CAPITAL UNIVERSITY OF SCIENCE AND
TECHNOLOGY, ISLAMABAD

A Gray-scale Image Encryption Algorithm Based on S-box and Logistic Map

by

Abdul Qadeer

A thesis submitted in partial fulfillment for the
degree of Master of Philosophy

in the

Faculty of Computing
Department of Mathematics

2022

I would like to dedicate my thesis to my beloved parents, and respected teachers especially my elder brother and my brother-in-law who encouraged and helped me in every field of my life.

CERTIFICATE OF APPROVAL

A Gray-scale Image Encryption Algorithm Based on S-box and Logistic Map

by

Abdul Qadeer

(MMT191002)

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner	Dr. Ayesha Rafiq	IST, Islamabad
(b)	Internal Examiner	Dr. M. Sabeel Khan	CUST, Islamabad
(c)	Supervisor	Dr. Rashid Ali	CUST, Islamabad

Dr. Rashid Ali
Thesis Supervisor
November, 2022

Dr. Muhammad Sagheer
Head
Dept. of Mathematics
November, 2022

Dr. M. Abdul Qadir
Dean
Faculty of Computing
November, 2022

Author's Declaration

I, **Abdul Qadeer** hereby state that my MS thesis titled "**A Gray-scale Image Encryption Algorithm Based on S-box and Logistic Map**" is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/abroad.

At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my MS Degree.

(Abdul Qadeer)
Registration No: MMT191002

Plagiarism Undertaking

I solemnly declare that research work presented in this thesis titled "**A Gray-scale Image Encryption Algorithm Based on S-box and Logistic Map**" is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been duly acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS Degree, the University reserves the right to withdraw/revoke my MS degree and that HEC and the University have the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized work.

(Abdul Qadeer)
Registration No: MMT191002

Acknowledgement

I am thankful to Almighty Allah, the Most Gracious and Merciful, Who?s confer me the courage and strength to complete this thesis. I could not have done anything without the grace of Allah Almighty. I thank my parents my sibling for encouraging me and helping me to complete my MPhil and financially supporting me. However, my deepest gratitude is reserved for my Parents for their earnest prayers, unconditional love and unflinching support in completing my degree program. Most of all I am thankful to my brother-in-law **Muhammad Nasir Abbas** who has encouraged and helped me in every phase of life.

I also very thankful to my supervisor **Dr. Rashid Ali** for his kindness and support. This thesis would not have been possible without his guidance. He was always there whenever I had a problem. I sincerely appreciate all of their time and effort and proud to be a student of such kind supervisor. He is also a professor as well as a very nice person.

I am also thankful to my friends for motivating me during my degree program, especially **Nazish Khan, Naeem Muzaffar, Muhammad Faisal Khan, Usama Riaz, Usama Malik and Tahir Gujjar**. Mostly, I would like to thank **Tahir Ali Sajad** also helped me a lot and guided me whenever I needed it.

(Abdul Qadeer)

Abstract

In this thesis, first a cubic fractional transformation (CFT) is used for the construction of S-box. The S-boxes are used to increase the confusion between the ciphertext and the key. The S-boxes are assessed using standard tests suit which includes nonlinearity, strict avalanche criterion, bit independence criterion, linear approximation probability and differential probability. The study of image encryption has grown in popularity and interest in the modern technological era. Image encryption ensures the secure transmission of images by converting recognizable images into unrecognizable ones. A new image encryption method based Substitution-box (S-box) and Logistic map is proposed. This S-box is used for the pixel values modification to generate element of non-linearity. After this, these modified values are further diffused with two other random sequences, generated by CFT. Finally the scrambling process with the help of `XOR` Boolean operation a random sequences generated by Logistic map are applied to the components of pre-encrypted image to gets the encrypted image. The use of Substitution-box (S-box) and Logistic map based image encryption scheme shows good results for key space analysis, key sensitivity, correlation analysis, number of pixel change rate (NPCR), unified average changing intensity (UACI), entropy analysis and histogram analysis. Security analysis demonstrate the good performance of the algorithm (1), (2) and (3) as a secure and effective communication method for images.

Contents

Author's Declaration iv

Plagiarism Undertaking v

Acknowledgement vi

Abstract vii

List of Figures x

List of Tables xi

Abbreviations xii

Symbols xiii

1 Introduction 1
- 1.1 S-Boxes in Cryptography . . . 2
  - 1.1.1 Evaluation for Good S-boxes . . . 3
- 1.2 Image Encryption . . . 4
- 1.3 Literature Review . . . 5
- 1.4 Software Tools For S-box Analysis . . . 7
- 1.5 Thesis Objective . . . 9

2 Preliminaries 10
- 2.1 Cryptology . . . 10
- 2.2 Mathematical Background . . . 14
- 2.3 Galois Field . . . 21
- 2.4 Boolean Function . . . 24
  - 2.4.1 Properties of Boolean Functions . . . 24
- 2.5 Cryptographic Properties of a Strong S-box . . . 29

3 S-box Construction Using Cubic Fractional Transformation (CFT) 35
- 3.1 Construction of S-box . . . 35
  - 3.1.1 Cubic Fractional Transformation (CFT) . . . 36

3.1.2 Inverse S-box . . . 38
3.2 Properties and Analysis of the Proposed S-box . . . 39

4 Logistic Map and S-box Based Image Encryption 44
4.1 Basic Terminologies . . . 45
4.1.1 Digital Image . . . 45
4.1.1.1 Pixel . . . 45
4.1.2 Component of Image Encryption Cryptosystem . . . 45
4.2 Chaotic Map . . . 46
4.2.1 Lyapunov Exponent . . . 47
4.2.2 Bifurcation Diagram . . . 47
4.3 Logistic Map . . . 47
4.4 Image Encryption Algorithm . . . 50
4.4.1 Encryption Algorithm (Gray-scale) . . . 50
4.5 Decryption Algorithm (Grayscale) . . . 53
4.6 Results and Discussions . . . 59
4.6.1 Security Analysis . . . 60
4.6.1.1 Key Space . . . 60
4.6.1.2 Key Sensitivity Analysis . . . 61
4.6.1.3 Differential Analysis . . . 61
4.6.1.4 Correlation Coefficients Analysis (CCA) . . . 62
4.6.1.5 Entropy Analysis . . . 63
4.6.1.6 Histogram Analysis . . . 64

5 Conclusion 68

Bibliography 69

List of Figures

1.1 Classification of image encryption algorithm 4

2.1 Symmetric-key encryption 12
2.2 Asymmetric-key encryption 13
2.3 Clasification of cryptology 14

3.1 Flow chart of proposed S-box 37

4.1 Lyapunove exponent of Logistic map 48
4.2 Bifurcation diagram for the Logistic map with iterations = 65536 . 49
4.3 Flow chart of image encryption algorithm 53
4.4 Flow chart of image decryption algorithm 55
4.5 Results of Clock Image encryption and decryption algorithm: (a) plainimage, (b) Encrypted Image, (c) Decrypted Image 59
4.6 Results of Chemical Plant image encryption and decryption algorithm: (a) plainimage, (b) Encrypted Image, (c) Decrypted Image 60
4.7 Key sensitivity test for Clock: (a) Plainimage (b) Encrypted image (c) Decrypted image by slightly changed key 61
4.8 The following is the encryption result for the grayscale image of Clock: (a) Plainimage (b) Encrypted image (c) Histogram of plainimage (d) Histogram of encrypted image 65
4.9 Histogram analysis of chemical plant plainimage and cipherimage . 65
4.10 Histogram analysis of girl plainimage and cipherimage 66
4.11 Histogram analysis of house plainimage and cipherimage 66

List of Tables

2.1 Addition table of integer in mod 8 . . . 17
2.2 Multiplication table of integers mod 8 . . . 17
2.3 Truth Table of `XOR`, `AND` functions . . . 25
2.4 The Hamming distance of two Boolean functions "f" and "g" . . . 27
2.5 Truth table of WHT_f . . . 28
2.6 S-box fixed point . . . 33
2.7 S-box opposite fixed point . . . 33

3.1 S-Box . . . 38
3.2 Inverse S-Box . . . 39
3.3 Comparing the non-linearity values of various S-boxes . . . 40
3.4 S-box and non-linearity . . . 41
3.5 BIC Value of S-box . . . 41
3.6 SAC Value of S-box . . . 42
3.7 DP Value of S-box . . . 43

4.1 UACI and NPCR values of encrypted images . . . 62
4.2 Two adjacent pixels' correlation coefficient in a plain and cipher image. . . . 63
4.3 Entropy analyses . . . 64

Abbreviations

AES	Advance Encryption Standard
AI	Algebraic Immunity
ANF	Algebraic Normal Form
BIC	Bits Independence Criterion
CI	Correlation Immunity
CC	Correlation Coefficients
CFT	Cubic Fractional Transformation
DES	Data Encryption Standard
DP	Differential Probability
GCD	Greatest Common Divisor
LFT	Linear Fractional Transformation
LE	Lyapunov Exponent
LP	Linear Probability
MATLAB	Matrix Laboratory
NPCR	Number of Pixel Change Rate
RC4	Rivest Cipher 4
RSA	Rivest-Shamir-Adleman
SET	S-box Evalution Tool
SAC	Strict Avalanche Criterion
UACI	Unified Average Changing Intensity
WHT	Walsh Hadamard Transformation

Symbols

C	Cipherimage
C	Ciphertext
γ	Control Parameter of Logistic Map
D	Decryption Algorithm
E	Encryption Algorithm
Λ	Lyapunov Exponent
n	Number of Rows
m	Number of Columns
P	Plaintext
I	Plainimage
$\mathbb{Z}$	Set of Integers
$\mathbb{R}$	Set of Real Number
$\mathbb{Z}_{>}$	Set of Integer under Modulo m
$\mathbb{Q}$	Set of Rational Number

Chapter 1

Introduction

How to secure personal secret information has been a problem for both states and individuals since the beginning of secure communication. Then, the think tanks of these states gather to develop a mechanism to protect the transfer of information of the relevant secret message from them to their loyalists. In the modern era, cryptography technology provides a solution for protecting secret information/messages from unauthorized resources. In cryptography, the original information is known as plaintext (P), while the codded information is known as ciphertext (C). The method of protecting information/messages during transmission so that only the intended person can read, change, and process it.

In cryptography, the methods for protecting information generally include an algorithm and a key. An encryption algorithm first converts the original information into some type of coded version of it, making it unintelligible to someone who is not supposed to receive it. This process ensures that communication is done securely. Once the information has been encrypted, it is able for transmission over a public network. To make the information readable, the receiver also does some tasks. He decrypts the information using the decryption key and algorithm. Caesar Cipher [1] is the simplest and most well-known example of such a technique. It works on the principle of substitution, where each alphabet is changed for another alphabet, or, to put it another way, each letter is moved a predetermined number of positions within the alphabet. Other algorithms, like mono-alphabetical, four

square cipher [2], playfair cipher [3] and hill cipher [4], etc have been frequently utilized.

Based on key management, cryptography can be divided into two categories: symmetric key cryptography and asymmetric key cryptography [5]. In symmetric key cryptography, a single key is used for both data encryption and decryption. To overcome the remarkable key problems in symmetric key cryptography, Diffie-Hellman in 1976 introduced [6] asymmetric key cryptography. Asymmetric key cryptography uses two different keys to perform encryption and decryption, so knowing one key does not mean necessarily knowing the other.

