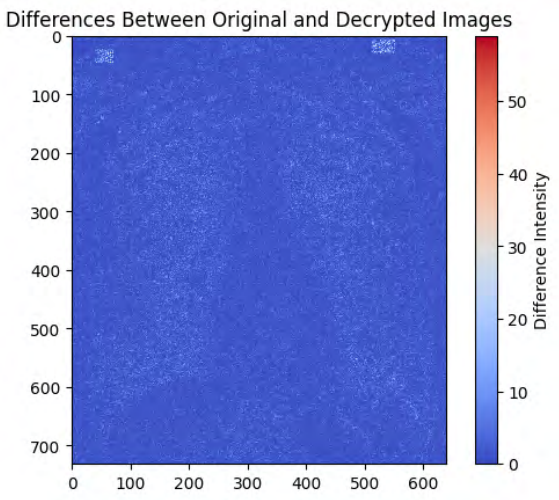



Figure 4.3: Pixel-wise Comparison

The model loads an alternative image to illustrate the differences between the two images. It accomplishes this by creating a heatmap, where cooler tones (blue) indicate minor differences and warmer tones (red) represent major differences. A color bar on the side assists in interpreting the intensity of these disparities. This visual aid facilitates the identification and examination of variations between the original and decrypted images.

In our comparison, the differences between the two images are minor, as the whole heatmap has a cooler tone (Figure 4.3). There are a few red dots (pixels), which are negligible. So, we can say that we have a decrypted image that is almost identical to the original image.

## 4.2 Result and Analysis

### 4.2.1 Keyspace Analysis

By systematically looking through every potential key, a brute-force attack can be used to crack a cipher. The strength of a cipher's key space (k) and the attacker's computing capacity determine whether a brute-force attack is feasible. It is widely accepted that a key space of size $k < 2^{100}$ is insufficiently safe given the speed of modern computers [27].

In the proposed algorithm, 256-bit blocks are eventually converted into 16x16 matrices, where each cell has 4-bit key values. Then the key values are read in a column-wise manner (four columns at a time).

Keyspace for one 256-bit block = $2^{4*16} = 2^{64}$

For four blocks, $2^{64} \times 2^{64} \times 2^{64} \times 2^{64} = 2^{256}$

After implementing GCM, total keyspace = $2^{256} \times 2^{256} = 2^{512}$, which is much higher than $2^{100}$. As such, brute-force attacks against the key are computationally challenging.

### 4.2.2 CIA Analysis

The "CIA triad" is an essential cybersecurity model. It is used to guide security policies and guard data. The letters "CIA" stand for confidentiality, integrity, and availability. The CIA Triad is a common blueprint for building security systems.

- **Confidentiality:** The goal of confidentiality is that only authorized people can see the information. This is achieved with protective actions by organizations. The AES algorithm's GCM mode has been used as the encryption algorithm. Known as the Galois Counter Mode (GCM), this mode does more than just encrypt the data; it also creates a unique code called a Message Authentication Code (MAC). The MAC is a type of security check. It is made by the sender and sent with the message. The recipient makes their own MAC from the message. If their MAC matches the one sent by the sender, the message is verified as authentic and hasn't been tampered with during the transfer. Message integrity and authentication are the terms for these two features, respectively. GCM uses counter-mode encryption to protect the plaintext's confidentiality [11]. It implies that GCM offers a way to verify the integrity and authenticity of data and encrypt it to keep it private.

- **Integrity:** Integrity is an assurance that data is accurate, comprehensive, and unaltered. In other words, it assures that the data is still reliable and trustworthy and hasn't been altered in any manner. Data integrity is crucial in medical image transmission, reaching beyond simple security and into the domain of life-and-death considerations. Image modification may cause a modest tumor to be misinterpreted, which might result in improper treatment plans, delayed diagnosis, and perhaps permanent effects for patients. It is guaranteed by data integrity that images go to medical staff unchanged, keeping important information for accurate evaluation and prompt response. In the proposed approach, GCM mode has been implemented in the DNA algorithm. GCM can ensure medical image data integrity and confidentiality, which provides a reliable and effective solution. Additionally, integrity can identify and detect data modifications. Therefore, integrity is a crucial factor in maintaining patients' information to safeguard their privacy.

