Thesis No: CSER-M-20-03

A Technique for Assuring Secrecy and Lossless Properties of Digital Image

By

Md. Siddiqur Rahman Tanveer

Department of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna 9203, Bangladesh
September, 2020

A Technique for Assuring Secrecy and Lossless Properties of Digital Image

By

Md. Siddiqur Rahman Tanveer

Roll No: 1707505

A Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Engineering in Computer Science and Engineering

Department of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna 9203, Bangladesh

Declaration

This is to certify that the thesis work entitled "**A Technique for Assuring Secrecy and Lossless Properties of Digital Image**" has been carried out by Md. Siddiqur Rahman Tanveer in the Department of Computer Science and Engineering, Khulna University of Engineering & Technology (KUET), Khulna 9203, Bangladesh. The above thesis work or any part of this work has not been submitted anywhere for the award of any degree or diploma.

29.09.2020

Signature of Supervisor
Dr. Kazi Md. Rokibul Alam
Professor
Dept. of Computer Science and Engineering,
Khulna University of Engineering & Technology

29.09.2020

Signature of Candidate
Md. Siddiqur Rahman Tanveer
Roll: 1707505
Dept. of Computer Science and Engineering,
Khulna University of Engineering & Technology

Approval

This is to certify that the thesis work submitted by Md. Siddiqur Rahman Tanveer entitled "**A Technique for Assuring Secrecy and Lossless Properties of Digital Image**" has been approved by the board of examiners for the partial fulfillment of the requirements for the degree of Master of Science in Engineering in the Department of Computer Science and Engineering, Khulna University of Engineering & Technology, Khulna, Bangladesh in September, 2020.

BOARD OF EXAMINERS

29.09.2020

1. ______________________________ Chairman
Dr. Kazi Md. Rokibul Alam (Supervisor)
Professor
Dept. of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna, Bangladesh

29/09/2020

2. ______________________________ Member
Head of the Department
Dept. of Computer Science and Engineering
Khulna University of Engineering & Technology

29/09/2020

3. ______________________________ Member
Dr. M. M. A Hashem
Professor
Dept. of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna, Bangladesh

29.09.2020

4. ______________________________ Member
Dr. K. M. Azharul Hasan
Professor
Dept. of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna, Bangladesh

29.09.2020

5. ______________________________
Dr. Md. Anisur Rahman Member
Professor (External)
Computer Science and Engineering Discipline
Khulna University
Khulna, Bangladesh

Acknowledgement

At first, I would like to thank Almighty for showering all his blessings on me whenever I needed. It is my immense pleasure to express my indebtedness and deep sense of gratitude to my supervisor Dr. Kazi Md. Rokibul Alam, Professor, Department of Computer Science and Engineering (CSE), Khulna University of Engineering & Technology (KUET) for his continuous encouragement, constant guidance and keen supervision throughout of this study. I am especially grateful to him for giving me his valuable time whenever I need and always providing continuous support in my effort.

I am especially grateful to all the faculty members of the Department of CSE, KUET to have their privilege of intensive, in-depth interaction and suggestions for the successful completion of my master degree.

At last I am grateful to my parents, family member and friends for their patience, support and encouragement during this period.

September, 2020 Author

Abstract

Due to various disease diagnosis, the volume of medical data is rising fast. Also, for telemedicine, while medical image transmits over the public network, the distortion of pixels may cause erroneous disease diagnosis. Here, encryption of the image by multiple chaos-based schemes along with DNA cryptography can be a safeguard. As chaotic schemes are very sensitive to the initial conditions, a small difference in the initial conditions yields entirely uncorrelated sequences that assure the strength of encryption. To get high randomness, several DNA encoding and computing rules are deployed. This thesis proposes a multi-stage chaotic encryption technique for the medical image through Logistic map along with Lorenz attractor and DNA cryptography, where both schemes possess the most significant value of control parameters. Thus, their consecutive deployment generates colossal chaotic sequences that ensure the robustness of the proposed technique. At first, the usage of the Logistic map with SHA-256 hash value generates a chaotic sequence that converts the plain medical image into a confusing image. Now, this sequence is used to create a confusion key to encrypt this blur image. Later on, to overcome the limitations of DNA computing rules and to get high randomness, encode this blur image and Lorenz attractor based key according to DNA encoding rules. These rules are determined randomly from eight encoding rules. Then, execute DNA operations between encoded blur image and Lorenz key using the four DNA computing rules and these rules are also determined by chaotic logistic sequence. Thus, the ultimate cipher is generated. Then, to approve the potency of the cipher, a randomness test according to NIST, security and statistical analyses and comparisons are performed.

Contents

LIST OF TABLES

LIST OF FIGURES

Nomenclature

AES	Advanced Encryption Standard
CT	Computed Tomography
CSO	Cuckoo search optimization
DCT	Discrete cosine transform
DES	Data encryption standard
DNA	Deoxyribo Nucleic Acid
DICOM	Digital Image and Communication in Medicine
DS	Digital signature
DWT	Discrete Wavelet Transform
FFT	Fractional Fourier Transform
GA	Genetic Algorithm
HIS	Hospital Information Systems
MRI	Magnetic resonance imaging
NIST	National Institute of Standards and Technology
OTP	One Time Pad
PSO	Particle Swarm Optimization
RSA	Rivest–Shamir–Adleman
USG	Ultrasonography
VC	Visual Cryptography
Key^{con}	Confusion Key
Key^{lor}	Lorenz key
D^{r}	DNA Encoding Rule
D^{op}	DNA Computing Operation Rule
h	Plain Image Hash
h'	Retrieved Image Hash
$C^{img}(M \times N)$	Cipher Image
$D^{img}(M \times N)$	Retrieved image
M	Height of the image
N	Width of the image
x	Logistic sequence

Seq_1, Seq_2, Seq_3	Lorenz sequences
a, c, b	Lorenz system parameters
n	Length of the Logistic sequence, $M \times N$
K	Logistic sequence key
t_1', t_2', t_3'	Initial values for secret key generation
r	Control parameter of Logistic map
A	Adenine, 00
G	Guanine, 10
C	Cytosine, 01
T	Thymine, 11
Permutation	Pixel position substitution
Diffusion	Pixel value modification
MSE	Mean square error
PSNR	Peak signal to noise ratio
D	Maximum deviation
I_D	Irregular deviation
S	Total key space
NPCR	Number of changing pixel rate
SPN	Salt & Pepper Noise
SN	Speckle noise
GN	Gaussian noise
UACI	Unified average changing intensity

CHAPTER I

Introduction

1.1 Background

In this era of e-health [1], the volume of medical data is supposed to reach about zettabytes by the year 2020 [2]. At the same time, the obligation of maintaining the privacy and security issues of medical data is soaring. Namely, in the branch of telemedicine [3], while medical image transfers between physicians and patients, its secure transmission over the public network, internet, etc. is essential. Besides, in applications like e-health, hospital information systems (HIS) [4], etc., ensuring the security of medical data is also expected. The underlying reason is medical image inherently associates aspects like authenticity, confidentiality, integrity [5], etc. Usually, medical evidence is private, as well as sensitive one. Also, a minor change of medical image may lead to wrong diagnostic outcomes. Further, maintaining the security of medical data is not only a moral issue but also a lawful obligation [6]. Hence, the protection of the medical image from any form of alteration and intrusion is essential.

Usually, medical image is distinct from conventional digital image. The standard medical image format for different diagnostic exams, namely, MRI, CT, X-ray, USG, etc. is Digital Image and Communication in Medicine (DICOM) [7]. Data contained by any DICOM file are header and pixel where the former one is like the text data, and DICOM itself maintains its confidentiality. In contrast, the security of the later one, i.e., pixel data that contains either image/short video or audio, is not provided by DICOM. Also, it holds strong correlations among adjacent pixels, high redundancy, large data capacity, etc. properties. Intuitively, many existing cryptosystems, e.g., OTP, visual cryptography (VC) [8], DES, watermarking, steganography, RSA, etc. are not competent enough for secure diffusion of medical image [34]. Alongside, some existing works have exploited Arnold’s cat map, DCT, DWT, fractional Fourier transform, etc. schemes to encrypt the DICOM image but they have drawbacks [41]. Namely, techniques proposed in [43], [49] etc. are possibly at risk [36]. In contrast, sensitive initial parameters, unpredictable, nonperiodic, ideal statistical, etc. merits of chaotic systems [38] enable them to design robust cryptosystems.

In this thesis, a multi-stage chaotic encryption along with DNA encoding technique has been developed. There are many excellent properties of DNA computing have been found: large-scale computational parallelism, tiny energy loss and huge storage space. From this point of view, the encryption algorithm proposed in this thesis combines DNA coding and computing rules selected randomly from chaotic map.

SHA-256 of the plain image is used to generate secret keys. A slight change in the plain image, SHA-256 yields a huge difference and enhances the sensitivity of the cryptosystem. Here, first, it adopts the Logistic map to generate a faster chaotic sequence that maintains ambiguity in confusion. Next, it uses Lorenz attractor to produce a superb chaotic behavior than any 1D or 2D chaotic map that ensures robustness in diffusion. Primarily, it generates the SHA-256 value of plain image to calculate the initial parameter of the chaotic Logistic sequence. Now, using external settings, it scrambles image pixels that erase the correlation between the adjacent pixels in the interim confused image.

Then, for further encryption, it applies confusion and Lorenz keys to conduct random DNA encoded and computing operations that ensure diffusion. Here, to generate different keys, it adopts distinct evasive skills. Namely, it develops logistic key from highly sensitive logistic sequence; confusion key by XOR-ing the logistic key with its singly shifted circular value, and Lorenz key from vast chaotic effect of Lorenz system. Thus, at the final stage of encryption, the usage of the Lorenz key, which is entirely independent of plain image, enables the technique to resist any form of security attack. Therefore, multi-stage security is ensured here to increase the level of secrecy.

1.2 Motivation

The chaotic system is a deterministic nonlinear system. It is highly sensitive to initial conditions, determinacy and so on [53-55]. Chaotic sequences yield pseudo-random sequences which are generated by chaotic maps. Their structures are very complex and difficult to analyze and predict. A slight change to initial condition leads to an uncorrelated sequence. Thus, the security of cryptosystems can be enhanced by the deployment of chaotic systems. The cryptography algorithms based on chaotic maps can be classified into two categories which are permutation and diffusion. In the permutation phase, the positions of pixels get changed by chaotic sequences or by some matrix transformation. This permutation algorithm has better encryption effect, but can’t change the pixel grey level value. Since pixels are not changed, leading to the histogram of the encrypted image and the original image being duplicates. That's why its security could be threatened by the statistical analysis. In the diffusion stage, the pixel values of the plain image are changed by chaotic sequences. Diffusion may lead to higher security, compared to permutation. Thereby, in order to improve the level of secrecy, some researchers have combined permutation and diffusion [35].

In 1994, Adleman [57] implemented the first experiment on DNA computing and it started a new era of information age. In further research, DNA computing, massive parallelism, massive storage and very-low power consumption these characteristics had been found. From this research on DNA computing, DNA cryptography emerged as a new cryptographic field, in which DNA is used as an information carrier and modern biological technology is used as implementation tool. For example, letter *A* is denoted by *CGA* and letter *B* is denoted by *CCA*. Then, the secret message can easily be encoded into a DNA sequence for an example, *AB* is expressed as *CCACGA*.

Permutation and diffusion mechanisms are performed by chaotic systems along with dynamic DNA encoding and computing enhance the level of secrecy of and can resist different types of malicious attacks. With a view to designing a robust cryptosystem, chaotic systems are incorporated with DNA cryptography in this thesis.

1.3 Problem Statement

In this modern era, medical images play very important role in diagnosis and treatment of diseases. For this important role in medical science they attract increasing attentions. Normally medical images involve the privacy of patients. Some images are very confidential and sensitive. If these private images get stolen, viewed or used by unauthorized accesses, disastrous incidents will occur. A hacker or a malicious database administrator may use the unauthorized images for their own benefits: medical marketing and fraudulent insurance claims. Thus, it may cause life threatening risks. In this regard, security of medical images is very important. Existing image encryption techniques have the following limitations.

- Traditional cryptosystems are not competent to encrypt digital medical images.
- Secrecy are breached while the underlying computational techniques are revealed and the diffusion mechanism is not properly maintained.
- Single chaotic map utilized to encrypt image may lead to a smaller key space and lower security.

- Often they treat DNA bases statically.

1.4 Objectives of the Thesis

To ensure more secrecy of medical images a multistage encryption technique is proposed. Chaotic schemes are very sensitive to initial parameters and can improve the level of secrecy. Also dynamic DNA encoding and computing mechanisms determined by chaotic map help to perform proper diffusion. These cryptographic building blocks have been exploited in this proposed encryption technique to increase the secrecy level of the medical images and to resist various types of malicious attacks. To reach the goal following issues will be considered.

- To ensure more security over traditional cryptosystems.
- To adopt dynamic encoding and computing mechanisms for DNA bases to enrich secrecy.
- To exploit a multi-stage chaotic cryptosystem to maintain robustness.
- To resist different types of malicious attacks.

1.5 Organization of the Thesis

To ensure secrecy of data the dynamic mechanisms is proposed. The thesis is organized as below.