1.1 S-Boxes in Cryptography

In cryptography, a substitution box (S-Box) is an essential component of symmetric key algorithms. They are frequently utilized in block ciphers to conceal the connection between the key and the ciphertext, which is Shannon's property of confusion and diffusion [7]. The concept of confusion and diffusion was first introduced by Claude Shannon in 1949 [7]. According to Claude Shannin's theory:

Confusion refers to making the relationship between the key and the ciphertext as complex and as involved as possible.

Diffusion refers to the property that redundancy in the statistics of the plaintext is 'dissipated' in the statistics of the ciphertext.

In block cipher, S-box is an important component. S-boxes are frequently chosen with care to avoid cryptanalysis. An S-box is a non-linear component in symmetric block ciphers that uses look-up tables to give the confusion property in the cryptosystem. As a result, in cryptographic methods, constructing an S-box with strong cryptographic characteristics is crucial.

The substitution process converts the input bits to the output bits after a number of layers of substitutions, hence produce the ciphertext. To evaluate the strength and confusion-creating ability of an S-box, the bit change patterns at the output, that is based on a single or more bit changes at the input, must be examined.

S-boxes are look-up tables that represent vectorial Boolean functions. An S-box

takes a tiny block of bits and replaces it with another block of bits. To make decryption effective, this substitution should be one-to-one. S-box converts n input bits into m output bits. The use os S-box as look-up table will give 2^n words, each with m bits. As employed in the Data Encryption Standard (DES) [8], a 4-bit S-box is a box of $2^4 = 16$ components with hexadecimal values ranging from 0 to F that is randomly organised. Similarly, the number of elements in an 8-bit S-box is $2^8 = 256$, with values ranging from 0 to 255, as used in the Advance Encryption Standard (AES) [9]. The output length can either be the same as the input length, as in AES [9], or different, as in DES [8, 10]. To make a cryptosystem strong, to constructed S-box it must be ensured that each output bit dependents on each input bit.

1.1.1 Evaluation for Good S-boxes

The structural simplicity, rapid encryption and decryption speeds, and resistance against known as cryptanalysis techniques are all desired features of an S-box. Since 1976, many researchers have analyzed the behavior of constructed S-boxes. The features listed below are commonly recognised as important criteria for evaluating good S-boxes.

1. Bijection Property
2. Non-linearity
3. Strict Avalanche Criterion (SAC)
4. Differential Probability (DP)
5. Linear Probability (LP)
6. Bits Independence Criterion (BIC)

1.2 Image Encryption

In the modern era, with the rapid development of information technology, necessitate the methods for the protection of data specially that containing images. The protection of sensitive image based data from unauthorized source is know as image encryption. A digital image is altered in a way that it become completely different from the original image and cannot be viewed directly. When the image is to be viewed, then it should be converted back into the original form through the decryption algorithm. Many researcher have proposed image encryption methods to improve the protection of digital images from unauthorized access. Ullah et al. [11] used a novel scheme to encrypt images using S-box and chaotic system. Fridrich [12] suggested a chaotic map based image encryption scheme.

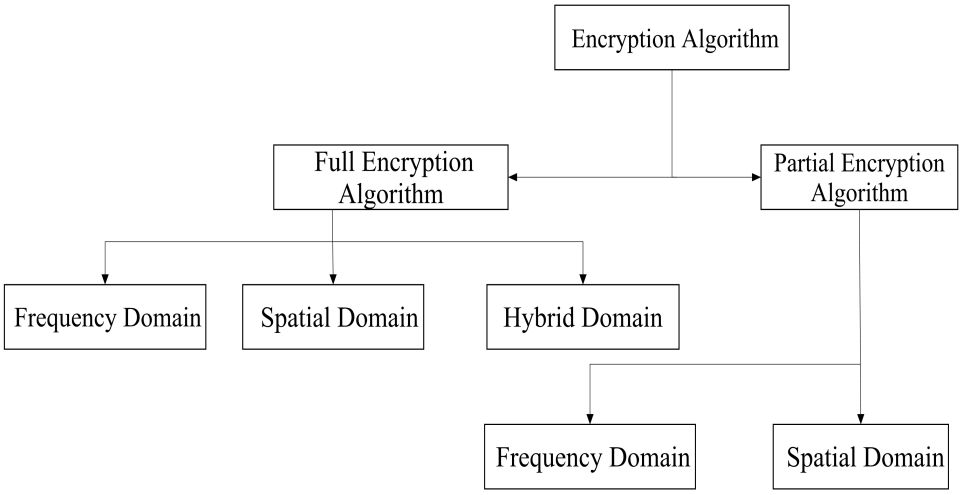

FIGURE 1.1: Classification of image encryption algorithm

Image encryption can be classified into two parts based on its domain. One is called full encryption algorithm and the other is called partial encryption algorithm [13]. As the name suggests, full image encryption algorithm encrypts the full image and partial encryption algorithm encrypts only a part of the image instead of the full image. Both of these types of algorithms can be further classified into frequency domain and spatial domain. The frequency domain encryption algorithms are

based on the transform function, such as fractional Fourier transform, quantum Fourier transform and reciprocal-orthogonal parametric transform. The spatial-domain encryption algorithms are based on the SPN (Substitution Permutation Network) that uses substitution and permutation to change pixel value and pixel position of the image respectively. Figure 1.1 show the encryption algorithm and its further classification.

1.3 Literature Review

In modren block ciphers an S-box is usually used to conceal the relationship between the ciphertext and the key, thus ensuring the Shannons [7] property of Confusion and diffusion. Therefore an S-box has a significant role in cryptography, consequently, it is vital to utilize a secure S-box, before using it in cryptography. Many techniques are use to create S-boxes with good cryptographic properties. Wang et al. [14] suggested a novel approach for the construction of an S-box using chaotic tent map. Tang et al. [15, 16] construct different S-boxes using chaotic maps. Zhang et al. [17] constructed an S-box by spatiotemporal chaotic system. Khan et al. [18] used chaotic partial differential equation to construct an S-box. Researchers and cryptographers proposed many methodologies and techniques for the construction of cryptographically strong S-boxes. In [19–25] many S-boxes were created by different scholars.

Research in this area has revealed that any reliable encryption method that use S-boxes generate S-boxes with cryptographically desired properties such as avalanche criteria, high non-linearity and bits independent criteria among others [26]. As a result, several approaches for the creation of this nonlinear component with all of these desirable characteristics have been proposed by researchers. To construct this nonlinear component, these approaches uses a variety of mathematical structures. Such as finite field, Galois ring, elliptic curves, and symmetry groups are all well-known mathematical structures [27].

In [28], Dragomir et al. proposed a method for constructing database or repositories systems of S-boxes with strong cryptographic topographies. These repositories

can be of great assistance in providing data and information security while customising block cyphers. In [29] author created a cryptographically good S-box based on the chaos theory and travelling salesman problems. Ahmad at el. [30] proposed a novel approach to construct cryptographically robust bijective S-boxes based on a new hyperchaotic system. When compared to other systems used for S-box construction, it was found that the new hyperchaotic system seemed to have good characteristics. Adams et al. [31] create s-boxes that are of bijection, SAC, BIC, NL, etc. A strong S-box design using chaotic maps has been proposed by Ozkaynak and Ozer [32]. They used the Lorentz system for the chaotic map and the proposed methodology is examined and evaluated against cryptographic criteria. The analysis results show the high reliability of the proposed cryptosystem, which is ideal for secure communication. Farwa et al. [33] proposed a simple but effective approach for generating an S-Box based on linear fractional transformation (LFT). The proposed S-box were analyzed for cryptographic properties such as nonlinearity (NL), bit independence criterion (BIC), strict avalanche criterion (SAC), linear probability (LP), differential probability (DP), etc.

The communication of data and information has grown to be a crucial component of modern technology and is now regarded as one of an individual or an organization's most valuable assets. As a result, information security has become extremely important in present-day sciences. Researchers are focused on developing effective and secure image encryption algorithms because of the rapid rise in image transmission. Researchers analyse, identify, and solve problems in the real world using image-based data. Wang at el. [34] suggests a new method for image encrypting images that is based on dynamic random growth and hybrid chaotic maps. Author uses dynamic random growth technique and the Arnold cat map cofuse the original image. In 2010 Patidar et al. [35] proposed a "substitution-diffusion based image encryption scheme" utilising chaotic logistic and standard maps. The proposed scheme was strong because it had properties of confusion and

diffusion. Francois et al. [36] in 2012 proposed a new image encryption algorithm that is based on the coupling of a chaotic function and the `XOR` operator. The major factors of his encryption algorithm were its capacity for producing a large key space, its ability to produce images with any entropy structure, and its confusion and diffusion properties.

A multiple image encryption method using the discrete wavelet transformation and nonlinear fractional mellin transformation was presented by Pan et al. [37]. Khan et al. [38] proposed an algorithm for constructing the nonlinear S-box used in image encryption algorithm. In the process of constructing nonlinear substitution components for image encryption, the chaotic Boolean bit function is used. Farwa et al. [39] proposed a new and reliable image encryption scheme that use then Arnold transform and algebraic S-box for scrambling. The proposed image encryption method is very easy to use and very efficient. In 2020 Lidong et al. [40] proposes a novel triple-image encryption scheme based on S-box, chaotic system and image compression. The security analysis shows tha the proposed scheme can effectively resist common cryptographic attack. Ali and Ali [41] proposed a new scheme based on chaotic maps for encryption using substitution, Boolean operation and permutation. In 2022 another scheme was proposed by Ali and Ali [42] for image encryption that uses a dynamic chaotic map and S-box.

These days, S-boxes are frequently used for security purposes and image encryption. Therefore, it is very important for the encryption algorithm to construct a good S-box. A good S-box is one that fulfills all cryptoghrapic properties.

1.4 Software Tools For S-box Analysis

A number of tools are available for investigating S-box characteristics. The main issue in those tools are: (i) not publicly available (ii) making them inaccessible to the general public (iii) Variation of results for the same test in different tools.

Here is a list of freely available tools for evaluating Boolean functions or S-boxes.

1. **Boolfun Package in R.** [43] R is a statistical computing and graphics environment that is free to use. It's compatible with Mac OS, Windows, and UNIX. "Albeit" the default version of R doesn't permit the evaluation of Boolean functions. A package called Boolfun may be loaded to offer functionality for the cryptographic analysis of Boolean functions [44, 45].

2. **Boolean Functions in Sage.** [46] Sage is a free and open-source mathematics software. Boolean functions in Sage module allows you to examine the cryptographic properties of Boolean functions. Most significant cryptographic features associated with differential and linear properties of Boolean functions may be evaluated with this tool.

3. **S-boxes in Sage.** [46] Sage is software that permits you to treat S-boxes algebraically. This module has a lot of features, but when it comes to cryptographic properties, the only ones that can be calculated are the linear approximation matrix and the difference distribution Table.

4. **VBF Library.** [47] VBF stands for Vector Boolean Functions. Library is also included for completeness. Zufiria and Alverez-Cubero proposed a method for cryptographically evaluating vectorial Boolean functions, that can be used to determine various S-box properties.

5. **S-box Evalution Tool (SET)** [48] stands for S-box Evaluation Program is an American National Standards Institute (ANSI) C-based tool for analysing the cryptographic properties of S-boxes and Boolean functions. This tool was developed by Stjepan Picek [48] and his colleagues to assess the cryptographic characteristics of S-boxes and Boolean functions. It's a free, open-source mathematics tool that is basic in nature and easy to use.