- **Availability:** It is ensured by system, application, and data availability that users may access them when required.The most common kind of denial-of-service attacks are those that cause access to information, devices, systems, or other network resources and affect availability. Without availability, users may not get the resources they need, and this leads to a decrease in productivity or even financial losses. Hence, organizations cannot help but put up necessary measures for protection against availability threats, such as redundant systems, backup and recovery procedures, security controls, and incident response plans. The proposed system ensures that it will always be available.

### 4.2.3 NPCR

To measure how powerful a DNA encryption scheme is against differentiating attacks, NPCR (the Number of Pixel Change Rate) is used in DNA cryptography. A higher NPCR score signifies that even minor modifications to the plaintext provide noticeably differing ciphertexts. An NPCR number that is almost 100% is ideal. For the proposed algorithm, the obtained NPCR values for different image formats were found to be close to 100%.

$$NPCR = \frac{\sum_{i,j} D(i.j)}{M \times N} \times 100\%$$

### 4.2.4 UACI

Unified Average Changing Intensity is referred to as UACI. It's another crucial parameter that determines how well an encryption method defends against differentiating attacks. A higher UACI value signifies that even slight changes to the plaintext cause a noticeable and consistent change in the ciphertext's intensity levels. The ideal value for UACI is around 33.33%, which implies entirely random variations in

intensity throughout the ciphertext. The obtained values are approximately 33.3% for different images, which are shown in the table.

$$UACI = \frac{1}{M \times N} \left[ \sum_{i,j} \left[ \frac{I(i,j) - K(i,j)}{255} \right] \right] \times 100\%$$

### 4.2.5 PSNR

In image processing and compression, peak signal-to-noise ratio, or PSNR, is employed. It serves as a comparison tool between the original and objectively assessed reconstructed image quality. To put it briefly, it shows how similar and consistent the two images are visually. Reduced MSEs correlate with higher PSNR values, suggesting that the reconstructed picture's pixel values and visual quality are closer to the original. A dB level above 40 is typically considered acceptable, and a dB level beyond 50 is extraordinary and looks almost exactly like the original. Excellent image quality was achieved with a PSNR of 42.75 dB. This suggests that when comparing the reconstructed image to the original, there shouldn't be much if any, noticeable distortion. This PSNR value would be suitable for tasks like casual watching, internet sharing, basic photo editing, and even certain professional applications where perfect fidelity isn't required.

$$PSNR = 10 \times lg \left( \frac{255^2}{MSE} \right)$$
$$MSE = \frac{1}{M \times N} \sum_{i=1}^{N} \sum_{j=1}^{M} \left[ I(i,j) - I'(i,j) \right]^2$$

### 4.2.6 Entropy

In computing and security purposes, entropy is referred to as randomness. It is considered as the amount of randomness an image can have. It also refers to the number of plaintext bits to be the ambiguity for ciphertext. For DNA cryptography, it measures the randomness of its security to make its sequence highly uncertain.

$$H(m) = \sum_{i=0}^{2^N - 1} P(m_i) log_2 \frac{1}{P(m_i)}$$

Here, p(mi) represents the strength of the probability of a particular function. Shannon's entropy formula also contains the average number of pieces of information for the randomness of ciphertext in a DNA sequence [28] [29].

### 4.2.7 Calculated Values for Different Size Images:

NPCR, UACI, Entropy and PSNR tests have been conducted upon images of different formats and sizes. It is observed in Table 4.1 that the values are very close to each other and all of the values are near ideal value or above.

| Image Size | Image Format | NPCR | UACI | PSNR | Entropy |
|---|---|---|---|---|---|
| 25 KB | JPG | 33.41% | 99.65% | 42.75 dB | 7.8854 |
| 73 KB | PNG | 33.38% | 99.61% | 39.99 dB | 7.9723 |
| 128 KB | JPG | 33.28% | 99.58% | 41.23 dB | 7.9519 |
| 210 KB | PNG | 33.50% | 99.62% | 42.12 dB | 7.9846 |
| 27 KB | JPG | 33.15% | 99.67% | 44.45 dB | 7.9997 |
| 2.2 MB | JPG | 33.15 % | 99.61% | 44.41 dB | 7.9908 |
| 7 MB | JPG | 33.43% | 99.61% | 36.16 dB | 7.9759 |