- **Chapter I** briefly explains the introduction of the thesis and motivation of works. Problem statement, objectives of this thesis and contributions are also discussed elaborately here.
- **Chapter II** represents the existing works in the related field and focuses on the advantages and drawbacks of existing works.
- **Chapter III** discusses the cryptographic building blocks of the thesis.
- **Chapter IV** explains the proposed cryptographic technique. The procedure of ensuring multistage security of digital image through chaotic schemes and dynamic DNA encoding and computing mechanisms is described here elaborately. Every step of encryption and decryption also discussed in this chapter.
- **Chapter V** explains the experimental and security analysis. This chapter also explains the experimental setup and the analyzing results and performance that compares with existing techniques.

- **Chapter VI** concludes this thesis together with the outline of probable future directions of research opened by this work.

CHAPTER II

Literature Review

2.1 Introduction

Nowadays, considering the security of the digital image has become an essential issue especially for medical (DICOM) images. Many traditional cryptosystems are not capable enough to encrypt the digital images image due to their intrinsic characteristics. Analyzing the existing works, the related works can be majorly categorized in the following types:

- Digital Watermarking [9, 52]
- Compression based methods [10, 11]
- Nature Inspired optimization based encryption technique [12-14]
- Chaos based encryption technique [15-18, 35-37]

2.2 Digital Watermarking

An early–timed technique proposed in [9] aimed to encrypt the DICOM image while exploiting traditional cryptosystems. Initially, it creates a digital signature (DS) using a hash value generated from the plain image. Now, this DS and DICOM header are combined to produce an invisible watermark. Then, watermark is embedded in the background of the image. Finally, AES and RSA are applied to encrypt this watermarked image. Although, it can encrypt only the information of a patient, cannot encrypt pixels of the DICOM image portion.

Based on watermarking, the technique proposed in [52] deals with the encryption of medical big data. It consists of several steps. First, it applies a hyper-chaotic system in the 3D-DCT domain to encrypt the big image. From here, it extracts feature vector in the domain of 3D-DFT. Finally, it associates this feature vector with watermark. Herein due to the exploitation of zero-watermarking technique, watermark embedding cannot change the encrypted data. However, this technique is not so robust, only can resist normal as well as geometric attacks. Therefore, the design of a multi-stage chaos-based technique can be alluring to ensure the secrecy of the DICOM image.

2.3 Compression based methods

Employing compression-based methods like DCT, DWT, DFT, etc. techniques proposed in [10, 11] aimed to encrypt the DICOM image. Through low computational complexity, they targeted to attain high-level security. However, the quality of the plain image cannot be maintained due to limited computer calculation precision, while performing transformation and inverse transformation. Hence, during decryption; the quality of the retrieved image impairs from the original image.

2.4 Nature Inspired optimization based hybrid encryption technique

For DICOM image encryption, hybrid techniques namely, VC with Gaussian based modified cuckoo search optimization (CSO) [12], particle swarm optimization (PSO) [13] and genetic algorithm (GA) [14] etc. also have been deployed. The technique proposed in [12] first applies DWT for partitioning the plain image into blocks. Then the utilization of Gaussian based CSO selects the optimal position for every block. Later on, it creates secret shares using VC to embed them in the regarding positions of the blocks. Besides, due to secret shares, image pixels' quality is degraded in decrypted image. Thus, the sensitiveness of the medical image is hampered. Regarding the technique proposed in [13, 14], the common problem of optimization algorithms is: they may converge to the local minima. Hence, they may not be able to reach an optimized solution.

2.5 Chaos based encryption technique

The technique proposed in [15] employed Arnold's cat map to obtain the permutation sequences for shuffling square sized medical images. The additional procedure of this image encryption technique is that it reshaped a non-square image into a square image. Further, the pseudorandom numbers are generated in a range between 0 and 255, which would be easy for a cryptanalyst to breach it. To encrypt the whole image efficiently, it requires many rounds. Besides, Arnold's cat map scheme doesn't perform the alteration of image pixel values. Therefore, its' security level is not so high.

Considering permutation, diffusion, substitution, etc. operations of chaos, several DICOM image encryption techniques are available in [16-18]. The problems associated with traditional chaos-based image encryption techniques are: easily analyzable and predictable, and employment of non-linear prediction technique, comparatively easily an intruder can mount various attacks, etc. However, the Latin square and chaos-based technique proposed in [35] assures high sensitivity of the plain medical image. Another technique proposed in [36] first inserts random values and then executes high-speed pixel scrambling and adaptive diffusion to protect any representation format of the medical image. Besides, summarizing the design of image encryption techniques including chaotic schemes developed in the year 2018, a review is available in [37].

2.6 Summary

The traditional cryptographic techniques are not suitable for medical image encryption. The compression based image encryption can ensure the secrecy but they lack the image quality while decryption. That’s why the sensitive images fall in danger due to data loss. The nature inspired optimized based hybrid encryption algorithms have the risk to converse into local optima. From this perspective optimization based algorithms can’t provide proper outcomes. Digital watermarking is used to provide copyright and theft protection of the digital images but their underlying mechanism can’t diffuse the pixel of the image. Traditional chaotic image encryption is faster but easily analyzable and predictable, and employment of non-linear prediction technique, comparatively easily an intruder can mount various attacks.

Unlike the above techniques, the technique proposed in this thesis consecutively exploits multiple chaotic schemes for this purpose. At first, using the Logistic map along with the SHA-256 value of the plain medical image; it generates a chaotic sequence. Now, it transforms the image into a blurred image. Then, according to this sequence, it produces a confusion key which is applied to encrypt this confusing image. Lastly, the deployment of the Lorenz attractor along with dynamic DNA encoding and computing operations generates the final cipher image.

CHAPTER III

Cryptographic Building Blocks

3.1 Introduction

This section describes the required cryptographic building blocks, *i.e.*, SHA-256, Logistic map, Lorenz attractor and DNA encoding and computing operations that are needed to develop the proposed medical image encryption technique.

3.2 SHA-256

The primary purpose of hash functions is to provide integrity in security services. A widely known cryptographic hash function is SHA-256. Generally, it generates a 256-bit hash value. Typically, this value is represented as a 64 digit hexadecimal number. It leads a significant difference between two images if there happens even one-bit alteration in the input of hash value. Thus, it provides an acceptable security feature. In this paper, 256 bits are generated using this hash function from plain DICOM image. It is split into 8-bit blocks H_i expressed as $H = H_1H_2H_3\ldots\ldots\ldots H_{32}$ and each block H_i is partitioned as

$$\begin{cases} t_1 = t_1' + \frac{H_1 \oplus H_2 \oplus H_3 \oplus \ldots\ldots\ldots\ldots \oplus H_{11}}{256} \\ t_2 = t_2' + \frac{H_{12} \oplus H_{13} \oplus H_{14} \oplus \ldots\ldots\ldots\ldots \oplus H_{22}}{256} \\ t_3 = t_3' + \frac{H_{23} \oplus H_{24} \oplus H_{25} \oplus \ldots\ldots\ldots\ldots \oplus H_{32}}{256} \end{cases} \tag{3.1}$$

where t_1', t_2' and t_3' are the given initial values which are used to calculate t_{avg} as

$$t_{avg} = \frac{t_1 + t_2 + t_3}{3} \tag{3.2}$$

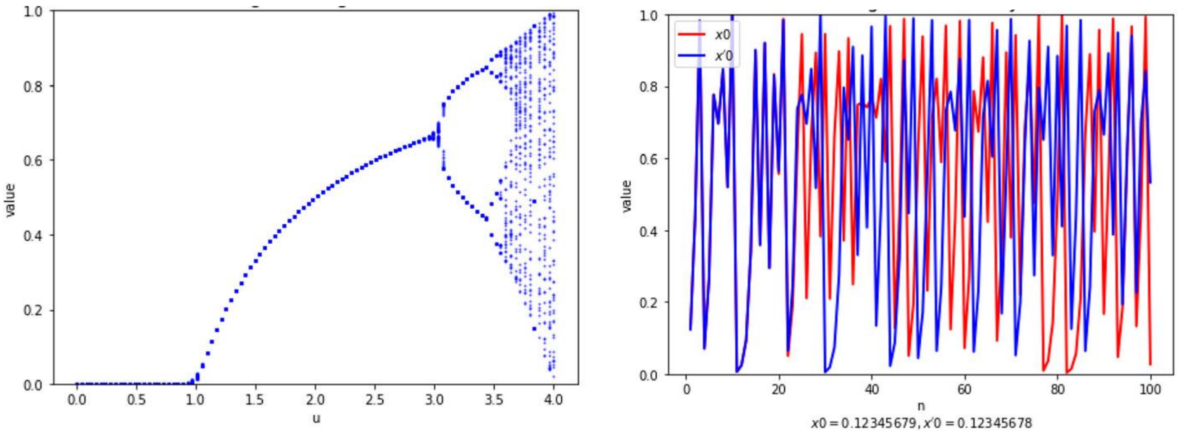
and it is used to generate the Logistic sequence.

3.3 Logistic Map

The widely-known Logistic map used in cryptography is the 1D chaotic map. The map was first popularized by the biologist Robert May [19]. Mathematically, the Logistic equation is written as

$$x_{p+1} = r * x_p(1 - x_p) \tag{3.3}$$

where, $x_p \in (0, 1)$ is the 1D discrete state that starts from one initial condition $x_1 \in (0, 1)$ set by t_{avg}. It has a control parameter r with an interval between 3.75 to 4 [42] that produces a highly random sequence. Hence, it assures the guarantee of the faster chaotic sequences. Here, $p = 1, 2, \ldots, N$ is the number of iterations.



(a) Bifurcation diagram of the Logistic Map

(b) Logistic Sensibility

Figure 3.1: Chaotic characteristics of logistic map

The bifurcation diagram and key sensibility of chaotic logistic map are presented in Fig. 3.1. The logistic map sensibility shown in Fig. 3.1 (b) for two initial values with slight difference which are $x0 = 0.12345679, x'0 = 0.12345678$.

3.4 Lorenz Attractor

To present the thermally induced fluid convection in the atmosphere, E.N Lorenz [20] first reported a mathematical model known as Lorenz attractor. It consists of three ordinary differential equations. Scientific studies on this model show that it has two wings, which looks like the butterflies [21]. Having this “butterfly effect,” shown in Fig. 3.2 this model has been extensively studied in numerous applications, namely, chaos-based theory, modeling of dynamic systems, and chaotic control based synchronization. Among all classical chaotic schemes, the Lorenz attractor equation represents a 3D dynamic equation

$$\begin{aligned} \bar{x} &= a(y - x) \\ \bar{y} &= x(c - z) - y \\ \bar{z} &= xy - bz \end{aligned} \tag{3.4}$$

where, x, y and z are the functions with respect to time of which derivative forms are $\bar{x}$, $\bar{y}$, and $\bar{z}$. Also, a, c, and b are the system parameters.

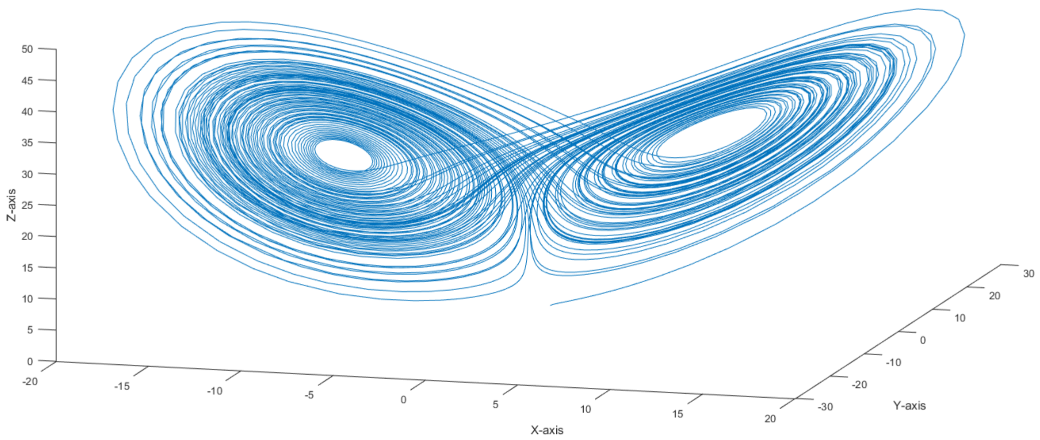


Figure 3.2: Butterfly effect of Lorenz attractor for $a = 10; c = 8/3; b = 28$

The Lorenz attractor presents much complicated chaotic behavior than any other 1D or 2D chaotic schemes.

3.5 DNA Encoding and Computing Operations

There are four DNA deoxynucleotides which are A (adenine), G (guanine), C (cytosine), T (thymine), bases. Among them G and C are complementary, so are A and T. Normally in binary system, 0 and 1 are complement to each other. Hence, 00, 11, 01, 10 can be encoded into the four bases. According to combinatorics there are 24 kinds possible DNA encoding methods. Due to complementary relationship between the four only 8 coding combinations are effective, as listed in Table 3.1.