1.5 Thesis Objective

The objective of this thesis is to investigate the scheme proposed by Zahid et al. [49] for the construction of a strong S-box. The proposed scheme is based on cubic fractional transformation (CFT). On the bases of best knowledge it can be said that this method is the first method to create an S-box from the cubic fractional transformation (CFT). The properties of the generated S-box are examined, as well as the cryptographic strength of the constructed S-box, using the MATLAB and SET tools.

After successful construction of S-box of the reviewed scheme [49] it is then used for image encryption, by utilizing Logastic map. First, propoed S-box is used to substitute the pixel value of the digital image. Then two sequences are construct by using the same method which used for construction of S-box are construct. The sequences are utilized for the purpose of row and column wise circular shift. After this, Logistic map is iterate to generate a random sequence. At the end, cipherimage obtained by scrambling process with the help of random sequence.

The remaining thesis is organized as follows

- **Chapter 2** This chapter includes discussion on the basic mathematical concepts that are useful in cryptography such as Boolean functions, Galois field, their general properties, and how they contribute in the creation of strong S-boxes. Various cryptoraphic properties according to the general design principles of S-boxes are also explained.
- **Chapter 3** describes the concept of CFT and introduces a method for constructing S-boxes using CFT. After that, the properties of proposed S-box are analysed. The properties of constructed S-box are checked using **MATLAB** and **S-box Evaluation Tool (SET)**.
- **Chapter 4** describe image encryption scheme by using the proposed S-box constructed in Chapter 3 and Logistic map. Moreover, some security analysis of proposed scheme are discussed.
- **Chapter 5** gives the conclusion to the work of **Chapter 3** and **Chapter 4**.

Chapter 2

Preliminaries

This chapter will discuss and explain an introduction to cryptology and some basic mathematics that are useful in cryptography, as well as the same features of S-boxes. In Section 2.1, brief introduction of cryptography and some of the fundamental concepts of cryptography. Section 2.2 some basic definitions (Group, Ring, Field and Finite Field etc.) are described with examples. In sections 2.3 and 2.4 give some basic idea of Galois Field and Boolean function. In section 2.5, some cryptographic properties of strong S-box are described.

2.1 Cryptology

Cryptology is the study of secret communication. The word cryptology is derived from the Greek words "kryptos" which means "hidden", and "logos", which means "word." In cryptology, communication between two people is made secure. One of them is the sender **(Alice)** and the other one **(Bob)** is the receiver. The sender converts the data into incomprehensible form using secret keys (which are pieces of information that are only known to them) and sends it to the receiver. After receiving the data Bob got the original form of data using the secret key that

Alice used to incomprehensible the data. The original message is referred to as plaintext, while the incomprehensible message is referred to as ciphertext.
Cryptology is divided into two main categories:

1. **Cryptography**
2. **Cryptanalysis**

Cryptography is the process of converting original message or data into an incomprehensible text, and vice versa. It is a method of collecting and transmitting data in a particular configuration that can only be pursued by those who need it. Cryptography can also be used to verify user identity and protect data from theft or tampering.
In earlier cryptography, people used synonymously word in the place of different words for encryption, but currently, it is mostly based on mathematical hypotheses and computer applications. Many applications, including financial transactions, e-commerce transactions, and computer passwords use cryptography.
Plaintext, Ciphertext, Encryption Algorithm, Decryption Algorithm, and Key are the five essential components of a conventional cryptosystem.

1. **Plaintext:** It is the original form of data or message.
2. **Ciphertext:** It is the coded form of data or message.
3. **Encryption Algorithm (E):** It used to convert plaintext into ciphertext.
4. **Decryption Algorithm (D):** It sued to convert ciphertext into plaintext data.
5. **Key:** It is the special information used in encryption and decryption algorithms.

Two types of cryptographic techniques are used in general:

- **Symmetric Key Encryption**

- **Asymmetric Key Encryption**

Symmetric Key Encryption is a type of encryption in which both the sender and the recipient use the same key to encrypt and decrypt data. Secret-key encryption is another name of symmetric key encryption. The sender uses the key to encrypt plaintext and sends it to the recipient. The receiver, on the other hand, uses the same key to decrypt the encrypted message and recover the plaintext. Various examples of symmetric-key encryption algorithms are Data Encryption Standard (DES) [8], Advanced Encryption Standard (AES) [9], Triple DES [5], RC4 [50].

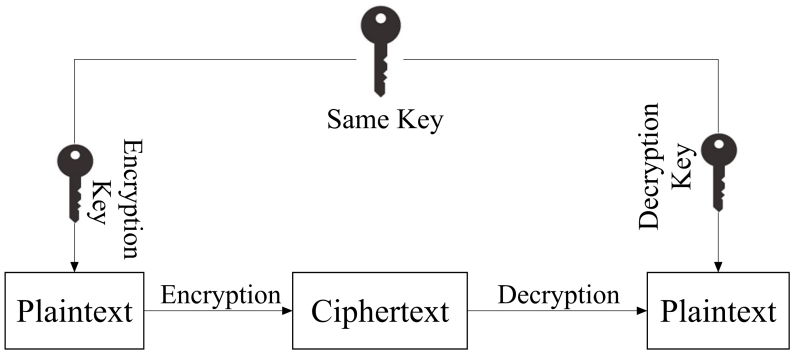

FIGURE 2.1: Symmetric-key encryption

The following encryption methods is used in symmetric key encryption:

1. **Stream Cipher:** A stream cipher is a text encryption symmetric-key scheme in which a cryptographic algorithm is applied to each binary digit in a data set by considering one bit at a time.

2. **Block Cipher:** A block cipher is a symmetric key encryption algorithm that changes a block of plaintext into an equal size block of ciphertext. The size of the block is fixed in the provided encryption technique. The block size has no effect on the encryption scheme's strength. The strength of the cipher is determined by the length of the key.

Asymmetric-key Encryption is a type of encryption in which data is encrypted and decrypted using a pair of keys. Asymmetric-key Encryption is also called public-key encryption.The first key is known as the public key, and it is accessible to everyone. The second key is known as the private key, and it must be kept hidden. The sender encrypts the plaintext with the public key, while the receiver decrypts ciphertext with the private key. Both keys are mathematically linked, but it is impossible to compute the private key using the public key. As a result, the receiver may be able to disseminate the public key widely. The public key can be used by anyone to encrypt communications for the receiver, but only the receiver can decode the information by using the private key. Examples of asymmetric-key encryption are McEliece [51], ElGamal [52] ,RSA [53]

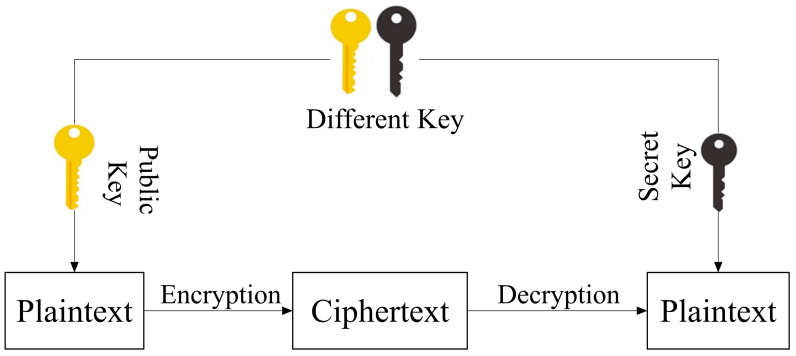

FIGURE 2.2: Asymmetric-key encryption

Cryptanalysis is the process of breaking codes to decode the information encoded. Cryptanalysis is normally thought of as looking for defects in the underlying mathematics of a cryptographic system; however, looking for implementation flaws for example side-channel threats or low entropy inputs is also part of the process. The examined information is utilized to investigate the system's hidden points. Somebody who endeavors to play out this undertaking is known as a cryptanalyst. Cryptanalyst utilizes a algorithm to decode ciphertext without knowing the plaintext sources or encryption keys. In addition, a cryptanalyst attempts to improve existing techniques by detecting holes in a security protocol. This can

be done by locating the key and enhancing the procedures if they lack the four properties of authentication, integrity, confidentiality, and non-repudiation.

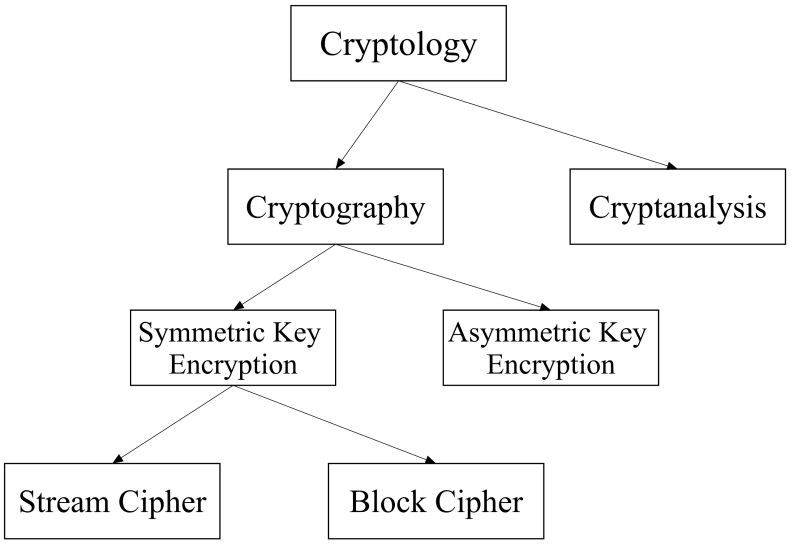

FIGURE 2.3: Clasification of cryptology

2.2 Mathematical Background

To understand the construction and performance of the S-boxes, some basic concepts of group, Ring, Field, Finite Field and Galois field are presented.

Definition 2.2.1.

Consider u and v be two integers where $v \neq 0$. If there exists a number w such that $u = vw$, we say that v divides u or that v is the **divisor** of u. We write $v|u$ to represent that v divides u. If v does not divides u, then it written as $v \nmid u$.

Definition 2.2.2.

Consider u and v to be two positive integers and w is called **greatest common divisor (GCD)** of both u and v, if w is their largest number which divides both u and v then it is represented as:

$$\gcd\ (u, v) = w$$

If greatest common divisor of u and v is one, they are said to be **relatively prime**.

The **Euclidean Algorithm** is helpful for determining the greatest common divisor of two positive numbers.
The Euclidean Algorithm for computing gcd (a,b) is as follows:

gcd $(u, b = v)$

1. $X = u \;\; ; \;\; Y = v$
2. if $Y = 0$ return $X = \gcd\,(u, v)$
3. $Z = X \bmod Y$
4. $X = Y$
5. $Y = Z$
6. go to 2

Example 2.2.3.
Compute gcd of $(4202, 3520)$. Evaluate gcd by using the Euclidean Algorithm.
First divide 4202 by 3520.

$$4202 = 3520(1) + 682$$

Next, divide 3520 by the remainder 682 and continue this process.

$$3520 = 682(5) + 110$$

$$682 = 110(6) + 22$$

$$110 = 22(5) + 0$$

So gcd $(4202, 3520) = 22$.

Example 2.2.4.
Compute gcd$(5429, 1567)$.