Table 4.1: Size-wise obtained values

### 4.2.8 Execution time

Execution time denotes the duration required for the conversion of a plaintext message to ciphertext and its reverse. The performance of the system exhibits an inverse relationship with the execution time in Table 4.2.

| Image | Encryption Time (ms) | Decryption Time (ms) |
|---|---|---|
| 73 KB | 351 | 206 |
| 25 KB | 285 | 179 |
| 128 KB | 425 | 233 |
| 210 KB | 432 | 287 |
| 27 KB | 226 | 167 |
| 2.2 MB | 5111 | 1536 |
| 7 MB | 1350 | 816 |

Table 4.2: Execution time

### 4.2.9 NIST

The National Institute of Standards and Technology is referred to as NIST. It is among the most crucial testing sets for determining how random an encrypted message is. This includes 15 statistical tests that search for various forms of sequence non-randomness. The tests use different bit statistics or bit block statistics to examine the unpredictability of the input. Every test in the NIST STS (Statistical Test Suite) examines the bit stream's overall randomness. Multiple tests, which divide the bit stream into several relatively large components and compute a bit characteristic for each, can also be used to detect local non-randomness. After that, all of these partial characteristics are used to construct the test statistic. The resulting ciphertext was subjected to NIST STS at a significance level of $\alpha = 0.01$. The Table 4.3 displays an example of the test result. The test results show that the suggested approach passes most of the NIST test suites. At the significance level of 0.01, the data can be considered random if less than 7 NIST STS tests are failed. This outcome shows that the produced cipher texts are randomly distributed, unexpected,

independent, and uniform, all of which are desirable statistical qualities [30]. Thus, the suggested system has a high level of security.

| **Statistical** | **P-value** |
|---|---|
| Monobit Test | 0.01430587843 |
| Frequency Block Test | 0.18734629196 |
| Runs Test | 0.20683697625 |
| Longest Run of ones | 0.85682480973 |
| Binary Matrix Rank Test | 0.02422579765 |
| Discrete Fourier Transform Test | 0.00026257887 |
| Non-Overlapping Test | 0.00000000046 |
| Overlapping Test | 0.03124567967 |
| Maurer's Universal Statistical | 0.29646798566 |
| Linear Complexity Test | 0.98654345879 |
| Serial Test | 0.00001944874 |
| Approximate Entropy Test | 0.98257867789 |
| Cumulative Sums Test (Forward) | 1.07642480987 |
| Cumulative Sums Test (Backward) | 0.00087654234 |
| Random Excursions Test | 0.00000012824 |
| Status | Pass |

Table 4.3: NIST Values

### 4.2.10 Time Complexity

Time complexity defines the number of resources an algorithm can take to execute. It also refers to the time it takes as a function of the input size. It is a measurement of how effective an algorithm can be. The time complexity determines the function of an algorithm's performance. The total time needed to transform a plain text into ciphertext is, in general, known as encryption time, and the total time needed to transform a ciphertext into plaintext is known as decryption time. The overall time complexity of an encryption algorithm is the result of adding the time needed for each phase. The time complexity of a constant time for generating a random DNA sequence is O(1). The time complexity for linear time for converting the DNA sequence to a matrix, image to binary conversion, key generation, XOR operation, and AES encryption is O(n).

$$\begin{aligned} T_{(encryption)} &= T_{(phase1)} + T_{(phase2)} + T_{(phase3)} + T_{(phase4)} + T_{(phase5)} \\ &= O(1) + O(n) + O(n) + O(n) + O(n) \\ &= O(n) \end{aligned}$$

The time complexity of decryption time calculation is also similar to the encryption time of DNA sequence. The only difference is that we do not need to randomly generate DNA sequences for matrix generation. The time complexity for decryption part becomes-

$$
\begin{aligned}
T_{(decryption)} &= T_{(phase1)} + T_{(phase2)} + T_{(phase3)} + T_{(phase4)} \\
&= O(n) + O(n) + O(n) + O(n) \\
&= O(n)
\end{aligned}
$$

### 4.2.11 AES-GCM Over AES-CBC:

At first, AES CBC (Cipher Block Chain) was implemented with DNA Cryptography, and all the necessary testing was conducted. The result is given in the Tables 4.4, 4.5, 4.6.