Table 3.1: DNA encoding and decoding rules

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

Table 3.2: Addition operation

+	**A**	**C**	**T**	**G**
A	A	A	T	T
C	C	A	G	T
T	T	G	A	C
G	G	T	C	A

Table 3.3: XOR operation

⊕	**A**	**C**	**T**	**G**
A	A	A	T	T
C	C	A	G	T
T	T	G	A	C
G	G	T	C	A

Table 3.4: Subtraction operation

–	**A**	**C**	**T**	**G**
A	A	A	T	T
C	C	A	G	T
T	T	G	A	C
G	G	T	C	A

Table 3.5: XNOR operation

⊙	**A**	**C**	**T**	**G**
A	A	A	T	T
C	C	A	G	T
T	T	G	A	C
G	G	T	C	A

In image encryption, the gray value of the image pixel can be expressed as its corresponding binary sequence, and then this binary sequence can easily be encoded into a DNA sequence. On the other hand, a DNA sequence can easily be translated into a pixel value. For example: a pixel value is 196 and its binary sequence 11000100. It can be encoded into a DNA sequence GCAC using DNA encoding Rule 5. And applying DNA decoding rule 7 on this sequence the retrieved pixel value is 55. Moreover, different operations have been applied on DNA sequence to encrypt the image. The details of the addition, XOR, subtraction and XNOR DNA operations rules are shown in the following tables, Table 3.2 to Table 3.5.

3.6 Encryption architecture of permutation and diffusion

In 1998, Fridrich [56] proposed the permutation-diffusion architecture for chaos-based image cryptosystem and it is shown in Fig. 3.3. This architecture consists of two mechanisms: permutation and diffusion.

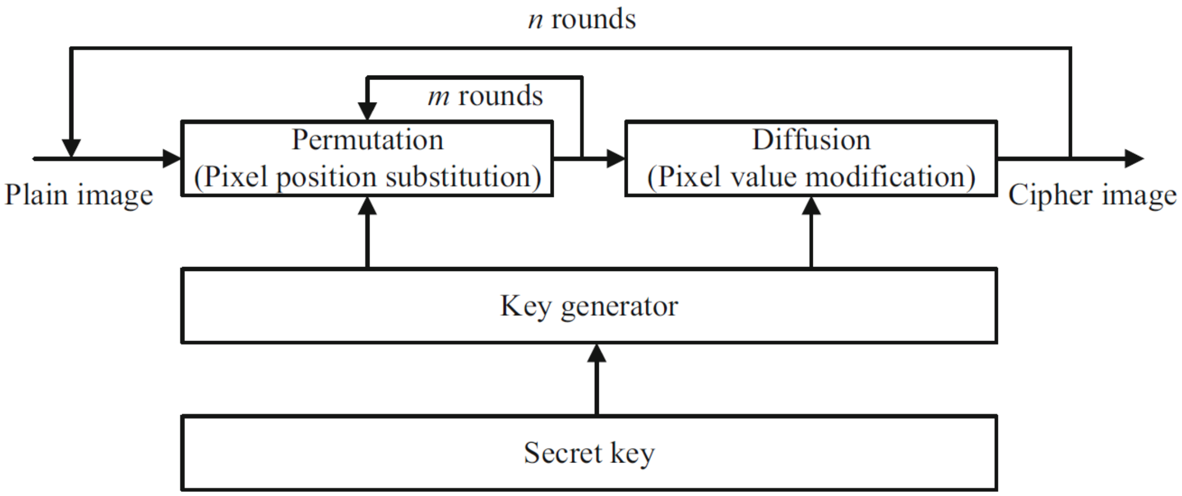


Figure 3.3: Encryption architecture of permutation and diffusion

In permutation phase, image pixels are shuffled by a two-dimensional area using chaotic map, such as Arnold cat map and Baker map. Then, in the diffusion phase, the pixel values are modified. In this modern time, Fridrich’s architecture has become the most popular and has been widely used in many chaos-based image cryptosystems.

In this thesis the permutation is performed by pixel scrambling using the logistic map that yields the confused image. After that dynamic DNA encoding and computing mechanism is deployed that modifies the pixel values and produces the ultimate cipher image.

CHAPTER IV

Proposed Technique to Ensure Secrecy and Lossless Property of Image

4.1 Introduction

This thesis combines the concepts of multi-stage chaotic encryption along with dynamic DNA encoding and computing mechanism. The proposed technique is designed based on the multi-stage encryption paradigm. It consists of three major stages:

- Key generation
- Encryption process
- Decryption process,

They are described below.

4.2 Key generation

Here, Logistic map-based confusion key Key^{con} and Lorenz attractor based chaotic key Key^{lor} are generated to conduct both encryption and decryption operations.

4.2.1 Confusion key $\boldsymbol{Key^{con}}$ generation

At first, the Logistic map-based confusion key Key^{con} is generated as follows. Fig. 4.1 depicts the process.

Step 1: Consider a plain medical image $I(M \times N)$, where M and N represent the height and width. Now, t_{avg} is calculated for the given initial values of t_1', t_2', and t_3'.

Step 2: Set $x_1 = t_{avg}$ to generate the Logistic sequence $x = [x_1, x_2, x_3, \dots \dots \dots \dots, x_n]$ of which the length n is $M \times N$.

Step 3: While encryption, the restriction of the generated Logistic sequence key value between [0, 255] is maintained as

$$K_i = abs\big(round((cos(p \times cos^{-1}(x_i))) \times 255)\big) \tag{4.1}$$

where x_i = $[x_1, x_2, x_3, \dots \dots \dots \dots, x_n]$ is the sequence, and the length of K is $M \times N$.

Step 4: Perform single time circular shift operation on K to generate K' as

$$K' = cirshift(K, 1) \tag{4.2}$$

Step 5: After this, perform XOR operation between K and K' to generate the confusion key Key^{con} of which the length is also $M \times N$, using

$$Key^{con} = (K \oplus K') \tag{4.3}$$

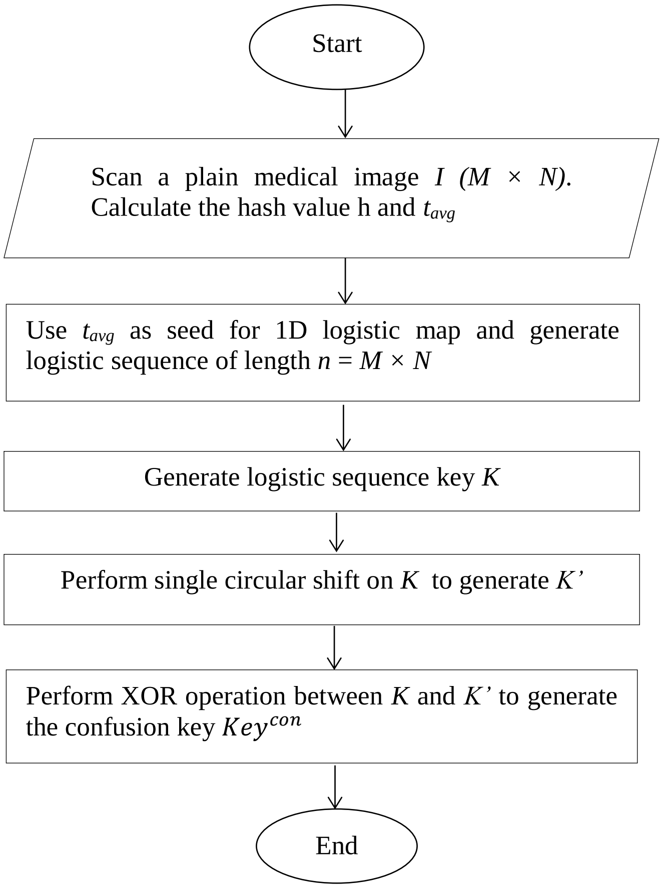

Start
Scan a plain medical image I $(M \times N)$. Calculate the hash value h and t_{avg}
Use t_{avg} as seed for 1D logistic map and generate logistic sequence of length $n = M \times N$
Generate logistic sequence key K
Perform single circular shift on K to generate K'
Perform XOR operation between K and K' to generate the confusion key Key^{con}
End

Figure 4.1: Confusion Key Key^{con} generation process.

After confusion key generation process, Lorenz key Key^{lor} is generated as follows.

4.2.2 Lorenz key Key^{lor} generation

Lorenz key Key^{lor} generation process is described below and Fig. 4.2 represents the process.

Step 1: At first, for the initial values of t_1', t_2' and t_3' and for the conditions of the system parameters a, c and b; the Lorenz scheme represented in chapter III (3.4) is executed. The result of the first 3400 iterations is ignored to remove the transient effect and to attain a good pseudo-random sequence. After that, the system is executed as $n = M \times N$ times, which creates three sequences as

$$Seq_1 = [\bar{x}_1, \bar{x}_2, \bar{x}_3 \dots \dots \dots .., \bar{x}_n] \tag{4.4}$$

$$Seq_2 = [\bar{y}_1, \bar{y}_2, \bar{y}_3 \dots \dots \dots .., \bar{y}_n] \tag{4.5}$$

$$Seq_3 = [\bar{z}_1, \bar{z}_2, \bar{z}_3 \dots \dots \dots .., \bar{z}_n] \tag{4.6}$$

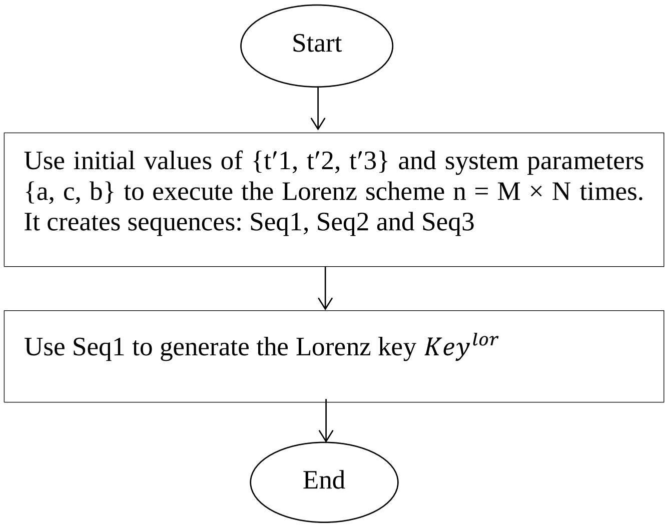

Start
Use initial values of {t′1, t′2, t′3} and system parameters {a, c, b} to execute the Lorenz scheme n = M × N times. It creates sequences: Seq1, Seq2 and Seq3
Use Seq1 to generate the Lorenz key Key^{lor}
End

Figure 4.2: Lorenz Key Key^{lor} generation process.

Step 2: From these sequences, Seq_1 is chosen to generate the Lorenz key. To keep the key values between [0, 255], it performs

$$Key_i^{lor} = floor(mod\ ((abs(Seq_1(i)) - floor(abs(Seq_1(i)))) \times 10^{14}\ , 256)) \tag{4.7}$$

where, i = 1, 2, … …, n. Thus, the Lorenz encryption key Key^{lor} with the length of $M \times N$ is generated.

4.3 Encryption Process

Already in chapter III (3.2), the calculation of t_{avg} has been discussed which is used to produce the Logistic sequence. Then the sequence is used to create the confusion image.

Secondly, the first stage encryption key is generated to encrypt this confusing image. Thirdly, the Lorenz key is generated and dynamic DNA encoding and computing rules determined by the logistic sequence is deployed between the confusing image and Lorenz key to produce the final encryption. The encryption process is shown in Fig. 4.3 and is described below.

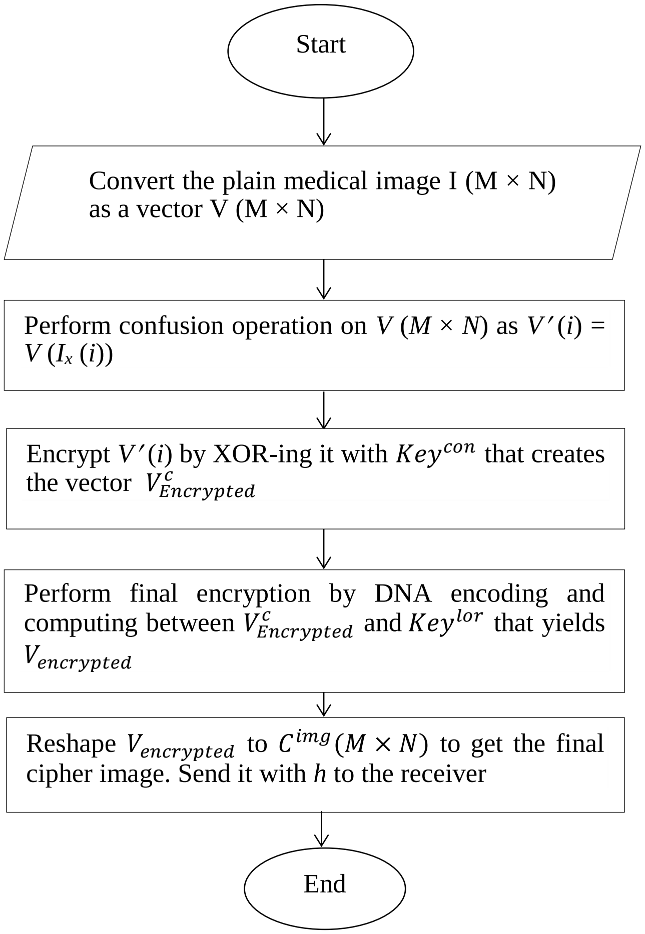

Start
Convert the plain medical image I (M × N) as a vector V (M × N)
Perform confusion operation on V (M × N) as $V'(i)$ = V (I_x (i))
Encrypt $V'(i)$ by XOR-ing it with Key^{con} that creates the vector $V^{c}_{Encrypted}$
Perform final encryption by DNA encoding and computing between $V^{c}_{Encrypted}$ and Key^{lor} that yields $V_{encrypted}$
Reshape $V_{encrypted}$ to $C^{img}(M \times N)$ to get the final cipher image. Send it with h to the receiver
End

Figure 4.3. Encryption process.