Using the Euclidean Algorithm.

$$5429 = 1567(3) + 728$$

$$1567 = 728(2) + 111$$

$$728 = 111(6) + 62$$

$$111 = 62(1) + 49$$

$$62 = 49(1) + 13$$

$$49 = 13(3) + 10$$

$$13 = 10(1) + 3$$

$$10 = 3(3) + 1$$

$$3 = 1(3) + 0$$

Hence gcd (5429, 1567)=1, therefore (5429,1567) are relatively prime.

Definition 2.2.5.

Fundamental Theorem of Arithmetic is define as: "Every natural number greater than 1 can be written as a product of primes, and the expression of a number as a product of primes is unique except for the order of the factors" [54]. Let X be a composite number then it can be written as:

$$X = y_1^{m_1} \times y_2^{m_2} \times y_3^{m_3} \times \cdots \times y_n^{m_n}$$

where $m_n \geq 0$ and $y_1, y_2, y_3, \cdots, y_n$ are prime numbers, written in an ascending order $y_1 \leq y_2 \leq y_3 \leq \cdots \leq y_n$. If prime number are same then they can be combined to give power of prime number.

Example 2.2.6.

Factorization of 20 and 15 can be performed as:

$$20 = 2 \times 2 \times 5 = 2^2 \times 5$$

$$15 = 3 \times 5$$

Definition 2.2.7.

The **Modular arithmetic** is a system of arithmetic for integers, in which numbers "wrap around" when reaching a given fixed number, this given number is called the modulus.

Let u, v and w are integers and w is modulus then mathematically written as:

$$u \equiv v \mod w$$

TABLE 2.1: Addition table of integer in mod 8

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	0
2	2	3	4	5	6	7	0	1
3	3	4	5	6	7	0	1	2
4	4	5	6	7	0	1	2	3
5	5	6	7	0	1	2	3	4
6	6	7	0	1	2	3	4	5
7	7	0	1	2	3	4	5	6

TABLE 2.2: Multiplication table of integers mod 8

×	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	0	2	4	6
3	0	3	6	1	4	7	2	5
4	0	4	0	4	0	4	0	4
5	0	5	2	7	4	1	6	3
6	0	6	4	2	0	6	4	2
7	0	7	6	5	4	3	2	1

Example 2.2.8.

Addition of element of the set of residue classes modulo 8 and multiplication of element of the set of residue classes modulo 8 are shown in Table 2.1 and Table 2.2, respectively.

Definition 2.2.9.

An integer v is said to be **multiplicative inverse** of another integer u modulo m, if $uv \equiv 1 \mod m$. The multiplicative inverse of a number u under modulo m exist if and only if $\gcd(u, m) = 1$

Example 2.2.10.

The multiplicative inverse of $23 \equiv 2 \mod 7$ under the modulo 7.

The $\gcd(23, 7) = 1$, therefore multiplicative inverse of 23 under the modulo 7 exists.

$$\begin{aligned} 23 \times 0 &\equiv 0, \mod 7 \\ 23 \times 1 &\equiv 2, \mod 7 \\ 23 \times 2 &\equiv 4, \mod 7 \\ 23 \times 3 &\equiv 6, \mod 7 \\ 23 \times 4 &\equiv 1, \mod 7 \end{aligned}$$

Hence multiplicative inverse of 23 under the modulo 7 is 4.

Definition 2.2.11.

A **binary operation** $\star$ on a set H is a function $\star : H \times H \to H$ such that for every $(u \star v) \in H \times H$ the binary operation $\star$ assigns a unique element $w \in H$, then we say that the pair $(H, \star)$ has a binary structure.

Definition 2.2.12.

The set H under operation $\star$ is called a **groupoid** if it is closed with respect to $\star$.

Example 2.2.13.

The set of even number E and the set of odd number O are groupoids with respect to addition.

Definition 2.2.14.

The set H under operation $\star$ is called a **semi group** if:

i) It is closed with respect to $\star$.

ii) It is associative with respect to $\star$.

Example 2.2.15.

Set of natural number $\mathbb{N}$ is a semi group with respect to addition.

Definition 2.2.16.

The set H under operation $\star$ is called a **monoid** if:

i) It is closed with respect to $\star$.

ii) It is associative with respect to $\star$.

iii) It has an identity element with respect to $\star$.

Example 2.2.17.

Set of natural number $\mathbb{N}$ is a monoid with respect to multiplication.

Definition 2.2.18.

The set H and $\star$ is an operation applied on it, then the structure $(H, \star)$ is said to be **group** if;

i) It is closed with respect to $\star$.

ii) It is associative with respect to $\star$.

iii) It has an identity element with respect to $\star$.

iv) Every element of H has an inverse element in H with respect to $\star$.

Definition 2.2.19.

A group under the operation of $\star$ is said to be **abelian or commutative** if it holds commutative property.

Example 2.2.20.

The following are the example of group:

i) Integers ($\mathbb{Z}$) is a group under the operation of addition.

ii) $\mathbb{R} - 0$ is a group under the operation of multiplication.

iii) The set of matrices M of order $(m \times n)$ i.e. $\mathrm{M}_{(m\times n)}$ $(\mathbb{R})$ is a group under the operation of addition.

iv) The set of integer under modulo m $\mathbb{Z}_m$ is a group under the operation of addition.

Definition 2.2.21.

For a nonempty set R with two binary operations '$+$' and '$\cdot$' (usually written as addition and multiplication), than the structure $(R, +, \cdot)$ is called **ring** if;

i) The set R under the binary operation of addition is an abelian group.

ii) It holds the associative property with respect to '$\cdot$'.

$$u \cdot (v \cdot w) = (u \cdot v) \cdot w \quad \forall\ u, v, w\ \in R$$

iii) Distributive law holds in R. i.e $\forall u, v, w \in R$

$$u \cdot (v + w) = (u \cdot v) + (u \cdot w) \qquad \text{(Left distributive law)}$$

$$(v + w) \cdot u = (v \cdot u) + (w \cdot u) \qquad \text{(Right distributive law)}$$

Example 2.2.22.

The following are the example of ring:

i) The set of real numbers $\mathbb{R}$ forms a ring $(\mathbb{R}, +, \cdot)$.

ii) The set of integers $(\mathbb{Z})$ forms a ring $(\mathbb{Z}, +, \cdot)$.

iii) The set $\mathbb{Z}_n$ of integers class modulo n forms a ring $(\mathbb{Z}_n, +, \cdot)$.

iv) The set of rational number $\mathbb{Q}$ forms a ring $(\mathbb{Q}, +, \cdot)$.

Definition 2.2.23.

For a non-empty set K with two operations '$+$' and '$\cdot$', then the structure $(K, +, \cdot)$ is said to be **field** if and only if the following conditions are satisfied:

i) The set K forms an abelian or commutative group with respect to addition.

ii) $K - \{0\}$ forms an abelian or commutative group with respect to multiplication.

iii) K holds distributive law. i.e $\forall\ u, v, w \in K$

$$u \cdot (v + w) = (u \cdot v) + (u \cdot w) \qquad \text{Left distributive law}$$

$$(v + w) \cdot u = (v \cdot u) + (w \cdot u) \qquad \text{Right distributive law}$$

Example 2.2.24.
The following are the example of field:

i) The set of real number $\mathbb{R}$ forms a field $(\mathbb{R}, +, \cdot)$.

ii) The set of rational number $\mathbb{Q}$ forms a field $(\mathbb{Q}, +, \cdot)$.

iii) The set of complex number $\mathbb{C}$ forms a field $(\mathbb{C}, +, \cdot)$.

iv) If $K = \{u + v\sqrt{3}\ \forall u, v \in \mathbb{R}\}$, then K is a field.

v) The set of $\mathbb{Z}_p$ is a field $(\mathbb{Z}_p, +, \cdot)$, where p is a prime number.

Definition 2.2.25.
A **finite field** is a field that has a finite number of elements.

2.3 Galois Field

A field with finite number of element is called **Galois Field**. The name of Galois field is in honour of the French mathematician Evariste Galois (1811-1832) [55]. The number of elements in a Galois field is 'y^n' where 'y' is prime number and 'n' is positive integer and denoted by $GF(y^n)$. For a prime number 'y' the set of integer $Z_y = \{0, 1, 2, 3, \cdots y - 1\}$ is called prime field and it is denoted by $GF(y)$. The Galois Field's elements are defined as following:

$$GF(y^n) = \{0, 1, 2, 3, \cdots, y - 1\} \cup$$

$$\{y, y+1, y+2, \cdots, y+y-1\}\cup$$

$$\{y^2, y^2+1, y^2+2, \cdots, y^2+y-1\}\cup\cdots$$

$$\cup\{y^{n-1}, y^{n-1}+1, y^{n-1}+2, \cdots, y^{n-1}+y-1\}$$

where 'y' and 'n' represent prime number and positive integer respectively. The order of the field is 'y^n' while the characteristics of the field is represented by 'y'. Each element has a polynomial of degree at most $n-1$.
From a cryptographical point of view, these cases are particularly interested as shown below:

- $GF(y)$, with $n = 1$
- $GF(2^n)$, with $y = 2$

Definition 2.3.1.
The elements of $GF(y^n)$, are the **polynomial** of degree atmost n.

$$f(y) = \sum a_i y^i \quad \forall \;\; i = 0, 1, 2, \cdots n$$

where a_i shows the coefficients and y^i are variables, the degree of polynomial is actually the highest power of w.

Definition 2.3.2.
The term **irreducible polynomial** refers to a polynomial $m(w)$ that can not be factorised as the product of more than one polynomials of lower degree. Otherwise, it is known as **"reducible polynomial"**.
The polynomials $w^3 + w^2 + 1$, $w^4 + w + 1$ and $w^5 + w^3 + 1$ are the examples of irreducible polynomials over $GF(2)$ whereas $w^3 + 1$, $w^2 + w$ are reducible polynomials over $GF(2)$.

Example 2.3.3.
Consider an irreducible polynomial $m(w) = (w^8 + w^6 + w^5 + w^4 + 1)$, and two polynomials $(w^7 + w^2 + 1)$ and $(w^6 + w^4 + w^2 + w + 1)$, then their product mod

$m(w)$ is:

$$\begin{aligned}(w^7+w^2+1)(w^6+w^4+w^2+w+1) &\mod (w^8+w^6+w^5+w^4+1)\\ =(w^{13}+w^{11}+w^9+w^7+w^3+w+1) &\mod (w^8+w^6+w^5+w^4+1)\\ =(w^4+w^3+w^2+w) &\mod (w^8+w^6+w^5+w^4+1)\end{aligned}$$

Example 2.3.4.