| Image | **AES-CBC** | | **AES-GCM** | |
|---|---|---|---|---|
| | NCPR (%) | UACI(%) | NPCR(%) | UACI(%) |
| 73 KB | 99.58 | 33.29 | 99.61 | 33.58 |
| 128 KB | 99.48 | 33.32 | 99.58 | 33.59 |
| 210 KB | 99.51 | 33.38 | 99.67 | 33.61 |
| 2.2 MB | 99.45 | 33.40 | 99.65 | 33.52 |

Table 4.4: NPCR-UACI Value Comparison

| **Statistical** | **P-value GCM** | **P- Value CBC** |
|---|---|---|
| Monobit Test | 0.01430587843 | 0.01349756879 |
| Frequency Block Test | 0.18734629196 | 0,18084387997 |
| Runs Test | 0.20683697625 | 0.18457698784 |
| Longest Run of ones | 0.85682480973 | 0.85435465548 |
| Binary Matrix Rank Test | 0.02422579765 | 0.01998765365 |
| Discrete Fourier Transform Test | 0.00026257887 | 0.00026879853 |
| Non-Overlapping Test | 0.00000000046 | 0.00000000046 |
| Overlapping Test | 0.03124567967 | 0.01247952562 |
| Maurer's Universal Statistical | 0.29646798566 | 0.21246478959 |
| Linear Complexity Test | 0.98654345879 | 0.98346557965 |
| Serial Test | 0.00001944874 | 0.00002834585 |
| Approximate Entropy Test | 0.98257867789 | 0.00003432544 |
| Cumulative Sums Test (Forward) | 1.07642480987 | 1.07886745023 |
| Cumulative Sums Test (Backward) | 0.00087654234 | 0.00076574589 |
| Random Excursions Test | 0.00000012824 | 0.00000012836 |
| passed | 10 | 9 |

Table 4.5: NIST Value Comparison

| Image | Encryption Time of GCM (ms) | Encryption Time CBC (ms) |
|---|---|---|
| 73 KB | 351 | 392 |
| 25 KB | 285 | 305 |
| 128 KB | 425 | 498 |
| 210 KB | 432 | 344 |
| 27 KB | 226 | 257 |
| 2.2 MB | 5111 | 5726 |
| 7 MB | 1350 | 1513 |

Table 4.6: Time Comparison

Then AES-256-GCM and DNA Cryptography were combined together which produced better results in terms of security and speed. The detailed comparison between our proposed model and existing models is given.

### 4.2.12 Comparison

NPCR, UACI, Entropy, and Keyspace were compared using the proposed algorithm's calculated values. Comparing the proposed DNA-AES hybrid algorithm to 9 other papers resulted in satisfactory results. All values are computed upon a $256 \times 256$ test image. The compared values are shown in the Table 4.7 below:

| Method | NPCR (%) | UACI(%) | Entropy | Keyspace |
|---|---|---|---|---|
| [31] | 99.61 | 33.42 | 7.9971 | $> 2^{128}$ |
| [32] | 99.61 | 33.48 | 7.9969 | $> 2^{300}$ |
| [33] | 99.63 | 33.41 | 7.9974 | $> 2^{716}$ |
| [34] | 99.64 | 33.63 | 7.9975 | $> 2^{200}$ |
| [35] | 99.29 | 33.58 | 7.9969 | – |
| [36] | 99.64 | 33.38 | 7.9976 | $> 2^{199}$ |
| [37] | 99.60 | 33.43 | 7.9973 | $> 2^{199}$ |
| [38] | 99.65 | 33.54 | 7.9975 | $> 2^{432}$ |
| [39] | 99.65 | 33.58 | 7.9978 | $> 2^{624}$ |
| Proposed | 99.67 | 33.61 | 7.9979 | $> 2^{512}$ |

Table 4.7: Comparison Table

It is observed that the proposed algorithm has the highest NPCR, UACI, and Entropy values. Upon this comparison, the proposed algorithm is better in terms of these parameters.