Step 1: Convert the plain medical image $I(M \times N)$ as a vector $V(M \times N)$.

Step 2: Perform the following operation on the generated Logistic sequence $x = [x_1, x_2, x_3, \dots \dots \dots \dots, x_n]$ described in chapter III (3.3) by the equation

$$(l_x, f_x) = sort(x) \tag{4.8}$$

where $sort()$ is the sequencing index function. It sorts x in ascending order, l_x and f_x represent the new sequence and the index value, respectively. Now the confusion is performed by

$$V^{'}(i) = V(l_x(i)) \tag{4.9}$$

Step 3: Encrypt the new vector V' using the confusion key Key^{con} according to

$$V^{c}_{Encrypted} = V' \oplus Key^{con} \tag{4.10}$$

Step 4: Then, Then encode the $V^{c}_{Encrypted}$ and Key^{lor} according to DNA encoding rules and perform final encryption using DNA computing by the Lorenz encryption key Key^{lor} as already discussed in chapter III (3.4), i.e., encrypt $V^{c}_{Encrypted}$ using this key

$$D^{r} = \lfloor x \times 8 \rfloor + 1 \tag{4.11}$$

$$D^{op} = \lfloor x \times 3 \rfloor + 1 \tag{4.12}$$

$$V_{encrypted} = \text{Enc}\left(V^{c}_{Encrypted}, D^{r}\right) D^{op} \text{Enc}(Key^{lor}, D^{r}) \tag{4.13}$$

In equation (4.11), D^{r} is the selected type of DNA rule shown in Table 3.1, which is determined according to the logistic sequence x. D^{op} denotes the DNA computing (+, XOR, -, XNOR) presented in Table 3.2 to table 3.5 are also determined from logistic sequence too. In equation (4.13) $Enc()$ performs the DNA encoding according to D^{r}.

Step 5: Finally, reshape $V_{encrypted}$ to $C^{img}(M \times N)$ matrix, which is the final cipher image. Now send the cipher image and hash value h of the plain image to the receiver.

4.4 Decryption Process

Although decryption operations are the reverse process of encryption operations, this section describes them. Before this stage, the receiver obtains the required parameters already. Using them, it generates the secret keys by itself to decrypt the cipher image. The decryption process is shown in Fig. 4.4.

Step 1: Convert the cipher image $C^{img}(M \times N)$ to a vector $V_{encrypted}$ of length $M \times N$.

Step 2: From the sender's hash value h and the initial value of t'_1, t'_2 and t'_3, calculate the value of t_{avg} and generate the Logistic sequence $x = [x_1, x_2, x_3, \dots \dots \dots \dots, x_n]$, $n = M \times N$. Generate the Lorenz key Key^{lor} according to the value of t'_1, t'_2 and t'_3 and perform the DNA encoding and computing rules determined by the logistic sequence according to equation (4.11) and (4.13).

$$V^c_{Decrypted} = \text{Enc}(V_{encrypted}, D^r)\ D^{op}\ \text{Enc}(\ Key^{lor}, D^r) \tag{4.14}$$

that retrieves the vector $V^c_{Decrypted}$.

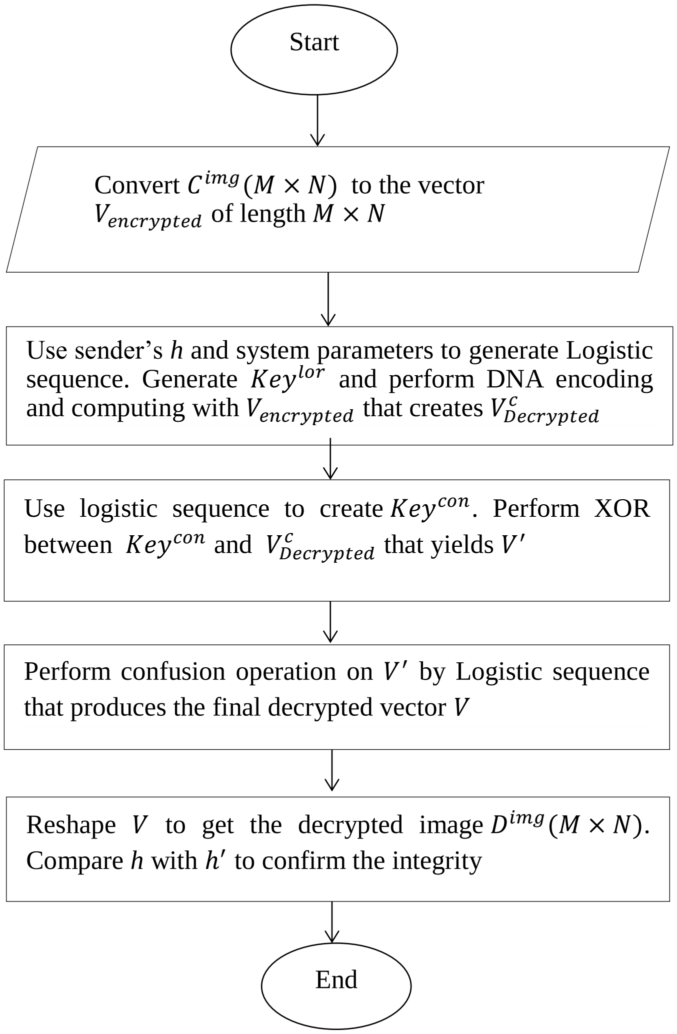

Start
Convert $C^{img}(M \times N)$ to the vector $V_{encrypted}$ of length $M \times N$
Use sender's h and system parameters to generate Logistic sequence. Generate Key^{lor} and perform DNA encoding and computing with $V_{encrypted}$ that creates $V^c_{Decrypted}$
Use logistic sequence to create Key^{con}. Perform XOR between Key^{con} and $V^c_{Decrypted}$ that yields V'
Perform confusion operation on V' by Logistic sequence that produces the final decrypted vector V
Reshape V to get the decrypted image $D^{img}(M \times N)$. Compare h with h' to confirm the integrity
End

Figure 4.4: Decryption process.

Step 3: Now the logistic sequence is used to create the confusion key Key^{con}. Then, perform the XOR operation with $V^c_{Decrypted}$ by

$$V' = V^{c}_{Decrypted} \oplus Key^{con} \tag{4.15}$$

that yields the vector V'.

Step 4: Now, using the Logistic sequence $x = [x_1, x_2, x_3, \dots\dots\dots\dots, x_n]$, perform the $sort()$ operation same as Step 2 of chapter IV (4.3). Then perform the confusion operation using this sorted sequence on V' by

$$V(i) = V'(l_x(i)) \tag{4.16}$$

that produces the final decrypted vector V.

Step 5: Finally, reshape V to $D^{img}(M \times N)$ matrix, which is the retrieved image. Now generate the hash value h' for this image. Check it with the senders' hash value h. If both are same, it confirms the integrity.

CHAPTER V

Experimental and Security Analyses

5.1 Experimental Setup

To evaluate the performance of the proposed technique, a prototype has been developed and analyzed exploiting an environment based on 64-bit Windows 10 operating system. The configuration of the environment is Intel$^{(R)}$ Core^{TM} i7-4790 3.60 GHz CPU with 8 GB RAM. MATLAB R2015a has been used to develop the prototype. Here, the standard size, *i.e.*, 1024×1024 medical images collected from [22-25] are used for experiments. Besides, for comparison, Lena 256×256 grayscale image is used where parameters related to performing encryption are set as $r = 3.998,\ p = 3.628,\ a = 10,\ \mathrm{b} = 28, \mathrm{c} = 8/3,\ \ t_1'$= 0.5346, t_2'= 0.4846, and t_3'= 0.6969.

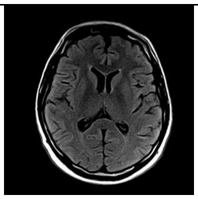
5.2 Experimental Results

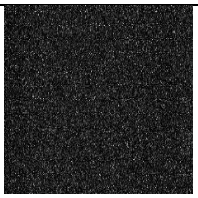
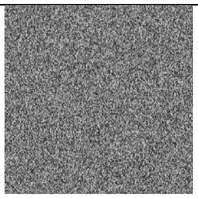
Experimental results consists of these: Output of the Encryption Stage, Output of the Decryption Stage, Histogram Analysis, Information Entropy, Correlation between Two Adjacent Pixels, MSE and PSNR Analysis, Maximum Deviation, Irregular Deviation, Energy Analysis, Contrast Analysis and Efficiency Analysis. They are described below.

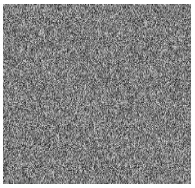
5.2.1 Output of the Encryption Stage

Based on chapter IV (4.3), Table 5.1 shows the step by step output of the encryption process.

Table 5.1: Output of the Encryption Stage Including Key Generation for MRI Image

Step	Operation	Output
Step 1	Plain MRI image (1024×1024)	

Step 2	*Generate hash h by using SHA-256	a5af04c49ebc0eaa4ed2c8cb01c4aa84390ea4933b39139c9a
Step 3	Calculate t_{avg} for initial parameters	0.400791503267974
Step 4	*Generate Logistic sequence X of size 1024×1024 by using t_{avg} and r	X=[0.400791503267974,0.960150381356334,0.152969982897406,……….,0.15108614]
Step 5	*Generate Logistic sequence key K of size 1024×1024	K=[161,254,33,126,198,225,250,76,221,78,217,......,12, 185]
Step 6	*Single time circular shift of K that produces K'	K′=[185,161,254,33,126,198,225,250,76,………...,140,12]
Step 7	*XOR K and K' to create the confusion key Key^{con}	Key^{con}=[24,95,223,95,184,39,27,182,145,147,………..,181]
Step 8	*Generate the Lorenz key Key^{lor} of size 1024×1024	Key^{lor}=[83,74,35,37,104,82,4,114,105,142,……..,18,37,0]
Step 9	*Convert the plain image to a vector V of size 1024×1024	V=[0,…,8,16,25,29,28,20,…,110,106,111,110,92,…...,0]
Step 10	Create a confused image by pixel scrambling by the Logistic sequence from V	
Step 11	Encrypt this confused image with Key^{con} to produce another interim cipher image	
Step 12	Encrypt this cipher with Key^{lor} to produce the final cipher image. Send it along with h to the receiver	Interim Cipher Image Pixel(1,1) = 201 $Key^{lor}(1)$ = 83 X(1) = 0.400791503267974 $D^{r} = \lfloor 0.400791503267974 \times 8 \rfloor + 1 = 4$ **DNA encoding rule 4 : 00-T, 01-G, 10-C, 11-A** $D^{op} = \lfloor 0.400791503267974 \times 3 \rfloor + 1 = 2$

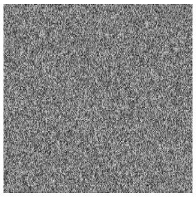
		DNA computing rule 2: XOR operation imgDNA = Enc$(201, D^r)$ = ATCG keyDNA = Enc$(83, D^r)$ = GGTA DNAOperation(D^{op},keyDNA,imgDNA) = GCGG Encrypted pixel value = **101** 

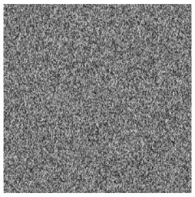
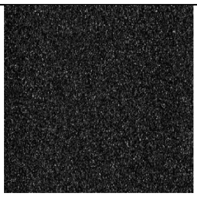
* Only a portion of data is shown

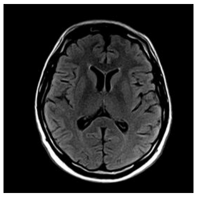
5.2.2 Output of the Decryption Stage

Based on chapter IV (4.4), Table 5.2 shows the step by step output of the decryption process.