Consider an irreducible polynomial $m(w) = (w^8 + w^6 + w^5 + w^4 + 1)$, and two polynomials $(w^7 + w^4 + w^2 + 1)$ and $(w^4 + w^3 + w$, then their product mod $m(w)$ is:

$$\begin{aligned}(w^7+w^4+w^2+1)(w^4+w^3+w) &\mod (w^8+w^6+w^5+w^4+1)\\ =(w^{11}+w^{10}+w^7+w^6+w^4+w) &\mod (w^8+w^6+w^5+w^4+1)\\ =(w^6+w^5+w^4+w^3+w^2) &\mod (w^8+w^6+w^5+w^4+1)\end{aligned}$$

The following are 30 irreducible polynomials [56] of degree 8 with coefficients in GF(2^8) as given below:

1. $w^8+w^7+w^6+w^5+w^2+w+1$,
2. $w^8+w^7+w^6+w^5+w^4+w+1$,
3. $w^8+w^7+w^6+w^5+w^4+w^2+1$,
4. $w^8+w^7+w^6+w^5+w^4+w^3+1$,
5. $w^8+w^7+w^6+w^4+w^2+w+1$,
6. $w^8+w^7+w^6+w^4+w^3+w^2+1$,
7. $w^8+w^7+w^6+w+1$,
8. $w^8+w^7+w^6+w^3+w^2+w+1$,
9. $w^8+w^7+w^5+w^4+w^3+w^2+1$,
10. $w^8+w^7+w^5+w+1$,
11. $w^8+w^7+w^4+w^3+w^2+w+1$,
12. $w^8+w^7+w^5+w^3+1$,
13. $w^8+w^7+w^5+w^4+1$,
14. $w^8+w^7+w^3+w+1$,
15. $w^8+w^7+w^3+w^2+1$,
16. $w^8+w^7+w^2+w+1$
17. $w^8+w^6+w^4+w^3+w^2+w+1$
18. $w^8+w^6+w^5+w^4+w^2+w+1$
19. $w^8+w^6+w^5+w^4+w^3+w+1$
20. $w^8+w^6+w^5+w+1$
21. $w^8+w^6+w^5+w^2+1$
22. $w^8+w^6+w^5+w^3+1$

23. $w^8 + w^6 + w^5 + w^4 + 1$
24. $w^8 + w^6 + w^3 + w^2 + 1$
25. $w^8 + w^5 + w^4 + w^3 + w^2 + w + 1$
26. $w^8 + w^5 + w^3 + w^2 + 1$
27. $w^8 + w^5 + w^4 + w^3 + 1$
28. $w^8 + w^5 + w^3 + w + 1$
29. $w^8 + w^4 + w^3 + w^2 + 1$
30. $w^8 + w^4 + w^3 + w + 1$

2.4 Boolean Function

A Boolean function is a mapping $f : B^n \to B$, where n is non-negative integer and $B = \{0, 1\}$. The name of Boolean function is given in honour of the British mathematician G. Boole (1815-1864) [57]. A Boolean function can be written as n-tuples $(w_1, w_2, ...w_n) \to \{0, 1\}$, where $w_i \in B$ and $1 \leq i \leq n$.

2.4.1 Properties of Boolean Functions

In cryptography, Boolean functions are important elements used for the construction for S-Box. There have been many criteria for designing Boolean functions to resist for known cryptographic attacks. Boolean functions are widely utilized in stream ciphers' stream production mechanism etc. This section discusses some important properties for the Boolean function that makes it very useful for the cryptographic point of view.

Definition 2.4.1.
For a set $B = \{0, 1\}$, a **linear Boolean function** is a function $f : B^n \to B$ that can be define as:

$$f(w_1, w_2, \cdots, w_n) = b_1 w_1 \oplus b_2 w_2 \oplus \ \cdots \oplus b_n w_n$$

where $(b_1, b_2, \cdots, b_n) \in B$ and $\oplus$ is the `XOR` operation, $b_i w_i$ has `AND` Boolean function on the i^{th} bits of b and w [58]. The linear function is denoted by $L_b(w)$.

Definition 2.4.2.

An **affine function** is denoted by $A_{b,c}(w)$ and defined as:

$$A_{b,c}(w) = L_b(w) \oplus c$$

where $L_b(w)$ is linear Boolean function and $c \in [0, 1]$. In **Affine Cipher** affine function over modulo 'm' is used to encrypt the plaintext. Affine cipher is a of the simple example of substitution cipher. Affine cipher performs addition and multiplication by using the function;

$$f(w) = \Phi w \oplus \Psi \mod m$$

where Φ and Ψ are the secret keys which can also be written as (Φ, Ψ).

Definition 2.4.3.

If the number of zeros and ones in the corresponding truth table are equal, then the Boolean function $f : B^n \to B$ is called **balanced**.

TABLE 2.3: Truth Table of `XOR`, `AND` functions

w_1	w_2	$w_1 \cdot w_2$	$w_1 \oplus w_2$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Example 2.4.4.

To provide a comparison of balanced and unbalanced functions, for this consider the following two Boolean functions, `XOR` and `AND` defined as:

$$f_1 = \oplus : B^2 \to B$$

$$f_2 = \cdot : B^2 \to B$$

These function can be defined by the following truth table for two variables w_1 and w_2.

The last column of Table 2.3 has equal number of zeros to number of ones, represent that `XOR` function is balanced while the third column of the Table 2.3 has unequal number of zeros and ones, represent that `AND` function is not balanced.

Definition 2.4.5.

In a binary sequence the total number of non-zero digits tells the **Hamming weight** of that sequence. It is denoted by "$H(w)$", where $w \in [0,1]^n$.

For example: $w = 101001$ then $H(101001) = 3$.

The **Hamming distance** between two boolean functions $f(w)$, $g(w)$ is defined as follows:

$$d(f,g) = H(f(w) \oplus g(w))$$

Here,

$$f(w) \oplus g(w) = f(w_0) \oplus g(w_0) \oplus f(w_1) \oplus g(w_1) \oplus \cdots \oplus f(w_{2^n}-1) \oplus g(w_{2^n}-1)$$

where $w = (w_0, w_1, w_2, \cdots, w_{2^n}-1) \in B^n$.

Example 2.4.6.

Consider two Boolean functions:

$$f(w) = 1\ 1\ 1\ 0\ 1\ 0\ 1\ 1$$
$$g(w) = 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0$$

then the Humming distance between f and g is $d(f;g) = 5$

Example 2.4.7.

Consider two Boolean functions with $w = (w_1, w_2, w_3)$

$$f(w) = w_1 \cdot w_2 \cdot w_3 \text{ and } g(w) = w_1 \oplus w_2 \oplus w_3$$

with input bits w_1, w_2, w_3.

then the Hamming distance of these Boolean functions is

$$\begin{aligned} d(f;g) &= H(f(w) \oplus g(w)) \\ &= H(w_1 \cdot w_2 \cdot w_3 \oplus w_1 \oplus w_2 \oplus w_3) \end{aligned}$$

TABLE 2.4: The Hamming distance of two Boolean functions "f" and "g"

w_1	w_1	w_3	$w_1 \cdot w_2 \cdot w_3$	$w_1 \oplus w_2 \oplus w_3$	$(f \oplus g)$
1	0	0	0	1	1
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	1	1	0
0	0	0	0	0	0
0	0	1	0	1	1
0	1	0	0	1	1
0	1	1	0	0	0

Thus from Table 2.4 Hamming distance of "f" and "g" is calculated as 3.

Definition 2.4.8.

The Boolean function's **Walsh-Hadamard transformation** is denoted by the symbol WHT_f and is defined as:

$$WHT_f(\beta) = \sum (-1)^{f(w) \oplus \beta \cdot w} \qquad \forall \quad \beta,\ w \in B^n$$

where $B = \{0,1\}^n$ and $\beta \cdot w$ represent the inner product of vectors β and w.

Example 2.4.9.

Walsh Hadamard transform with Boolean function of,

$$f(w) = w_1 w_2 w_3 \oplus w_1 w_4 \oplus w_2$$

is given in the Table 2.5 below:

TABLE 2.5: Truth table of WHT_f

$w = w_1w_2w_3w_4$	$f(w)$	$(-1)^{f(w)}$	$dim3$	$dim2$	$dim1$	$dim0$
0 0 0 0	0	1	2	4	0	0
0 0 0 1	0	1	0	0	0	0
0 0 1 0	1	-1	-2	-4	8	8
0 0 1 1	1	-1	0	0	0	8
0 1 0 0	0	1	2	0	0	0
0 1 0 1	0	1	0	0	0	0
0 1 1 0	1	-1	-2	0	0	0
0 1 1 1	0	1	0	0	0	0
1 0 0 0	0	1	0	0	0	4
1 0 0 1	1	-1	2	4	4	-4
1 0 1 0	1	-1	0	0	0	4
1 0 1 1	0	1	-2	0	4	-4
1 1 0 0	0	1	0	0	0	-4
1 1 0 1	1	-1	2	0	-4	4
1 1 1 0	1	-1	0	0	0	0
1 1 1 1	1	-1	2	-4	4	-4

So the walsh transform of f is 12.

Definition 2.4.10.

A **Walsh Hadamard matrix** H is an "$n \times n$" matrix with element ± 1 of order n. If first column and row of a Hadamard matrix are all one then its called normalized [59]. The rows and column of Walsh Hadamard matrix are pairwise orthogonal.

Hadamard matrices were first constructed by James Joseph Sylvester in 1867 [60]. The Walsh Hadamard matrices of dimension 2^n are given by the recursive formula. Some examples of Walsh Hadamard matrix are given below.

$$H(2^1) = H(2) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$H(2^2) = H(4) = \begin{bmatrix} H(2) & H(2) \\ H(2) & -H(2) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

$$H(2^3) = H(8) = \begin{bmatrix} H(4) & H(4) \\ H(4) & -H(4) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}$$

In general

$$H(2^n) = \begin{bmatrix} H(2^{n-1}) & H(2^{n-1}) \\ H(2^{n-1}) & -H(2^{n-1}) \end{bmatrix}$$

2.5 Cryptographic Properties of a Strong S-box

S-box is an important tool of symmetric cryptographic algorithms. Its cryptographic properties are important in many encryption scheme algorithms security cipher such as AES [61], DES [10]. Diffusion and Confusion [7] are important properties of a block cipher such as AES [61], DES [10], etc. Generally an S-box has $n \times m$ bits in which n-bits are taken an input to produce m-bits as an output

using some bijective function. Constructing an S-box that fulfills the linear and differential characteristics is critical. There are well-known requirements that a good S-box must meet in for the cipher to be resistant against differential and linear cryptanalysis. A strong S-box has the following properties.

1. **Balanced:** If the number of zeros and ones in an S-box with n input and m output bits are equal, then the S-box is balanced [62].

2. **Non-linearity:** The non-linearity of a Boolean function $f(w) : B^n \rightarrow B$ indicates the number of bits that changed into the truth table to approach the nearest affine function [33], where $B = \{0, 1\}$.
   Non-linearity of an S-box ensures the protection against the attack of linear cryptanalysis. Non-linearity of a Boolean function is represented by $NL(f)$ and calculated as:

$$NL(f) = 2^{n-1} - \frac{1}{2}\left(\max\left(WHT_f(\beta)\right)\right)$$

   where, $WHT_f(\beta)$ is the Walsh Hadamard transformation of Boolean function $f(w)$, which is calculated as:

$$WHT_f(\beta) = \sum (-1)^{f(w)\oplus\beta\cdot w} \qquad \forall \quad \beta,\ w \in B^n$$

   where $\beta \cdot w$ is the inner product of vectors β and w.

3. **Bent Function:** Rothaus introduced bent Boolean function in 1976 [63]. A bent function is a maximally nonlinear Boolean function with an even number of variables.

4. **Strict Avalanche Criterion (SAC):** Webster and Tavares [64] proposed the strict avalanche criterion (SAC) in 1985. The SAC is defined as the output bit changed by $\frac{1}{2}$ when a single input is changed. The value of SAC nearer to 0.5 is deemed suitable.
   A function $f : B^m \rightarrow B^n$ fulfills the SAC if $\forall\ i, j \in (1, 2, 3, \cdots, m)$ flipping input bit i change the output j with the probability $\frac{1}{2}$. The SAC is satisfied

by an S-box if and only if the following conditions are met:

$$\frac{1}{2^n}W(a_j^{e_i}) = \frac{1}{2} \quad \forall\ i, j$$

where

$$W(a_j^{e_i}) = \sum_{\forall X \in B^m} a_j^{e_i}$$

and $a_j^{e_i} \in B$, where $B = \{0, 1\}$.