# Chapter 5

# Conclusion

Securing medical data has become one of the major concerns of our time. It costs not only a huge amount of money but also endangers the patient's security as well as the organization's reputation. Recent studies have indicated that DNA cryptography works better than conventional cryptography in terms of securing the transmission of medical data. Therefore, research is being continued in the field of DNA cryptography. The algorithm progressed by generating a key and executing an XOR operation with the DNA complement blocks of the images. Additionally, the AES-256 GCM algorithm was integrated to enhance the security of the conventional DNA algorithm, forming a hybrid DNA-AES cryptosystem. Comprehensive testing was subsequently conducted to evaluate the performance of the proposed algorithm. For future work, the code complexity needs to be improved to speed up the encryption and decryption processes. Besides, a 200kb image becomes 640kb of cipher text, which requires more space and needs to be studied further. Moreover, encryption and decryption time are not only influenced by image size; some other factors, like dimension, also have a significant impact on the overall process time of our algorithm. The keyspace can be improved further in the future. Additionally, this system needs to be implemented in a real-life environment to observe how it behaves, especially for end-to-end users. All these things will be studied and experimented thoroughly in the future.

# Bibliography


[1] R. R. Kumar and J. Mathew, "Image encryption: Traditional methods vs alternative methods," in *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, IEEE, 2020, pp. 1–7.

[2] I. Q. Abduljaleel and A. H. Khaleel, "Hide medical images in a speech signal using dna coding and fuzzy c-means," in *2020 2nd Annual International Conference on Information and Sciences (AiCIS)*, IEEE, 2020, pp. 119–126.

[3] N. Ahmad. "Advanced encryption standard with galois counter mode using field programmable gate array." (2018), [Online]. Available: https://www.semanticscholar.org/paper/Advanced-Encryption-Standard-with-Galois-Counter-Ahmad-Wei/8da4572483be0a9646e9d2ce3357bf7723889b6e.

[4] J. Roman and R. Ross. "Anthem breach: 78.8 million affected." Retrieved May 19, 2023. (n.d.), [Online]. Available: https://www.bankinfosecurity.com/anthem-update-a-7946.

[5] "Protenus: Healthcare compliance analytics." Retrieved May 19, 2023. (2023), [Online]. Available: https://www.protenus.com/.

[6] T. Stone. "The importance of healthcare data security: Solutions & tips," Prime TSR. (2022), [Online]. Available: https://primetsr.com/insights/the-importance-of-healthcare-data-security/ (visited on 05/18/2023).

[7] "2023 cyber security statistics: The ultimate list of stats, data & trends." Retrieved May 20, 2023. (n.d.), [Online]. Available: https://purplesec.us/resources/cyber-security-statistics/.

[8] J. D. Watson and F. H. Crick, "Molecular structure of deoxypentose nucleic acids," *MOLECULAR STRUCTURE OF NUCLEIC ACIDS*, Apr. 1953. [Online]. Available: https://dosequis.colorado.edu/Courses/MethodsLogic/papers/WatsonCrick1953.pdf.

[9] D. H. ElKamchouchi, H. G. Mohamed, and K. H. Moussa, "A bijective image encryption system based on hybrid chaotic map diffusion and dna confusion," *NCBI*, Feb. 2020, Retrieved May 18, 2023. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7516607/.

[10] Q. Liu and L. Liu, "Color image encryption algorithm based on dna coding and double chaos system," *IEEE Access*, vol. 8, pp. 83 596–83 610, 2020.

[11] C. Paar and J. Pelzl, *Understanding Cryptography: A Textbook for Students and Practitioners*. Springer, 2010.

[12] P. B. R, "A new technique for data encryption using dna sequence," 2017.

[13] D. S. G. Gollagi, D. S. R, D. S. K. G, and D. P. K. Pareek, "A novel image encryption optimization technique," 2021.

[14] P. T. Akkasaligar and S. Biradar, "Medical image compression and encryption using chaos based dna cryptography," 2021.

[15] L. Nayak and V. Jayalakshmi, "Protecting medical images by using fused cryptographic technique with fog computing," in *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, IEEE, 2021, pp. 333–336.

[16] F. Pengfei, L. Miaomiao, L. Min, and L. Han, "Image encryption algorithm based on hyperchaotic system and dna coding," 2021.

[17] K. S. Kumari and C. Nagaraju, "Dna encrypting rules with chaotic maps for medical image encryption," n.d.