Table 5.2: Output of the Decryption Stage Including Key Exploitation for MRI Image

Step	Operation	Output
Step 1	Final cipher image	
Step 2	*The sender's hash *h*	a5af04c49ebc0eaa4ed2c8cb01c4aa84390ea4933b39139c9a
Step 3	Calculate t_{avg} for initial parameters	0.400791503267974
Step 4	*Generate Logistic sequence *X* of size 1024×1024 by using t_{avg} and *r*	*X*=[0.400791503267974,0.960150381356334,0.152969982897406,………,0.15108614]
Step 5	*Generate Logistic sequence key *K* of size 1024×1024	*K*=[161,254,33,126,198,225,250,76,221,78,217, …, 12,185]

Step 6	*Single time circular shift of K that produces K'	K'=[185,161,254,33,126,198,225,250,76,………..., 140,12]
Step 7	*XOR K and K' to create the confusion key Key^{con}	Key^{con}=[24,95,223,95,184,39,27,182,145,147,……..,181]
Step 8	*Generate the Lorenz key Key^{lor} of size 1024×1024	Key^{lor}=[83,74,35,37,104,82,4,114,105,………….,18,37,0]
Step 9	Decrypt final cipher image with Key^{lor} to retrieve the interim cipher image	Cipher Image Pixel(1,1) = 101 $Key^{lor}(1)$ = 83 X(1)=0.400791503267974 $D^r \ = \lfloor 0.400791503267974 \times 8 \rfloor + 1 = 4$ **DNA encoding rule 4 : 00-T, 01-G, 10-C, 11-A** $D^{op} = \lfloor 0.400791503267974 \times 3 \rfloor + 1 = 2$ **DNA computing rule 2: XOR operation** imgDNA = Enc$(101, D^r)$ = GCGG keyDNA = Enc$(83, D^r)$ = GGTA DNAOperation(D^{op},keyDNA,imgDNA) = ATCG Encrypted pixel value = **201** 
Step 10	Decrypt the interim cipher with Key^{con} that yields the confused image	
Step 11	*Perform pixel scrambling using Logistic sequence and convert the image to a vector V of size 1024×1024	*V*=[0,…,8,16,25,29,28,20,…,110,106,111,110,92,…...,0]

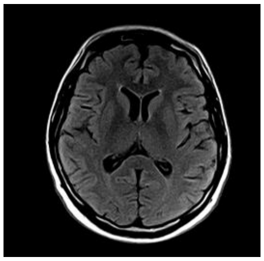
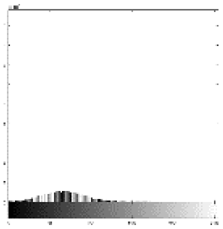
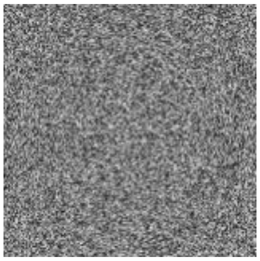
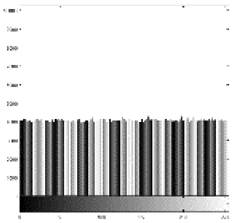
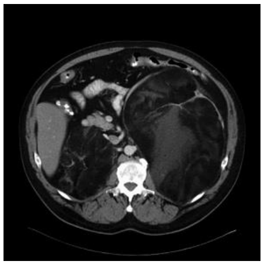
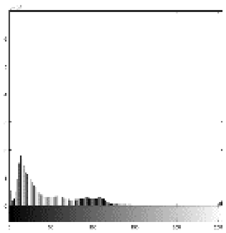
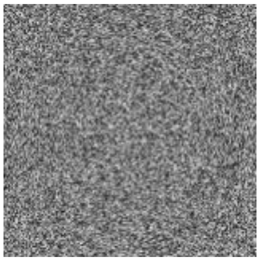
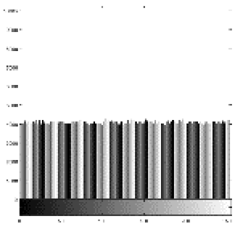
Step 12	Reshape V to retrieve the plain image. Generate hash h' of this image, compare it with h to check the integrity.	

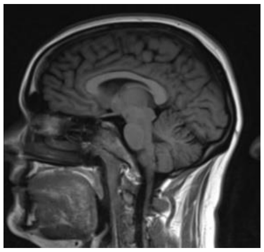
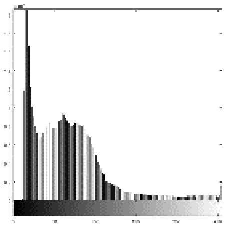
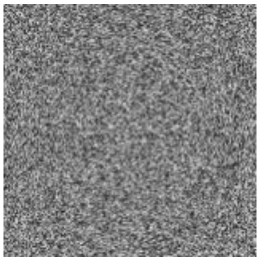
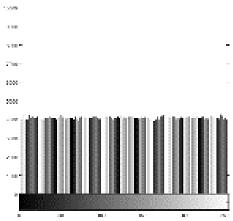
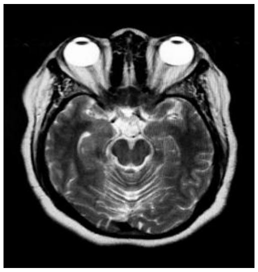
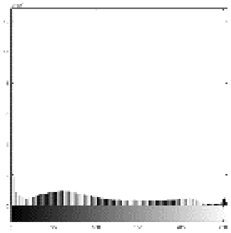
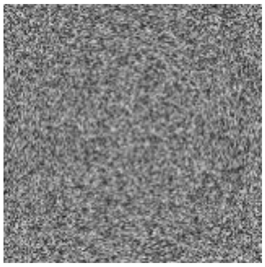
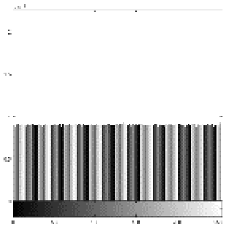
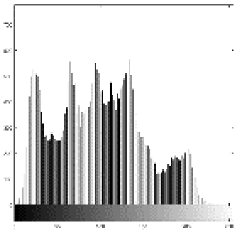
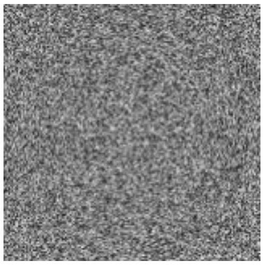
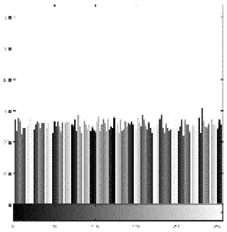
[*] Only a portion of data is shown

5.2.3 Histogram Analysis

Histogram analysis shows the pixels' distribution within an image that has a significant characteristic for image analysis [41]. Uniform frequency distribution is an important characteristic of the ideal cipher image. For chosen medical images, corresponding histograms of the plain and cipher images are shown in Table 5.3. These histograms noticeably show that the cipher images are uniform and random. That's why any useful statistical information cannot be derived from the cipher image generated by the proposed technique.

Table 5.3: Histograms of plain Lena and medical images [22-25] and their cipher images

Item	**Plain image**	**Histogram**	**Cipher image**	**Histogram**
(a)				
(b)				

(c)				
(d)				
(e)				

5.2.4 Information Entropy

Information entropy is a significant feature, which plays a vital role to measure the randomness of any cipher image. It indicates the uncertainty of information. It is calculated as [36]

$$E(m) = -\sum_{j=0}^{M-1} p(m_j) log_2 p(m_j) , \sum_{i=0}^{M-1} p(m_j) = 1 \qquad (5.1)$$

where, m_j and $p(m_j)$ stand for gray level value and probability, respectively. It is known that the ideal entropy for any cipher image having 256 gray levels should be 8. A value of entropy closer to its ideal value ensures that the gray values are more uniformly distributed.

Table 5.4: The entropy analysis for images in Table 5.3 and comparison with other techniques

Images of Table 5.3	**Entropy**
(a)	7.99982

Images of Table 5.3	**Entropy**
(b)	7.99982
(c)	7.99981
(d)	7.99992
For Lena image a comparison with other techniques	
Proposed	7.9973
[26]	7.9891
[27]	7.7626

As shown in Table 5.4, the average information entropy of cipher images (*i.e.*, the 4th column of Table 5.3) for the proposed technique is 7.99984, which is very close to the ideal value. Thus, the proposed technique shows high randomness of cipher images. Also, considering the Lena image, a comparison with other techniques is shown in the same table. It shows that the proposed technique has more significant information entropy than compared techniques. Besides, it is observed that the grey value of the cipher image is uniformly distributed. As a result, the proposed technique can resist different malicious attacks.

5.2.5 Correlation between Two Adjacent Pixels

In this study, 2000 pairs of adjacent pixels have been chosen randomly from both plain and cipher images to analyze the correlation between them. The correlation coefficients are calculated as [35]

$$D(x) = \frac{1}{M}\sum_{j-1}^{M} x_j \tag{5.2}$$

$$F(x) = \frac{1}{M}\sum_{j-1}^{M}(x_j - D(x))^2 \tag{5.3}$$

$$Cov(x,y) = \frac{1}{M}\sum_{j-1}^{M}\Big(x_j - D(x)\Big)\Big(y_j - D(x)\Big) \tag{5.4}$$

$$r_{xy} = \frac{Cov(x,y)}{\sqrt{F(x)}\,\sqrt{F(y)}} \tag{5.5}$$

The grey level values of any two adjacent pixels are represented by x and y.

Table 5.5 shows the horizontal, vertical, and diagonal correlation coefficients of plain and cipher images of Table 5.3 for the proposed technique. It is observed that the dependency between any two adjacent pixels of the cipher image is extensively smaller than that of the plain image. Here, in plain images correlation coefficients are very close to 1 while those are nearly 0 for cipher images. That's why, adjacent pixels are fairly uniformly distributed. Besides, considering the Lena image a comparison with other techniques is presented in the same table.

Table 5.5: Correlation co-efficient for cipher images in Table 5.3 and comparison with others.

Images of Table 5.3	**Horizontal**		**Vertical**		**Diagonal**	
	Original	***Cipher***	***Original***	***Cipher***	***Original***	***Cipher***
(a)	0.9875	-0.0002	0.9930	0.0009	0.9811	-0.0016
(b)	0.9961	-0.0011	0.9954	0.0012	0.9916	-0.0027
(c)	0.9973	-0.0008	0.9981	0.0004	0.9952	-0.0007
(d)	0.9915	-0.0007	0.9970	-0.0001	0.9896	0.0001
For Lena image a comparison with other techniques						
Proposed	0.9399	0.0029	0.9693	0.0005	0.9179	-0.0029
[11]	0.9399	0.0249	0.9693	0.0505	0.9179	0.0280
[26]	0.9399	-0.0028	0.9693	0.0171	0.9179	-0.0022
[28]	0.9399	0.0127	0.9693	0.0190	0.9179	0.0012
[29]	0.9399	0.0136	0.9693	0.0062	0.9179	0.0175
[30]	0.9399	0.2546	0.9693	-0.0573	0.9179	-0.0024
[31]	0.9399	0.0242	0.9693	0.0194	0.9179	0.0024

It shows a favorable diffusion performance of the proposed technique as it has a smaller correlation coefficient. The table shows that the correlation of adjacent pixels of plain images has been successfully eliminated. As a result, virtually there exists no correlation among neighboring pixels in the cipher image. Hence, statistical attacks can be resisted by the proposed technique.

5.2.6 MSE and PSNR Analysis

The ratio of mean square error (MSE) difference between plain and cipher images to the maximum MSE difference is called the peak signal to noise ratio (PSNR). To measure the quality of the cipher image, PSNR is calculated as [36]

$$MSE = \frac{1}{M \times N} \sum_{a=0}^{M-1} \sum_{b=0}^{N-1} [[F(a,b) - F_0(a,b)]^2] \tag{5.6}$$

$$PSNR = 20.\,log \frac{255^2}{\sqrt{MSE}} \tag{5.7}$$

Larger PSNR value implies a higher quality of cipher image. PSNR and MSE analysis of cipher images (*i.e.*, 4th column of Table 5.3) and a comparison are listed in Table 5.6.

Table 5.6: PSNR and MSE analysis for cipher images and comparison with other techniques.

<table>
<tr><th>Images of Table 5.3</th><th>MSE
(Plain-Encrypted)</th><th colspan="2">PSNR
(Plain-Decrypted)</th><th>PSNR
(Plain-Encrypted)</th></tr>
<tr><td>(a)</td><td>1.717e+04</td><td colspan="2">∞</td><td>5.7834</td></tr>
<tr><td>(b)</td><td>1.719e+04</td><td colspan="2">∞</td><td>5.7792</td></tr>
<tr><td>(c)</td><td>1.217e+04</td><td colspan="2">∞</td><td>7.2787</td></tr>
<tr><td>(d)</td><td>1.562e+04</td><td colspan="2">∞</td><td>6.1947</td></tr>
<tr><td>(e)</td><td>8.969e+03</td><td colspan="2">∞</td><td>8.6032</td></tr>
<tr><td colspan="5"><b>For Lena image (5% data loss) a comparison with other technique</b></td></tr>
<tr><td></td><td colspan="2"><b>MSE (Plain-Decrypted)</b></td><td colspan="2"><b>PSNR (Plain-Decrypted)</b></td></tr>
<tr><td>Proposed</td><td colspan="2">0.43437</td><td colspan="2">33.780</td></tr>
<tr><td>[51]</td><td colspan="2">21.172</td><td colspan="2">34.873</td></tr>
</table>

5.2.7 Maximum Deviation

Another parameter to check the statistical security of the image encryption is the maximum deviation. It measures the divergence between pixel values of an original image and its corresponding cipher image [44]. A higher cost of the maximum deviation ensures the deviation in the cipher image from its plain image. The value is calculated as

$$D = \frac{d_0 + d_{255}}{2} + \sum_{1}^{254} d_i \tag{5.8}$$

where d_i is the difference between the histogram of the original image and that of cipher image at value i, and d_0 and d_{255} denote different values at index 0 and 255. For cipher images shown in Fig. 5, Table 5.7 presents the results of maximum deviation.