5. **Bit Independence Criterion (BIC):** Tavares and Webster [64] also introduced the BIC. By the definition of BIC two output bits change independently when a single input bit is inverted.
   A function $f : B^m \rightarrow B^n$ fulfills the bit independent criterion if $\forall\ i, j, k \in \{1, 2, 3, \cdots, m\}$ with $j \neq k$, the change in an input bit 'i' causes a in two output bits j and k independently.
   "To measure the bit independence concept the correlation coefficient is needed between the j^{th} and k^{th} components of the output difference string, which is called the avalanche vector A^{ei}. A bit independence parameter corresponding to the effect of i^{th} input bit change on the j^{th} and k^{th} bits of A^{ei} is defined as [65]

$$BIC(a_j, a_k) = \max_{1 \leq i \leq n} |corr(a_j^{e_i}, a_k^{e_i})|$$

   Overall, the bit independence criterion parameter for S-box function f is then found as

$$BIC(f) = \max_{\substack{1 \leq j,\ k \leq n \\ j \neq k}} BIC(a_j, a_k)$$

   which demonstrates how close f is to satisfying the BIC. $BIC(f)$ takes values in B. It is ideally equal to 0 and, in the worst case, it is equal to 1."

6. **Algebraic Degree** A Boolean function f with n-variable can be represented in various forms. One of the simplest form is representaed as polynomials over $GF(2)$. This is referred to as a Boolean function's algebraic normal form (ANF). This polynomial's degree represents the algebraic degree or

simply the degree of a Boolean function. The maximum algebraic degree of a Boolean function with n-variable balanced function is $n-1$ [66].

7. **Correlation Immunity (CI):** Siegenthaler [67] proposed correlation immunity of functions to protect against correlation attacks many shift register-based stream ciphers. The n-variable Boolean function has correlation immunity if $\forall\ i \in \{1, 2, 3, ..., n\}$

$$\text{Prob}(f(w_1,\ w_2, \cdots, w_n) = w_i) = \frac{1}{2}$$

for $(w_1,\ w_2, \cdots, w_n)$ randomly picked from $\{0,1\}^n$.

The high correlation immunity value is a powerful tool for retaining against the Siegenthaler correlation attack. [67]. The CI of the concept of statistical independence is designed with a stronger constraint.

8. **Algebraic Immunity (AI):** Let $g(w) \in F_n$ and there exists a Boolean function $h(w) \in F_n$ and $h(w) \neq 0$, then algebraic immunity is define as the lowest degree of function $h(w)$ such that

$$g(w)h(w) = 0 \text{ or } \big(g(w) \oplus 1\big)h(w) = 0$$

where function $h(w)$ is known as annihilator of $f(w)$ if $g(w)h(w) = 0$.

Meier et al. [68] was the first to introduce AI. The expression $\text{AI}\big(f(w)\big)$ represents the function $f(w)$ algebraic immunity. The Boolean function used in a cryptographic system needs to have high algebraic immunity in order to fend off algebraic attacks. "It is well known that the algebraic immunity of an n-variable Boolean function is upper bounded by $\frac{n}{2}$" [69].

9. **Fixed Points:** For an $n \times m$ S-box $S : GF(B^n) \rightarrow GF(B^m)$ and for $w \in GF(B^n)$, then the point is called fixed point of S-box if

$$S(w) = w$$

Example 2.5.1.

Consider a 2×2 S-box with 2 Boolean functions as shown in Table 2.6

TABLE 2.6: S-box fixed point

$GF(2)$	Binary format $GF(2)$	S-box	Binary format S-box
0	00	1	01
1	01	3	11
2	10	2	10
3	11	0	00

Here in the above Table 2.6, it can be seen that "1" is a fixed point of S-box.

10. **Opposite Fixed Points:** For an $n \times m$ S-box $S : GF(B^n) \rightarrow GF(B^m)$ and for $w \in GF(B^n)$, the point is called opposite fixed point of S-box if

$$S(w) = \overline{w}$$

where $\overline{w}$ is the bit-wise complement of w.

Example 2.5.2.

Consider a 2×2 S-box with 2 Boolean functions as shown in Table 2.7

TABLE 2.7: S-box opposite fixed point

$GF(2)$	Binary format $GF(2)$	S-box	Binary format S-box
0	00	1	01
1	01	2	10
2	10	3	11
3	11	0	00

In this example discussed in Table 2.7 "1" is an opposite fixed point of S-box.

Any S-box without fixed and opposite fixed points is considered as more resistant to differential cryptanalysis attacks when compared to S-boxes with fixed and opposite fixed points.

11. **Absolute Indicator and Sum of Square Indicator:** The Boolean function $f : GF(B^n) \rightarrow GF(B)$ with absolute indicator is defined as:

$$\Delta_{f(w)} = \max_{a \in GF(B^n), a \neq 0} \mid \Delta_{f(w)}(a) \mid$$

For a Boolean function $f : GF(B^n) \rightarrow GF(B)$, the sum of square indicator is defined as:

$$\sigma_{f(w)} = \sum_{a \in GF(B^n)} \left(\Delta_{f(w)}(a)\right)^n$$

The above indicators are known as the "global avalanche characteristics" of two Boolean function [70]. Where $\Delta_{f(w)}$ is Auto-correlation (AC) of a Boolean function $f(w)$ and can be defined on all $a \in GF(B^n)$ as:

$$\Delta_{f(w}(a) = \sum (-1)^{f((w) \oplus f((w \oplus a)} \qquad \text{where } (w \in GF(B^n)$$

12. **Linear Probability (LP):** The S-box's resistance against linear attacks is evaluated by linear probability. When the nonlinearity of the S-box increases, cryptanalysis attacks become more difficult.
"The value of LP of an S-box is eveluated as" [71]:

$$\text{LP} = \max_{C_w, D_w \neq 0} \left| \frac{\#\{w \in Z | w \cdot C_w = S(w) \cdot D_w\}}{2^n} - \frac{1}{2} \right|$$

where C_w and D_w are the input and output masks and $Z = \{0, 1, 2, ..., 2^n - 1\}$.

13. **Differential Probability (DP):** The S-box's resistance against differential attacks is evaluated by differential probability. The value of DP of an substitution box is calculated as:

$$\text{DP}_h(\Delta d \rightarrow \Delta x) = \max_{C_w \neq 0, D_w} \left(\frac{\#\{d \in C | h(d) \oplus h(d \oplus \Delta d) = \Delta w\}}{2^n} \right)$$

where C is the total number of inputs possible and 2^n is the number of elementes and Δd and Δy respectively, represent the input and output differentials. [72].

Chapter 3

S-box Construction Using Cubic Fractional Transformation (CFT)

Zahid et al [49], in 2019 proposed a method for constructing of efficient S-box by using cubic fractional transformation (CFT). This chapter discusses the techniques of constructing an S-box using cubic fractional transformation. The simulation and comparison analyses demonstrate that the suggested method for constructing S-box yields effective S-box for use in block ciphers. **S-box Evaluation Tool (SET)** [48] and **MATLAB** has been used to analyze the cryptographic characteristics of the S-box.

3.1 Construction of S-box

Rapid progress in communication technology and transfer of sensitive information through internent has become a challenge. For this purpose many beneficial encryption techniques have been developed. There are two types of cryptographic encryption algorithms: symmetric encryption algorithms and asymmetric encryption algorithms. AES [61] and DES [61] are the most popular symmetric encryption algorithms.

The literature that is currently available makes it clear that S-boxes generated

using random approaches, such as chaos or some other pseudo-random source, are not found to have strong cryptographic properties when compared to S-boxes generated using algebraic methods [73]. Another area that assists in the generation of new S-Boxes is linear fractional transformation (LFT). In [33, 73–76] authors propose effective algorithms to construct better S-boxes based on LFT. A LFT can be define as:

$$L(w) = \frac{v_1 w + v_2}{v_3 w + v_4}, \qquad w \in Z_n \tag{3.1}$$

where, the four values $v_1, v_2, v_3, \text{and} v_4$ are from a finite field $GF(2^8)$. Linear fraction transformation is also called Mobius transformation.

3.1.1 Cubic Fractional Transformation (CFT)

As in Section 3.1 many techniques are mentioned that are used to contruct proposed S-boxes. Many techniques have been used for construction of a strong non-linear S-box such as the chaotic and algebraic structures. In this section, an extended idea of LFT is used to generate an effective proposed S-box. This extended transformation is known as cubic fractional transformation (CFT). CFT is given as:

$$F(w) = \frac{1}{uw^3 + v} \mod (2^n + 1) \text{ where } u,\ v,\ w \in Z_n \tag{3.2}$$

and $Z_n = \{0, 1, 2, \cdots, 2^n - 1\}$, both $u, v \neq 0$ at the same time also $uw^3 + v \neq 0$. When using it to generate the output value $F(w)$, this should be avoided. During the operation of the proposed scheme, the existence of the CFT discontinuity point is gingerly checked when creating the S-box elements. Because of nonlinear nature of the CFT, it is frequently use in byte substitution. For the construction of the S-box Equation (3.2) is utilized. For $n = 8$ so, Z_n becomes $\{0, 1, 2, \cdots, 255\}$.
For the construction of the S-box presented in this thesis the value of u and v are chosen as $u = 95,\ v = 15$ then Equation (3.2) become

$$F(w) = \frac{1}{95w^3 + 15} \mod 257 \tag{3.3}$$

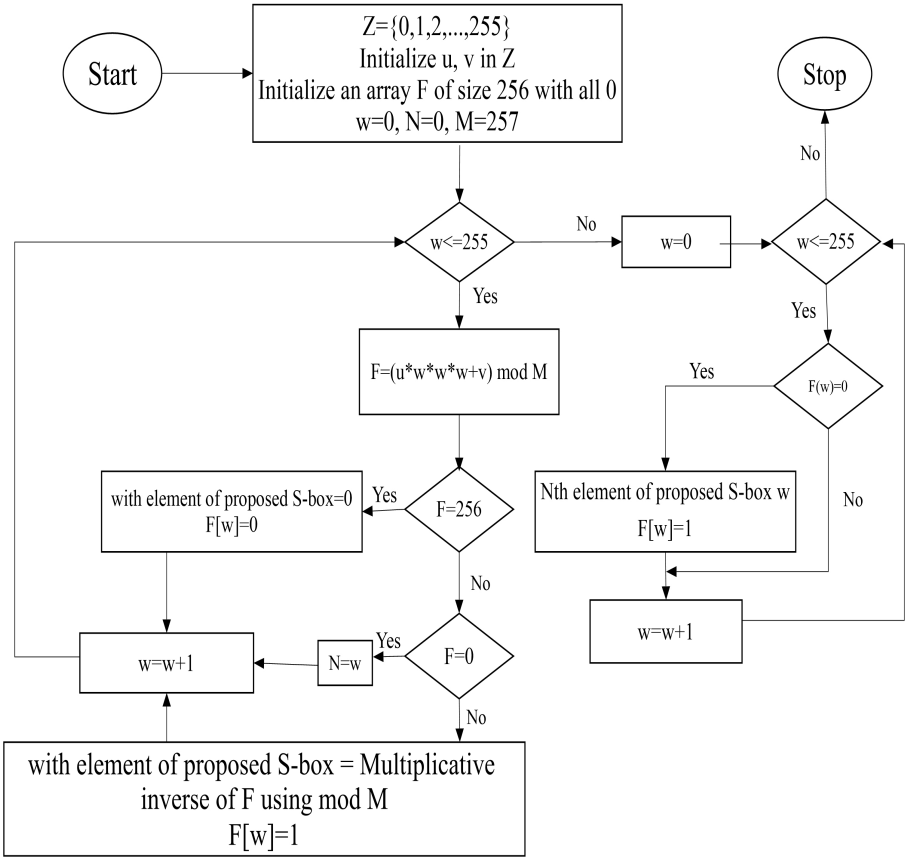

FIGURE 3.1: Flow chart of proposed S-box

The Equation (3.3) gives the value of $Z_n - \{0, 106\}$ when $w \in Z_n - \{176, 184\}$. This Eq, 3.3 gives $256 \notin Z_n$ at $w = 176$, and for $w = 184$ the denominator of Equation (3.3) becomes 0 which contradicts the condition $uw^3 + v \neq 0$. Hence the modified CFT becomes as following:

$$F(w) = \begin{cases} \dfrac{1}{uw^3 + v} \mod 257 & \text{if } w \in Z_n - \{176, 184\} \\ 0 & \text{if } w = 176 \\ 106 & \text{if } w = 184 \end{cases} \tag{3.4}$$

This modified CFT as presented in Equation (3.4) generates the element of proposed S-box, which are arranged in 16×16 matrix in Table 3.1.