[18] D. N. R and N. M. Sivamangai, "A state-of-art model of encrypting medical image using dna cryptography and hybrid chaos map - 2d zaslavaski map: Review," 2022.

[19] M. Vaziri, M. M. Rafimifar, and H. Jahanirad, "An enhanced chaotic system based color image encryption using dna encoding," 2022.

[20] R. Amdouni, M. Gafsi, M. A. Hajjaji, and A. Mtibaa, "Combining dna encoding and chaos for medical image encryption," 2022.

[21] N. Srilatha and G. Murali, "Fast three level dna cryptographic technique to provide better security," n.d.

[22] D. Ravichandran, P. Praveenkumar, J. B. B. Rayappan, and R. Amirtharajan, "Dna chaos blend to secure medical privacy," 2017.

[23] P. S. Goswami, T. Chakraborty, and A. Chattopadhyay, "A secured quantum key exchange algorithm using fermat numbers and dna encoding," 2023.

[24] Y. Wu and S. Wan, "Medical image encryption by content-aware dna computing for secure healthcare," 2023.

[25] A. Majumder, A. Majumder, T. Podder, N. Kar, and M. Sharma. "Secure data communication and cryptography based on dna based message encoding." (2014), [Online]. Available: https://ieeexplore.ieee.org/document/7019464.

[26] "Chest radiograph." Accessed on 2023-09-18. (), [Online]. Available: https://en.wikipedia.org/wiki/Chest_radiograph#.

[27] G. Alvarez and S. Li, "Some basic cryptographic requirements for chaos-based cryptosystems," *International Journal of Bifurcation and Chaos*, n.d. DOI: 10.1142/S0218127406015970.

[28] D. N. R and N. M. Sivamangai, "A state-of-art model of encrypting medical image using dna cryptography and hybrid chaos map - 2d zaslavaski map: Review," *Journal Name*, 2022.

[29] A. S. Padmanabhan and S. Sapna, "Secure image transmission scheme based on dna sequences," Sep. 2022.

[30] M. Padmapriya and P. V. Eric, "A technique of data security using dna cryptography with optimized data storage," n.d. DOI: 10.33168/JSMS.2022.0425.

[31] X. Chai *et al.*, "An image encryption algorithm based on the memristive hyperchaotic system, cellular automata and dna sequence operations," *Signal Process. Image Commun.*, vol. 52, no. February, pp. 6–19, 2017.

[32] J. Liu *et al.*, "A new simple chaotic system and its application in medical image encryption," *Multimedia Tools and Applications*, vol. 77, no. 17, pp. 22 787–22 808, 2018.

[33] A. Belazi *et al.*, "Novel medical image encryption scheme based on chaos and dna encoding," *IEEE Access*, vol. 7, pp. 36 667–36 681, 2019.

[34] S. Stalin *et al.*, "Fast and secure medical image encryption based on nonlinear 4d logistic map and dna sequences (nl4dlm_dna)," *J. Med. Syst.*, vol. 43, no. 8, pp. 1–17, 2019.

[35] C. Rajvir *et al.*, "Image encryption using modified elliptic curve cryptography and hill cipher," in *Smart Innovation, Systems and Technologies*, vol. 159, New York: Springer, 2020, pp. 675–683.

[36] Y. Hui and H. Liu, "A dna image encryption based on a new hyperchaotic system," *Multimedia Tools and Applications*, pp. 1–25, 2021.

[37] R. Guesmi and M. Farah, "A new efficient medical image cipher based on hybrid chaotic map and dna code," *Multimedia Tools and Applications*, vol. 80, no. 2, pp. 1925–1944, 2021.

[38] X. Wang and Y. Li, "Chaotic image encryption algorithm based on hybrid multiobjective particle swarm optimization and dna sequence," *Opt. Lasers Eng.*, vol. 137, no. January 2020, p. 106 393, 2021.

[39] I. Aouissaoui, T. Bakir, and A. Sakly, "Robustly correlated key-medical image for dna-chaos based encryption," *IET Image Processing*, vol. 15, no. 12, pp. 2770–2786, 2021. DOI: 10.1049/ipr2.12261. [Online]. Available: https://doi.org/10.1049/ipr2.12261.