Table 5.7: Results of maximum deviation analysis for images of Table 5.3

Images of Table 5.3	**Maximum Deviation**
(a)	1043980
(b)	1056597
(c)	1078126
(d)	1638538
(e)	37976

5.2.8 Irregular Deviation

The maximum deviation alone itself is not enough to ensure the statistical randomness of the cipher image. The encryption technique should randomly change the pixel values to become a statistically robust scheme [45]. A method that makes a substantial change in some image pixel values and produces an insignificant change in others is not statistically secure. The procedure to calculate the amount of irregular deviation (I_D) is to take the histogram, say h of absolute difference of the plain image, and the cipher image. Now, to calculate the mean value of h, which is denoted as M_h, the I_D is estimated as

$$I_D = \sum_{i=0}^{255} |h_i - M_h| \tag{5.9}$$

Here, a smaller amount of I_D indicates that the histogram is closer to the uniformity and better the statistical properties of encryption. For cipher images shown in Table 5.3, Table 5.8 shows the results of I_D.

Table 5.8: Results of irregular deviation analysis for images of Table 5.3

Images of Table 5.3	**Irregular Deviation**
(a)	1254090
(b)	1029584
(c)	431683

Images of Table 5.3	Irregular Deviation
(d)	1939840
(e)	20168

5.2.9 Energy Analysis

It measures the energy of the cipher image, which estimates the sum of squared elements in the gray level co-occurrence matrix. It is calculated as [46]

$$Energy = \sum_{i,j} P(i,j)^2 \tag{5.10}$$

where $P(i, j)$ is the number of gray-level co-occurrence matrices. Here, the cost of the energy of a cipher image smaller than its corresponding plain image implies the efficiency of the encryption. For cipher images shown in Table 5.3, Table 5.9 shows the results of the energy analysis.

Table 5.9: Results of energy analysis for images of Table 5.3

Images of Table 5.3	Energy	
	Plain image	Cipher image
(a)	4.625705794060000e+11	1.714731669000000e+10
(b)	5.756694208300000e+11	1.714690980400000e+10
(c)	2.529138897840000e+11	1.714734726200000e+10
(d)	1.539354601014000e+12	8.283035050400000e+10
(e)	399812478	66644740

5.2.10 Contrast Analysis

In general, the contrast analysis of an image enables its viewer to recognize objects in the texture of that image vividly. Any cipher image gets higher contrast level due to high level of randomness at the encryption process. We have measured the contrast parameters of the cipher image and evaluate the effectiveness of the proposed technique. Contrast analysis yields a measure of the intensity contrast between a pixel and its neighbor over the whole image. The mathematical representation is described as [46]

$$C = \sum_{i,j} |i - j|^2 P(i, j) \tag{5.11}$$

where $P(i, j)$ is the number of gray-level co-occurrence matrices. For cipher images shown in Table 5.3, Table 5.10 shows the results of the contrast analysis

Table 5.10: Results of contrast analysis for images of Table 5.3

Images of Table 5.3	**Contrast**	
	Plain image	**Cipher image**
(a)	98730	11000853
(b)	50296	11013312
(c)	56830	11005228
(d)	346848	24207349
(e)	29449	683039

5.2.11 Efficiency Analysis

In real-time applications, efficiency is an important factor for secrecy. Table 5.11 represents the average encryption time for images of shown in Table 5.3. The proposed encryption technique is mainly based of permutation, diffusion, also iteration operation of the chaotic system costs much time.

Table 5.11: Encryption and Decryption time requirement by the proposed technique

Images of Table 5.3	**Key generation**	**Encryption**	**Decryption**	**Total**
	(time in sec.)			
(a)	1.826225	1.879723	1.865896	5.571843
(b)	1.822040	1.872924	1.837722	5.532688
(c)	1.832462	1.867613	1.854774	5.554849
(d)	1.831362	1.877685	1.854889	5.563936

Table 5.12: Time comparison with other techniques for Lena image (256 × 256)

Techniques	Encryption time (sec.)
Proposed	0.2525
[35]	1.1737
[47]	2.25
[48]	3.23

Table 5.12 shows the time comparison with other techniques for Lena (256 × 256) grayscale image. It is observed from the table that among the compared techniques the proposed one is faster than [35], [47] and [48].

5.3. Security Analyses

This section discussess the randomness of key and security of cipher image. For this reason, to validate the strength of the cipher image obtained by the proposed technique, the randomness test based-on the National Institute of Standards and Technology (NIST) test suite [58], and standard security and statistical tests have been performed.

5.3.1 Key Space

An enormous key space is necessary to guarantee a high level of security [41]. Herein, the keys are generated depending on 256-bit hash value along with $t_1',\ t_2',\ t_3', r, p, c$ parameters that are already introduced. If the precision is considered to be 10^{14} for the initial conditions of these parameters, the size of the key space becomes 10^{84}. Since, SHA-256 key space security is 2^{128}; thus, the total key space S is $10^{84} \times 2^{128} \approx 3.4\times 10^{122}$. This key space is adequate to prevent against exhaustive key attack. As a result, it is quite impossible to mount brute force attack on keys.

5.3.2 Key Sensitivity

An image cryptosystem should be sufficiently sensitive about slight changes in the secret key both at the encryption and the decryption stages [50]. Here, the MRI image shown in Table 5.3 (a) is used as the plain image. To show the key sensitivity of this proposed technique visually, first, one of the key parameters is changed while keeping other parameters as constant. Fig. 5.1

represents the key sensitivity in encryption stage and the NPCR for the corresponding cipher images of Fig. 5.1 is listed in Table 5.13.

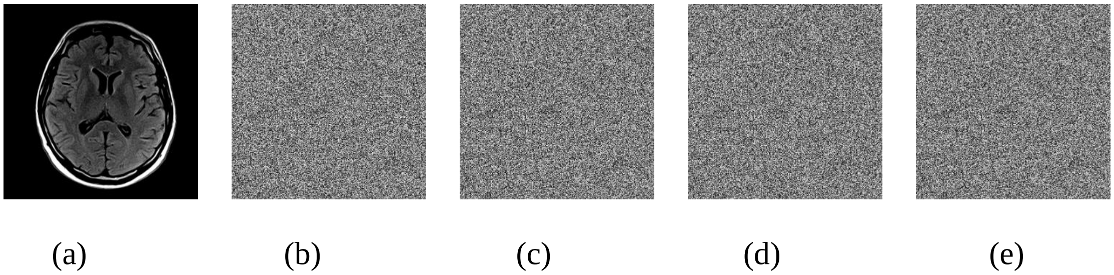


Figure 5.1: Key sensitivity analysis of the encryption stage: (a) the plain MRI image; (b) the cipher image obtained by using the original encryption key K; (c) - (e) the cipher image obtained by using 3 modified keys K1, K2 and K3.

Table 5.13: NPCR of corresponding cipher images for Fig. 5.1 (b) – (e) in encryption stage

Key	**Value**	**NPCR**
K	a5af04c49ebc0eaa4ed2c8cb01c4aa84390ea4933b39139c9a5e5b02015c2da5	0
*K*1	**d**5af04c49ebc0eaa4ed2c8cb01c4aa84390ea4933b39139c9a5e5b02015c2da5	99.61%
*K*2	a5af04c49ebc0eaa4ed2c8cb01c4aa84390**f**a4933b39139c9a5e5b02015c2da5	99.60%
*K*3	**7**5af04c49ebc0eaa4ed2c8cb01c4aa84390ea4933b39139c9a5e5b02015c2da5	99.61%

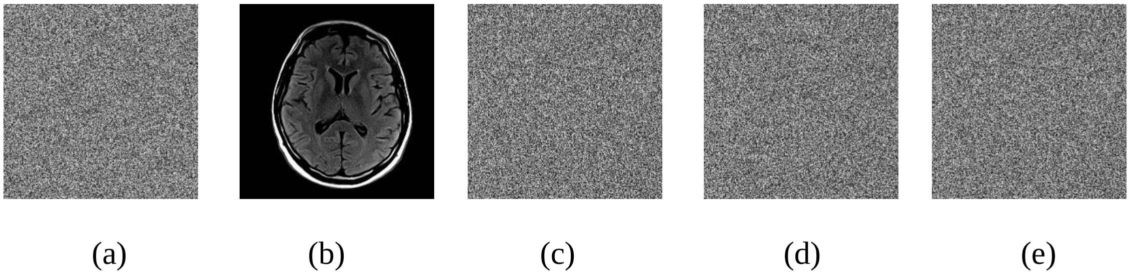


Figure 5.2: Key sensitivity analysis of the decryption stage: (a) the cipher image; (b) the retrieved image obtained by the original decryption key K; (c) - (e) the retrieved image obtained by using 3 modified keys K1, K2 and K3.

Fig. 5.2 shows the key sensitivity in the decryption stage and the NPCR for the corresponding decrypted images of Fig. 5.2 is presented in Table 5.14. In Table 5.13, the original SHA-256 hash key of the plain image is *K*, and *K*1, *K*2, *K*3 are three modified keys. Now, it is observed that when the plain image is encrypted by using three slightly modified keys, the corresponding cipher images are entirely different, and the NPCR values shown in Table 5.13 are more than 99.6%. Besides, while decrypting the cipher image with three slightly changed keys, more than 99% pixels are different from the retrieved images with the plain image presented in Table 5.14.

Table 5.14: NPCR of corresponding retrieved images for Fig. 5.2 (b) – (e) in decryption stage

Key	**Value**	**NPCR**
K	a5af04c49ebc0eaa4ed2c8cb01c4aa84390ea4933b39139c9a5e5b02015c2da5	0
K1	**d**5af04c49ebc0eaa4ed2c8cb01c4aa84390ea4933b39139c9a5e5b02015c2da5	99.59%
K2	a5af04c49ebc0eaa4ed2c8cb01c4aa84390**f**a4933b39139c9a5e5b02015c2da5	99.61%
K3	**7**5af04c49ebc0eaa4ed2c8cb01c4aa84390ea4933b39139c9a5e5b02015c2da5	99.60%

Moreover, an image encryption technique must be sensitive to any slight change of the plain image. Hence, a single pixel of the test image, *i.e*., Table 5.3 (a) is randomly changed to obtain a different hash key, and the NPCR of the corresponding cipher images are represented in Table 5.15. It shows that the modification of any single pixel produces a completely different hash value. When compared with the original cipher image shown in Table 5.3 (a), more than 99% of the pixels are changed. In brief, the experimental results show that the proposed technique is very sensitive concerning the secret key and the plain image. When there are any small changes, it influences the encryption and the decryption results.

Table 5.15: Sensitivity test results of the plain image in Table 5.3 (a)

Changed Pixel	**256-bit hash key**	**NPCR**
Original: P(614,453) = 77 Changed: P(614,453) = 78	K4=b57bf1f61f9a54cf4093a13c608b2eebc65f7da a873e7f951375652178825b6a	99.61%
Original: P(739,658) = 110 Changed: P(739,658) = 111	K5=a0b9eefbdf40a66daae16b50371064eb17a8b4 2cad9bc34e615e004747ee3e49	99.59%
Original: P(469,881) = 255 Changed: P(739,658) = 254	K6=e64b2618596ff481c61a7ad6a61903b363bef4 751ed240e5d502969e90eb9922	99.62%

5.3.3 Known-plaintext, Chosen-plaintext and Ciphertext-only attacks

Many image encryption methods get affected by known-plaintext, chosen-plaintext, ciphertext-only [39] etc. attacks. This thesis considers the following three points so that effectively it can resist these attacks. At first, the initial values of 1D Logistic map system is calculated by the SHA-256 value of the plain image and three external key (t_1', t_2' and t_3') parameters. Now, the shuffling is performed by the Logistic sequence, which is related with the plain image. Also, the construction of confusion key is mainly based on the plain image, and it performs the initial diffusion. If the plain image is modified, the usage of distinct SHA-256 values creates different initial value for the chaotic system. Thus, further confusion and diffusion result also become different. The final diffusion process performed by the Lorenz key is also generated from t_1', t_2' and t_3'. Even for the same plain image when these three external keys are changed, usually it produces distinct cipher image.

5.3.4 Occlusion Attack

Any ideal encryption technique should be robust against occlusion (data loss) attack [39] during transmission and storage.