TABLE 3.1: S-Box

78	FA	C1	B4	58	DF	B9	70	D2	F2	E9	F1	5B	5F	35	AE
84	73	7D	DC	4A	87	BE	50	48	68	2B	08	EF	26	C2	BA
B7	99	1F	A0	74	9D	72	A5	30	0D	34	DD	F4	3F	18	77
2E	AB	A9	9E	09	B1	2A	7B	8C	7A	6F	D8	F5	62	46	C5
CB	EB	A8	BB	0E	1A	89	8A	65	3C	E1	64	71	1C	C3	92
1D	C7	BD	56	D6	66	C8	27	B2	BF	E3	2C	1B	0F	F6	8D
90	86	FF	13	16	CC	12	8B	52	23	9C	39	D1	B5	4F	5D
BC	E7	CE	61	4D	80	8F	9B	A7	3B	D0	AF	FD	03	49	DA
3E	3D	2F	9F	4E	44	88	7E	3A	24	98	FC	F9	2D	43	E5
36	38	63	06	5E	C6	91	E2	AD	F7	22	0B	55	57	F8	76
C0	D5	85	D4	ED	15	5C	14	D7	79	DB	31	6D	32	EE	40
00	B0	42	01	4C	FE	96	DE	6A	81	CD	28	C4	7F	E6	B3
9A	45	1E	A3	21	0A	04	37	02	69	07	75	47	41	51	FB
94	AA	B6	D9	E8	EC	97	7C	E0	11	83	29	A6	A1	60	B8
6B	53	A2	25	82	AC	E4	4B	19	67	F0	93	6C	CF	D3	EA
20	6E	33	17	10	C9	CA	A4	0C	54	95	F3	8E	05	5A	59

As already mentioned that any value of u and v $(u, v \in Z_n)$ can be used in Equation (3.2) to construct the elements of an S-box. The process to construct the S-box is shown in Figure 3.1 for $n = 8$, $u = 95$ and $v = 15$.

3.1.2 Inverse S-box

The S-box is simply run in reverse to create the inverse S-box. For example, the inverse S-box value of 78 is $B0$. Table 3.2 shows the inverse S-box element of the proposed S-box.

TABLE 3.2: Inverse S-Box

B0	B3	C8	7D	C6	FD	93	CA	1B	34	C5	9B	F8	29	44	5D
F4	D9	66	63	A7	A5	64	F3	2E	E8	45	5C	4D	50	C2	22
F0	C4	9A	69	89	E3	1D	57	BB	DB	36	1A	5B	8D	30	82
28	AB	AD	F2	2A	0E	90	C7	91	6B	88	79	49	81	80	2D
AF	CD	B2	8E	85	C1	3E	CC	18	7E	14	E7	B4	74	84	6E
17	CE	68	E1	F9	9C	53	9D	04	FF	FE	0C	A6	6F	94	0D
DE	73	3D	92	4B	48	55	E9	19	C9	B8	E0	EC	AC	F1	3A
07	4C	26	11	24	CB	9F	2F	00	A9	39	37	D7	12	87	BD
75	B9	E4	DA	10	A2	61	15	86	46	47	67	38	5F	FC	76
60	96	4F	EB	D0	FA	B6	D6	8A	21	C0	77	6A	25	33	83
23	DD	E2	C3	F7	27	DC	78	42	32	D1	31	E5	98	0F	7B
B1	35	58	BF	03	6D	D2	20	DF	06	1F	43	70	52	16	59
A0	02	1E	4E	BC	3F	95	51	56	F5	F6	40	65	BA	72	ED
7A	6C	08	EE	A3	A1	54	A8	3B	D3	7F	AA	13	2B	B7	05
D8	4A	97	5A	E6	8F	BE	71	D4	0A	EF	41	D5	A4	AE	1C
EA	0B	09	FB	2C	3C	5E	99	9E	8C	01	CF	8B	7C	B5	62

3.2 Properties and Analysis of the Proposed S-box

Substitution boxes have a prominent place in both modern and ancient public key cryptography ciphers. In modern cryptography encryption scheme based on cryptosystms of 4-bit and 8-bit S-boxes have vital importance. Robust cryptosystem is based on strong S-boxes. Cryptographically strong S-boxes have high non-linearity, bijection, linear and differential probability, strict avalanche and bit independence criterion, and differential probability.

To show the strength of the suggested S-box its performance results are compared to those S-boxes that have already been studied. The S-box has a number of important cryptographic properties, which are listed below.

1. **Bijection:** A function $f : B^m \rightarrow B^n$ is bijective if for all $u \in B^m$ there exists only a unique element $v \in B^n$, where $B = \{0, 1\}$. In other words, when $u = v$, then $f(u) = f(v)$ or equiuvalently $u \neq v$, then $f(u) \neq f(v)$. The proposed S-box is analyzed for the bijective property by using SET [48] tool and it is found that it is bijective.

2. **Balanced:** The proposed S-box is tested for the balanced property using the SET [48] tool, and it is discovered that all of the components of the proposed S-box Boolean function are balanced.

TABLE 3.3: Comparing the non-linearity values of various S-boxes

S-Box Method	Minimum	Maximum	Average
Vaicekauskas et al. [77]	98	108	102.5
Mahmoud et al. [78]	96	110	104.3
Hussain et al. [74]	98	108	104
Alkhaldi ei al. [79]	98	108	104
Chen et al. [80]	102	106	104
Belazi et al. [81]	102	108	105.3
Mahmood et al. [82]	100	110	105.5
Siddiqui, et al. [83]	104	106	105.3
Hussain et al. [84]	100	108	105.7
Hussain et al. [85]	100	108	104.8
Hussain et al. [86]	94	104	99.5
Proposed	104	108	106.75

3. **Non-linearity:** It is not advisable for an S-box operation to map an input to an output linearly because this reduces the security of any cipher. The $NL(f)$ of the proposed substitution box are given in Table 3.4. The maximum non-linearity for an ideal S-box $GF(2^8)$ is 120. The most important cryptographic property of S-box is non-linearity. Below Table 3.3 show the comparison of different S-boxes with proposed S-box.

From Table 3.3 it is clearly shown that the average non-linearity is 106.75 which is greater then all other S-boxes [74, 77–86].

TABLE 3.4: S-box and non-linearity

f	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
$NL(f)$	106	108	108	108	108	106	104	106

4. **Strict Avalanche Criterion (SAC):** An acceptable SAC value is the one that is closer to 0.5. In order to fulfill the requirement, the Boolean function must be 50 percent dependent on each of its input bits. The value of the SAC of the S-box is given in Table 3.6 and it is observed that the maximum value of the SAC of the S-box is 0.578125 and minimum value of the SAC of the S-box is 0.421875 and the SAC of the S-box has an average value of 0.496582 which is approximately 0.5.

TABLE 3.5: BIC Value of S-box

0.0	102	106	104	102	102	100	104
102	0.0	104	106	98	100	104	108
106	104	0.0	102	104	102	108	100
104	106	102	0.0	98	106	108	102
102	98	104	98	0.0	104	102	108
102	100	102	106	104	0.0	104	108
100	104	108	108	102	104	0.0	104
104	108	100	102	108	108	104	0.0

5. **Bit Independence Criterion (BIC):** If a given S-box fulfill the BIC, all of the components of a boolean function satisfy the SAC and have high non-linearity. The Table 3.5 demonstrates the values of the BIC for all Boolean functions of the S-box. The maximum and the minimum value for the Boolean function of the S-box is 108 and 98 receptively and the average value of BIC is 103.571.

TABLE 3.6: SAC Value of S-box

0.453125	0.453125	0.515625	0.484375	0.515625	0.531250	0.531250	0.484375
0.468750	0.468750	0.468750	0.468750	0.515625	0.531250	0.437500	0.468750
0.515625	0.578125	0.468750	0.468750	0.468750	0.562500	0.515625	0.531250
0.531250	0500000	0.453125	0.546875	0.453125	0.453125	0.453125	0.468750
0.531250	0.515625	0.468750	0.515625	0.515625	0.468750	0.421875	0.531250
0.578125	0.531250	0.578125	0.453125	0.531250	0.437500	0.438750	0.484375
0.5546875	0.546875	0.500000	0.453125	0.515625	0.421875	0.484375	0.562500
00.531250	0.437500	0.484375	0.484375	0.453125	0.531250	0.484375	0.546875

6. **Linear Probability (LP):** The S-box's resistance against linear attacks is evaluated by linear probability. The lower LP increases the strength of the S-box against the linear attack.

   "The value of LP of an S-box is eveluated as" [71]:

$$\mathrm{LP} = \max_{C_w, D_w \neq 0} \left| \frac{\#\{w \in Z | w \cdot C_w = S(w) \cdot D_w\}}{2^n} - \frac{1}{2} \right|$$

   where C_w and D_w are the input and output masks and $Z = \{0, 1, 2, ..., 2^n - 1\}$.

   The S-box has maximum value of LP 0.1484.

7. **Differential Probability (DP):** The S-box's resistance against differential attacks is evaluated by differential probability. An S-box with a lower DP is more resistant to differential cryptanalysis.

   The Table 3.7 shows the value of DP of the S-box. The maximum value of DP of the S-box 10 and the DP of the S-box is 0.0391.

8. **Bent Boolean:** By using the SET [48] it is found that Bent Non-linearity value of proposed S-box is 116.6863

9. **Algebraic Immunity (AI):** The algebraic immunity of proposed substitution box is 4.

10. **Fixed Points:** The proposed S-box has no fixed point.

11. **Opposite Fixed Points:** The proposed substitution box has no opposite fixed point.

12. **Absolute Indicator and Sum of Square Indicator:** The absolute indicator of substitution box is 104, where the sum of square indicators of S-box is 289792.

13. **Algebraic Degree:** The higher the degree of a function, the more complex its algebraic structure and the more resistant it is to low approximation attacks. It is suggested that algebraic degree is equal to or greater than 4 to avoid higher-order differential cryptanalysis. The algebraic degree of proposed substitution box is 7.