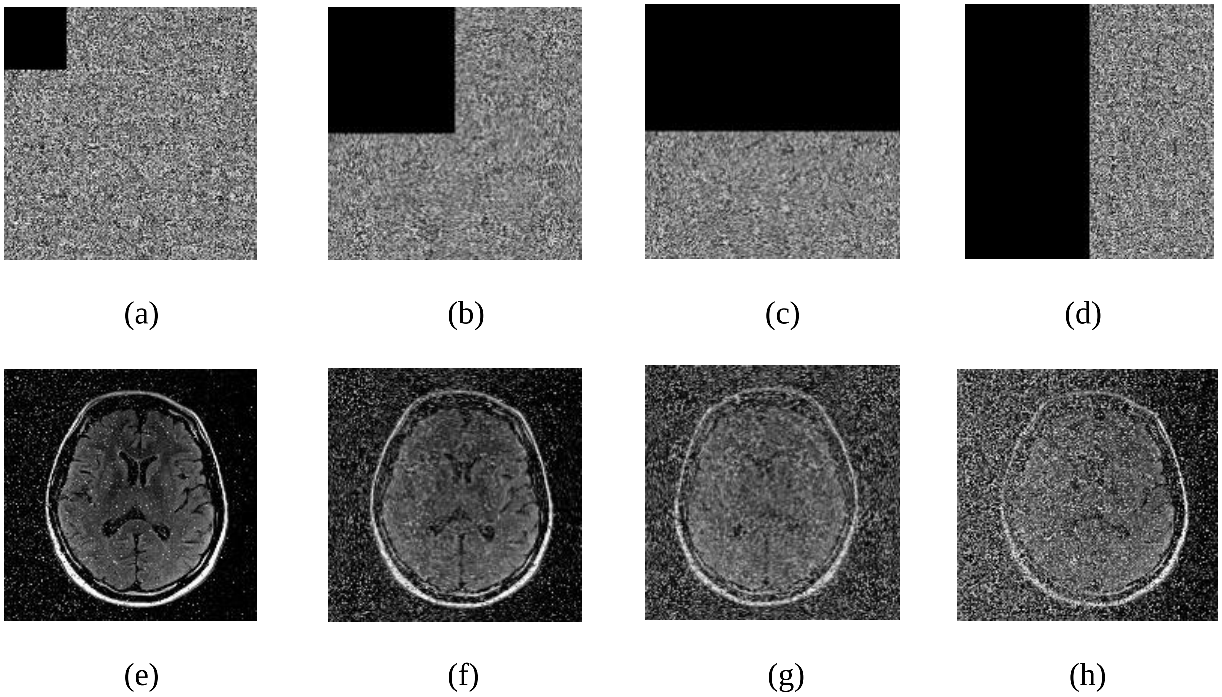


Figure 5.3: The encrypted medical image with (a) 256×256, (b) 512×512, (c) 512×1024, and (d) 1024×512 data cropping; the corresponding decrypted images are (e)-(h).

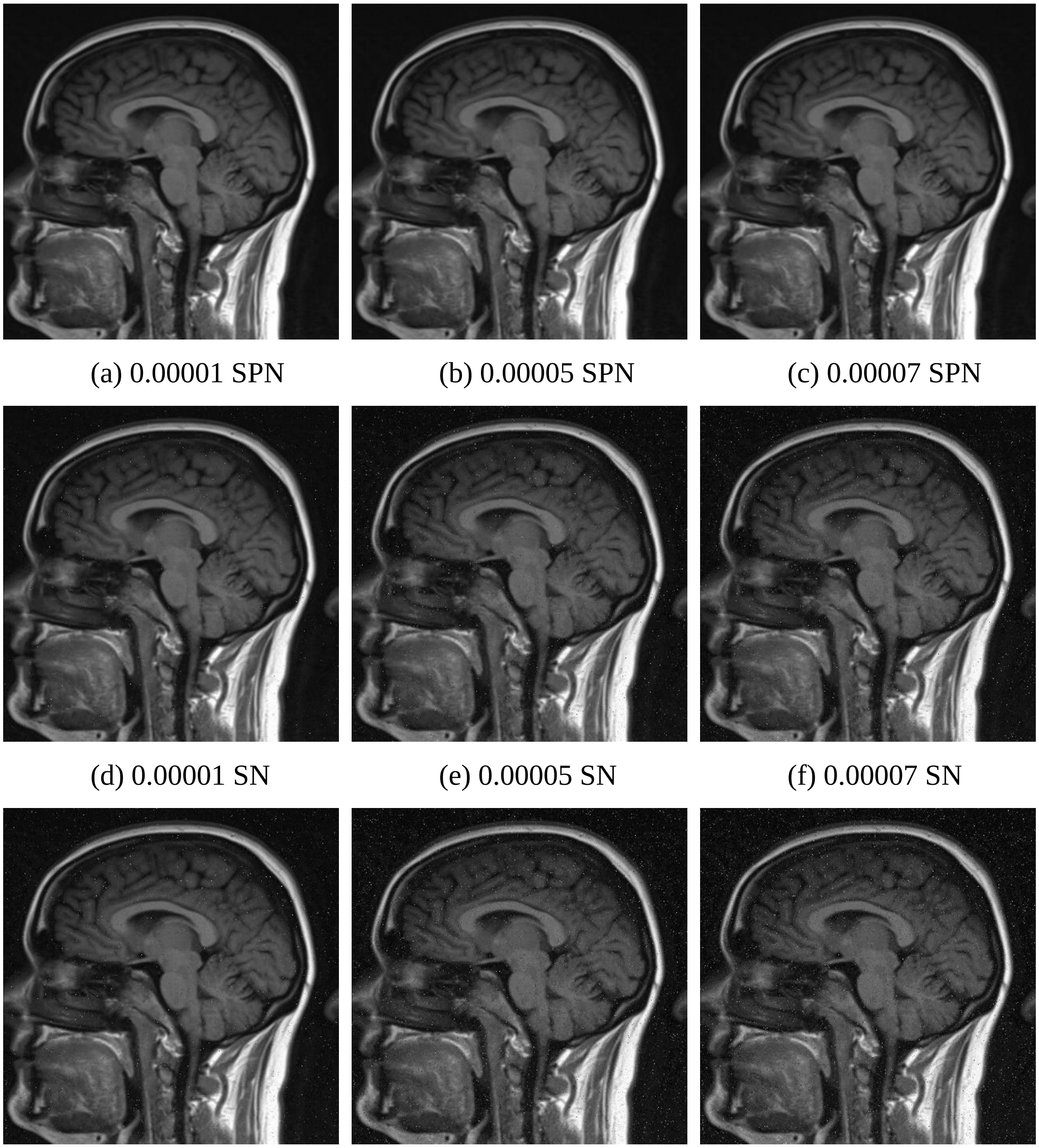
In order to justify the strength of the proposed technique against this attack, the cipher image has been cropped with the size of 256×256, 512×512, 512×1024, and 1024×512, as shown in Fig. 5.3 (a)-(d). Then the corresponding retrieved images are presented in Fig. 5.3 (e)-(h). From the figure, it is observed that still now they are recognizable. Therefore, the proposed technique sustains against this attack.

5.3.5 Noise Analysis

Three level of security are embedded with generating cipher text. While the cipher image transmits over the public network, it may be affected by any form of noise, namely, Salt & Pepper noise (SPN) [40], Gaussian noise (GN), and Speckle noise (SN), which makes the decryption process problematic [50]. If any proposed encryption technique can retrieve the cipher image with noise effectively, it ensures that it possesses the capability to resist noise attacks. The simulation considered the image of Table 5.3 (c) as the test plain image, and then noises contaminate the test image through variances 0.00001, 0.00005, 0.00007. Now, employing the decryption process, Fig. 5.4 presents the retrieved images.

Table 5.16: PSNR values of retrieved images under different kind of noise attacks on plain image shown in Table 5.3 (c).

Noise	Variances	PSNR (dB)
Salt & Pepper noise (SPN)	0.00001	59.556355453644684
	0.00005	51.975997825880633
	0.00007	49.035685629250679
Speckle noise (SN)	0.00001	31.076507615943502
	0.00005	26.972058536921974
	0.00007	26.061141714707905
Gaussian noise (GN)	0.00001	27.866778902939025
	0.00005	24.145174798695123
	0.00007	23.342118671173154



(a) 0.00001 SPN (b) 0.00005 SPN (c) 0.00007 SPN

(d) 0.00001 SN (e) 0.00005 SN (f) 0.00007 SN

(g) 0.00001 GN (h) 0.00005 GN (i) 0.00007 GN

Figure 5.4: For the developed technique, retrieved images under different noises for plain image of Table 5.3 (c).

The PSNR between the decrypted images and the plain image is computed and listed in Table 5.16. It is observed that (*i*) the developed technique has a robust resistance capability to SPN while the noise variances are from 0.00001 to 0.00007, the PSNR values are more extensive than 49 dB, and the retrieved images have a high level of visual appearance; (*ii*) besides, the GN has a sound effect on the extracted images, the PSNR values are around 23 dB, and visually they are recognizable; and (*iii*) the developed technique has better resistance capability to SN, PSNR values, vary from 26.061 dB to 31.076 dB, and the recovered images are recognizable, also. Thus, the developed technique possesses the capability to resist noise attacks.

5.3.6 Differential Attack

To test the capability of resistance against differential attack, number of pixels change rate ($NPCR$), and unified average changing intensity ($UACI$) is calculated as [41]

$$E(a,b) = \begin{cases} 0, P(a,b) = C(a,b) \\ 1, P(a,b) \neq C(a,b) \end{cases} \tag{5.12}$$

$$NPCR = \frac{\sum_{a=0}^{M-1} \sum_{b=0}^{N-1} E(a,b)}{M \times N} \times 100\% \tag{5.13}$$

$$UACI = \frac{1}{M \times N} \left[\sum_{a=0}^{M-1} \sum_{b=0}^{N-1} \frac{|P(a,b) - C(a,b)|}{255} \right] \times 100\% \tag{5.14}$$

where, P and C stand for plain and cipher images, respectively and their dimension is $M \times N$. For the plain images shown in Table 5.3, outcomes of $NPCR$ and $UACI$ for the proposed technique are presented in Table 5.17. It shows that $NPCR$ and $UACI$ are very near to the desired values. Therefore, the proposed technique has high resistance against this attack.

The standard expected values of *NPCR* and *UACI*, as shown in [33] for a gray image are 99.6094% and 33.4635%, respectively. For the proposed technique, the values of *NPCR* and *UACI* for the Lena image are 99.61% and 33.43%, respectively. It confirms a distinct variation of cipher image, even if there is any small change in the plain image. That’s why the proposed technique possesses keen sensitivity of the plain image. As a result, it proves better robustness than compared techniques listed in Table 5.17.

Table 5.17: NPCR and UACI Analysis for cipher images and comparison with different techniques.

Images of Table 5.3	NPCR (%)	UACI (%)
(a)	99.61	43.04
(b)	99.61	43.06
(c)	99.62	35.37
(d)	99.61	40.66

Images of Table 5.3	NPCR (%)	UACI (%)
For Lena image a comparison with other techniques		
Proposed	99.61	33.43
[11]	99.59	33.34
[29]	99.59	33.42
[32]	88.99	30.21
[33]	99.60	33.46

5.3.7 NIST Randomness Test

In order to verify the randomness of the generated cipher data, a statistical test suite has been designed by NIST, which consists of 15 different tests. To show the performance of each test, it estimates $P - value$. If this value is > 0.01 for the tests, the proposed technique is considered to pass (denoted as '√') the randomness test. For the proposed technique for the cipher images of Table 5.3, the results according to these tests have been presented in Table 5.18.

Table 5.18: NIST Randomness test for cipher images (*i.e. 4th* column) of Table 5.3

Test	P-value					Pass
	cipher of Fig.5 (a)	**cipher of Fig.5 (b)**	**cipher of Fig.5 (c)**	**cipher of Fig.5 (d)**	**cipher of Fig.5 (e)**	
The Frequency (Monobit) Test	0.739918	0.911413	0.534146	0.017912	0.739918	√
Frequency Test within a Block	0.350485	0.739918	0.122325	0.911413	0.350485	√
The Runs test	0.350485	0.122325	0.911413	0.534146	0.739918	√
Longest-Run-of-Ones in a Block	0.739918	0.739918	0.122325	0.911413	0.017912	√
The Binary Matrix Rank Test	0.739918	0.213309	0.739918	0.066882	0.350485	√
Discrete Fourier Transform Test	0.911413	0.066882	0.911413	0.534146	0.534146	√
Non-overlapping Template Matching	0.534146	0.350485	0.213309	0.739918	0.911413	√
Overlapping Template Matching Test	0.911413	0.350485	0.739918	0.534146	0.739918	√
The Linear Complexity Test	0.066882	0.534146	0.350485	0.739918	0.213309	√
The Serial test						
$P - value$ 1	0.739918	0.739918	0.739918	0.911413	0.739918	√
$P - value$ 2	0.122325	0.213309	0.213309	0.991468	0.213309	√
The Cumulative Sums Test	0.739918	0.739918	0.350485	0.911413	0.911413	√

CHAPTER VI

Conclusions

6.1 Concluding Discussion

By combining SHA-256, Logistic map, Lorenz attractor and dynamic DNA encoding computing rules; the proposed multi-stage encryption technique enriches the security-level of the medical image. Here initially, a chaotic sequence is generated by exploiting Logistic map and SHA-256 value altogether, for the plain image. Then according to this sequence, the plain image is converted into a confusing image. Again using the same sequence, a confusion key is generated to encrypt this blur image. Finally, the deployment of Lorenz attractor makes another encryption key. Then dynamic DNA encoding and computing performed between this key and encrypted blur image which rules are determined by the logistic sequence. Thus, the final cipher image is produced. As a result the ultimate cipher is extensively confusing, and for the intruder, it is not possible to mount any form of attack. The randomness, security and statistical test analyses considered here also guarantee the robustness of the proposed technique.

6.2 Future work

The proposed technique is a specially designed for digital images especially for medical image data. This technique can be applied in any cryptographic application in the cloud storage and data distribution platform. Further modification of the technique will apply this in real world security protocol. A further plan of extension is to incorporate following techniques to maintain faster transmission and lossless properties:

- Compression
- De-noising

REFERENCES

1. S. Jahan, M. Chowdhury, R. Islam, and J. Gao, “Security and privacy protection for ehealth data,” in Int. Conf. on Future Network Systems and Security, pp. 197-205. Springer, Cham, Jul. 2018.

2. M. Aiello, C. Cavaliere, A. D’Albore, and M. Salvatore, “The challenges of diagnostic imaging in the era of big data,” J. of clinical medicine, Vol. 8, No. 3, p. 316, Mar. 2019

3. M. A. Murillo-Escobar, L. Cardoza-Avendaño, R. M. López-Gutiérrez, and C. Cruz-Hernández, “A double chaotic layer encryption algorithm for clinical signals in telemedicine,” J. of medical systems, Vol. 41, No. 4, Apr. 2017.

4. J. W. Lian, D. C. Yen, and Y. T. Wang, “An exploratory study to understand the critical factors affecting the decision to adopt cloud computing in Taiwan hospital,” In. J. of Information Management, Vol. 34, No. 1, pp. 28-36, Feb. 2014.

5. Y. Zhang, M. Qiu, C. W. Tsai, M. M. Hassan, and A. Alamri, “Health-CPS: Healthcare cyber-physical system assisted by cloud and big data,” IEEE Systems Journal, Vol. 11, No. 1, pp. 88-95, Aug. 2015.

6. F. Cao, H. K. Huang, and X. Q. Zhou, “Medical image security in a HIPAA mandated PACS environment,” Computerized medical imaging and graphics, Vol. 27, No. 2-3, pp. 185-196, 2003.

7. P. Ruotsalainen, “Privacy and security in teleradiology,” European J. of radiology, Vol. 73, No. 1, pp. 31-35, Jan. 2010.

8. M. T. I. Siyam, K. M. R. Alam, and T. Jami, “An exploitation of visual cryptography to ensure enhanced security in several applications,” Int. Journal of Computer Applications, Vol. 65, No. 6 pp. 42-46, 2013.

9. A. Kannammal, and S. S. Rani, “DICOM image authentication and encryption based on RSA and AES algorithms,” in Int. Conf. on Intelligent Robotics, Automation, and Manufacturing, pp. 349-360. Springer, Berlin, Heidelberg, Nov. 2012.

10. J. B. Lima, F. Madeiro, and F. J. Sales, “Encryption of medical images based on the cosine number transform,” Signal Processing: Image Communication, Vol. 35, pp. 1-8, Jul. 2015.

11. S. M. Pan, R. H. Wen, Z. H. Zhou, and N. R. Zhou, “Optical multi-image encryption scheme based on discrete cosine transform and nonlinear fractional Mellin transform,” Multimedia Tools and Appl, Vol. 76, No. 2, pp. 2933-2953, 2017.

12. A. Begum, A. Siddiqua, and S. Nirmala, “Secure visual cryptography for medical image using modified cuckoo search,” Multimedia Tools and Appl, Vol. 77, No. 20, pp. 27041-27060, 2018.

13. M. Abdel-Basset, A. E. Fakhry, I. El-Henawy, T. Qiu, and A. K. Sangaiah, “Feature and intensity based medical image registration using particle swarm optimization,” J. of medical systems, Vol. 41, No. 12, pp.197, 2017

14. N. K. Pareek, and V. Patidar, “Medical image protection using genetic algorithm operations,” Soft Computing, Vol. 20, No. 2, pp. 763-772, Feb. 2016.

15. C. Fu, W. H. Meng, Y. F. Zhan, Z. L. Zhu, F. C. Lau, K. T. Chi, and H. F. Ma, “An efficient and secure medical image protection scheme based on chaotic maps,” Computers in biology and medicine, Vol. 43, No. 8, pp. 1000-1010, Sep. 2013.

16. D. Ravichandran, P. Praveenkumar, J. B. Rayappan, and R. Amirtharajan, “Chaos based crossover and mutation for securing DICOM image,” Computers in biology and medicine, Vol. 72, pp. 170-184, May. 2016.

17. M. Y. M. Parvees, J. A. Samath, and B. P. Bose, “Secured medical images-a chaotic pixel scrambling approach,” J. of medical systems, Vol. 40, No. 11, pp. 232, Nov. 2016.

18. H. Wang, J. M. Ye, H. F. Liang, and Z. H. Miao, “A medical image encryption algorithm based on synchronization of time-delay chaotic system,” Advances in Manufacturing, Vol. 5, No. 2, pp. 158-164, 2017.

19. R. M. May, “Single mathematical dynamics with very complicated dynamics,” Nature, Vol. 276, pp. 458-467, 1976.

20. E. N. Lorenz, “Deterministic nonperiodic flow,” J. of the atmospheric sciences, Vol. 20, No. 2, pp. 130-141, Mar. 1963.

21. A. N. Pisarchik, N. J. F. Carmona, and M. C. Valadez, “Encryption and decryption of images with chaotic map lattices,” Chaos: An Interdisciplinary J. of Nonlinear Science, Vol. 16, No. 3, 2006.

22. Brain MRI Images, http://cn.anke.com/ProductsView.as p?CasesID=3, last accessed 2019/11/11.

23. Full-EMD, https://edm.bioscientifica.com/view/journals /edm/2015/1/EDM15-0079.xml, last accessed 2019/11/15.

24. MRI Brain Scan. https://www.rock-cafe.info/suggest/br ain-stem-anatomy-ct-scan-627261696e.html, last accesssd 2019/11/11.

25. Brain MRI Axial, https://www.rock-cafe.info/suggest/brain-stem-anatomy-ct-scan-627261696e.html, last accessed 2019/11/13.

26. X. Chen, and C. J. Hu, “Adaptive medical image encryption algorithm based on multiple chaotic mapping,” Saudi J. of biological sciences, Vol. 24, No. 8, pp. 1821-1827, 2017.

27. S. J. Deng, G. C. Huang, and Z. J. Chen, “Research and implement of Self adaptive image encryption algorithm based on chaos,” J. Comput., Vol. 31, No. 6, pp. 1502-1504, 2011.

28. X. Liao, S. Lai, and Q. Zhou, “A novel image encryption algorithm based on self-adaptive wave transmission,” Signal Processing, Vol. 90, No. 9, pp. 2714-2722, 2010.

29. G. B. Xie, and Y. M. Ding, “Image encryption algorithms with variable confusion parameters based on logistic mapping,” Microelectron. Comput, Vol. 32, No. 4, pp. 111-115, 2015.

30. A. Kanso, and M. Ghebleh, “A novel image encryption algorithm based on a 3D chaotic map,” Communications in Nonlinear Science and Numerical Simulation, Vol. 17, No. 7, pp. 2943-2959, 2012.

31. L. Teng, and X. Wang, “A bit-level image encryption algorithm based on spatiotemporal chaotic system and self-adaptive,” Optics Communications, Vol. 285, No. 20, pp. 4048-4054, 2012.

32. J. Zhang, D. Hou, and H. Ren, “Image encryption algorithm based on dynamic DNA coding and Chen’s hyperchaotic System,” Mathematical Problems in Engineering, 2016.

33. X. H. Deng, C. L. Liao, C. X. Zhu, and Z. G. Chen., “Image encryption algorithms based on chaos through dual scrambling of pixel position and bit,” J. Commun, Vol. 3, p. 025, 2014.

34. A. Mondal, K. M. R. Alam, G. G. M. N. Ali, P. H. J. Chong, and Y. Morimoto, “A Multi-Stage Encryption Technique to Enhance the Secrecy of Image,” KSII TIIS, Vol. 13, No. 5, pp. 2698-2717, 2019.

35. X. Chai, J. Zhang, Z. Gan, and Y. Zhang, “Medical image encryption algorithm based on Latin square and memristive chaotic system,” Springer Multimedia Tools and Applications, Vol. 78, No. 24, pp. 35419-35453, 2019.

36. Z. Hua, S. Yi, and Y. Zhou, ”Medical image encryption using high-speed scrambling and pixel adaptive diffusion,” Elsvier Signal Processing, 144, pp. 134-144, 2018.

37. C. Li, Y. Zhang, and E. Y. Xie, “When an attacker meets a cipher-image in 2018: A year in review,” Elsevier J. of Information Security and Applications (JISA), 48, p. 102361, 2019.

38. Y. Luo, J. Yu, W. Lai, and L. Liu, “A novel chaotic image encryption algorithm based on improved baker map and logistic map,” Multimedia Tools and Applications, 78 (15), pp. 22023-22043, 2019.

39. M. R. Biswas, K. M. R. Alam, S. Tamura, and Y. Morimoto, “A technique for DNA cryptography based on dynamic mechanisms”, J. of Information Security and Applications (JISA), Elsevier, Vol. 2019, No. 48, p 102363, October, 2019.

40. M. S. R. Tanveer, K. M. R. Alam, M. A. M. Akash, and Y. Morimoto, “A technique to reconstruct wavelet-based watermark immune against salt & pepper noise,” in 4th Int. Conf. on Electrical Information and Communication Technology (EICT), pp. 1-5, IEEE, December, 2019.

41. X. Chai, X. Fu, Z. Gan, Y. Lu, and Y. Chen, “A color image cryptosystem based on dynamic DNA encryption and chaos,” Signal Processing Vol.155, pp. 44-62, 2019.

42. Li, Hao, Lianbing Deng, and Zhaoquan Gu. “An image encryption scheme based on precision limited chaotic system,” Multimedia Tools and Applications, pp. 1-24, 2020.

43. L. Zhang, Y. Liu, F. Pareschi, Y. Zhang, K. Wong, R. Rovatti, and G. Setti, “On the security of a class of diffusion mechanisms for image encryption,” IEEE Trans. on cybernetics Vol. 48, No. 4, pp. 1163-1175, 2017.

44. M. Usama, M. K. Khan, K. Alghathbar, and C. Lee, “Chaos-based secure satellite imagery cryptosystem,” Computers & Mathematics with Applications, Vol. 60, No. 2, pp. 326-337, 2010.

45. B. Mondal, P. K. Behera, and S. Gangopadhyay, “A secure image encryption scheme based on a novel 2D sine–cosine cross-chaotic (SC3) map,” J. of Real-Time Image Processing, pp. 1-18, 2020.

46. T. Shah, I. Hussain, M. A. Gondal, and H. Mahmood, “Statistical analysis of S-box in image encryption applications based on majority logic criterion,” Int. J. Phys. Sci, Vol. 6, No. 16, pp. 4110-4127, 2011.

47. J. Ahmad, and S. O. Hwang, “A secure image encryption scheme based on chaotic maps and affine transformation,” Multimedia Tools and Appl, Vol. 75, No. 21, pp. 13951-13976, 2016.

48. F. Ahmed, A. Amir, V. U. Abbas, and M. Y. Siyal, “A noisy channel tolerant image encryption scheme,” Wireless personal communications, Vol. 77, No. 4, pp. 2771-2791, 2014.

49. C. Li, D. Lin, and J. Lü, “Cryptanalyzing an image-scrambling encryption algorithm of pixel bits,” IEEE MultiMedia, Vol. 24, No. 3, pp. 64-71, 2017.

50. Z. H. Gan, X. L. Chai, D. J. Han, and Y. R. Chen, “A chaotic image encryption algorithm based on 3-D bit-plane permutation,” Neural Computing and Applications, Vol. 31, No. 11, pp.7111-7130, 2019.

51. Z. Hua, B. Xu, F. Jin, and H. Huang, “Image encryption using josephus problem and filtering diffusion,” IEEE Access, 7, pp.8660-8674, 2019.

52. X. Lian, J. Li, M. Huang, and Y. Duan, “A Robust Watermarking Algorithm for the Encryption of Medical Big Data,” Int. J. of Simulation Systems, Science & Technology, Vol.17, No. 46, 2016.

53. Huang, C. K., and Hsiau-Hsian Nien. "Multi chaotic systems based pixel shuffle for image encryption." *Optics communications* 282, no. 11, pp. 2123-2127, 2009.

54. Lian, Shiguo, Jinsheng Sun, and Zhiquan Wang. "A block cipher based on a suitable use of the chaotic standard map." *Chaos, Solitons & Fractals* 26, no. 1, pp.117-129, 2005.

55. Zhang, Qiang, Ling Guo, and Xiaopeng Wei. "Image encryption using DNA addition combining with chaotic maps." *Mathematical and Computer Modelling* 52, no. 11-12, pp.2028-2035, 2010.

56. Fridrich, Jiri. "Symmetric ciphers based on two-dimensional chaotic maps." *International Journal of Bifurcation and chaos* 8, no. 06, pp.1259-1284, 1998.

57. Adleman, Leonard M. "Molecular computation of solutions to combinatorial problems." *Science* 266, no. 5187, pp.1021-1024, 1994.

58. NIST, https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-22r1a.pdf