TABLE 3.7: DP Value of S-box

0.0	6.0	6.0	6.0	6.0	6.0	8.0	8.0	6.0	6.0	8.0	6.0	10.0	6.0	6.0	6.0
8.0	8.0	8.0	6.0	8.0	6.0	8.0	8.0	6.0	6.0	6.0	6.0	6.0	6.0	8.0	6.0
6.0	6.0	6.0	8.0	6.0	8.0	8.0	6.0	6.0	6.0	6.0	8.0	6.0	6.0	6.0	6.0
8.0	6.0	10.0	6.0	6.0	8.0	6.0	8.0	6.0	8.0	6.0	4.0	6.0	6.0	8.0	8.0
6.0	6.0	6.0	6.0	8.0	6.0	8.0	8.0	8.0	6.0	8.0	8.0	8.0	6.0	8.0	6.0
6.0	6.0	6.0	6.0	4.0	6.0	6.0	6.0	6.0	6.0	6.0	8.0	6.0	6.0	6.0	6.0
6.0	6.0	6.0	6.0	6.0	6.0	6.0	8.0	8.0	6.0	6.0	6.0	6.0	6.0	8.0	10.0
6.0	8.0	6.0	6.0	6.0	8.0	6.0	8.0	6.0	6.0	6.0	6.0	8.0	6.0	10.0	6.0
6.0	8.0	8.0	6.0	6.0	6.0	8.0	6.0	6.0	6.0	8.0	6.0	8.0	6.0	10.0	10.0
6.0	8.0	6.0	8.0	6.0	8.0	8.0	6.0	8.0	6.0	8.0	6.0	10.0	8.0	6.0	8.0
8.0	6.0	6.0	6.0	8.0	6.0	6.0	6.0	8.0	6.0	6.0	6.0	6.0	6.0	6.0	8.0
6.0	8.0	8.0	6.0	8.0	8.0	6.0	8.0	6.0	6.0	8.0	6.0	6.0	6.0	6.0	8.0
8.0	6.0	6.0	6.0	8.0	6.0	8.0	6.0	6.0	6.0	6.0	6.0	6.0	8.0	6.0	8.0
6.0	6.0	6.0	6.0	6.0	6.0	8.0	8.0	8.0	6.0	8.0	8.0	6.0	6.0	8.0	6.0
6.0	6.0	6.0	6.0	8.0	6.0	8.0	8.0	8.0	6.0	6.0	6.0	8.0	6.0	6.0	8.0
6.0	8.0	8.0	6.0	6.0	6.0	6.0	8.0	8.0	8.0	6.0	6.0	6.0	6.0	6.0	8.0

Chapter 4

Logistic Map and S-box Based Image Encryption

With the development of internet technology, a lot of information is transferred from one end to the other in the form of data and images. Security should be provided to ensure the confidentiality of any information, whether it relates to the military, the defense, or the medical field. With the advancement of image encryption technology comes the advancement of image information theft technology [87]. To keep up with the developing information theft technology, a better image encryption algorithm is required.

An image is an artwork that illustrates visual perception, such as a photograph or other 2-dimensional depiction. It is described as a 2-variable function, $f(x, y)$, where each position in the image plane (x, y), corresponds to the light intensity at that position. There are two types of images: digital image and analog image. Analog images are represented mathematically as a continuous range of values representing position and intensity. The Digital image is composed of the picture element known as Pixel. Each pixel in an image has consists intensity number, or a tiny set of numbers that define some feature of the pixel, like its brightness or colour. It is used in everyday life, such as satellite television and magnetic resonance imaging, as well as in research, such as astronomy and geographical information systems.

The process of encrypting an image using an encryption algorithm is known as image encryption. The successful transmission of sensitive information using encrypted images has led to the development of countless techniques. However, despite the phenomenal rise in the use of images in all forms of digital communication, it still draws researchers.

4.1 Basic Terminologies

There are some basic terms and terminalogies that are commonly used for image encryption schemes. These terminalogies are discussed as following:

4.1.1 Digital Image

A digital image is a numeral portrayal of an image that can be stowed and processed by a digital computer. The numeral representation of an image is divided into small parts that are called pixels. Each pixel in an image has consists intensity number, or a tiny set of numbers that define some feature of the pixel, like its brightness or colour. The numbers are grouped in an assemblage of rows and columns that delineate the image's vertical and horizontal pixel positions.

4.1.1.1 Pixel

A pixel (pix-el) is abbreviation of picture element. A digital image is composed of many of these pixels. This means that the image is a collection of different pixels. The colors in any pixel are the functions of red, green, and blue portions.

4.1.2 Component of Image Encryption Cryptosystem

1. **Plainimage:** This is the type of image that requires protection while being transmitted over a public network. It is also referred to as the source or input image.

2. **Cipherimage or Encrypted Image:** A cipherimage is the converted form of a plainimage into an unintelligible form after encryption.

3. **Encryption:** A plainimage is converted into a cipherimage using an encryption method and a secret key in this process.

4. **Decryption:** The cipherimage is converted to a plainimage by the receiver side using a decryption method and a secret key. This is referred to as decryption.

5. **Key:** The key determines the encryption method's security. It can be either numeric or alphanumeric. The key is required for both encryption and decryption to be performed. Strong keys are always required for better information security.

4.2 Chaotic Map

The chaotic map is a mathematical structure that shows some kind of chaotic behavior. Random sequences are generated using a chaotic map. A discrete-time and continuous-time parameter can be used to parameterize the maps. Discrete maps are typically iterated functions. The output of such maps is very dependent on the control parameters and initial conditions. When using chaotic maps in cryptography, these parameters can be treated as secret keys. Many chaotic maps exhibit chaotic behavior on a specific control parameter region.

Chaotic cryptology is divided into two branches: chaotic cryptography and chaotic cryptanalysis. Chaotic cryptography encrypts information, while chaotic cryptanalysis decrypts the encrypted messages. Chaos-based cryptography has gained so much attention that it now has a vast range of uses in different of fields, including telecommunications, medical imaging, video, image, DNA cryptography, and others.

4.2.1 Lyapunov Exponent

In the study of dynamical systems, the term 'Lyapunov Exponent' (LE) [88] is commonly used. LE is a quantitative measure of the sensitivity to initial conditions. The degree of divergence between two close trajectories of a dynamical system is described by LE. A positive LE indicates that, no matter how near the two trajectories are, the divergence between them grows with each iteration, in the end making them be totally unique. LE is defined as follows:

$$\Lambda = \lim_{m\to\infty} \frac{1}{m} \sum_{j=1}^{m-1} \ln |f'(\hat{x})| \tag{4.1}$$

To obtain an average LE, the LE can be evaluated for sample points near the attractor. "If the average LE is negative, then the system is periodic; if at least one of the average LE is positive, then the system is chaotic; if the average LE is zero, a bifurcation occurs. The more chaotic a system, the higher the value of the LE" [89].

4.2.2 Bifurcation Diagram

A bifurcation occurs, which tells a period-doubling change from an M-point attractor to a $2M$-point attractor. A bifurcation diagram is a visual representation of the succession of period-doublings that occurs as γ increases. By displaying the parameter value against all related equilibrium values, the bifurcation diagram is created. If a small change in the bifurcation parameters results in a large change in the system behaviour, the bifurcation appears. This characteristic occurs in both continuous and discrete systems.

4.3 Logistic Map

The Logistic map is a second degree polynomial mapping. Is is one dimensional chaotic map [90]. Its mathematical structure is simple, but its chaotic behaviour

is complex. Logistic mapping is a non-linear, discrete-time, one-dimensional map with quadratic non-linearity. It is defined by the equation below:

$$t_{n+1} = \gamma \times t_n \left(1 - t_n\right) \tag{4.2}$$

where $\gamma \in [0, 4]$ and $t_n \in (0, 1)$. The value of γ is the most essential part of the logistic map expression because it determines the chaotic behaviour:

- When $\gamma \in (0, 1)$, the points on the logistic map's expression graph approach 0, independently of the value t_0. [91].
- When $\gamma \in [1, 2)$,the points on the logistic map's expression graph approach the value $\frac{\gamma-1}{\gamma}$ [91].

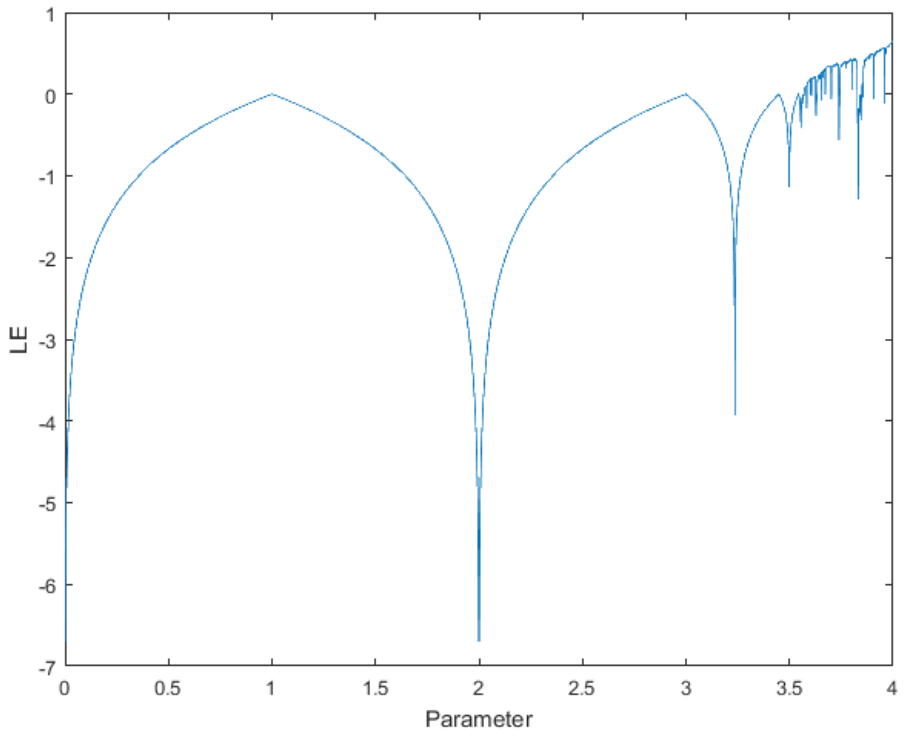

FIGURE 4.1: Lyapunove exponent of Logistic map

- When $\gamma \in [2, 3)$, the points on the logistic map's expression graph approach the value $\frac{\gamma-1}{\gamma}$, but on an initial phase, they will vary around this value [91].

- When $\gamma \in [3, 3.44949)$, the points on the logistic map's expression graph vary between two values, lets denote them t_1 and t_2. t_1 and t_2 are independent of γ [91].
- When $\gamma \in [3.44949, 3.54409)$, the points on the logistic map's expression graph will vary between four values, t_1, t_2, t_3, t_4. When γ increases until it has a value of approximately 3.56995, the points on the logistic map's expression graph will vary between eight values, then sixteen values, and so on [91].
- When $\gamma \in [3.56995, 4]$,the points on the logistic map's expression graph are placed chaotically, and it is said that the map is in the chaotic state. The value of n for which the map is stopped is called the dimension of the logistic map [91].

Figure 4.1 shows the logistic map's Lyapunove exponent. Figure 4.2 shows the logistic map's bifurcation diagram.

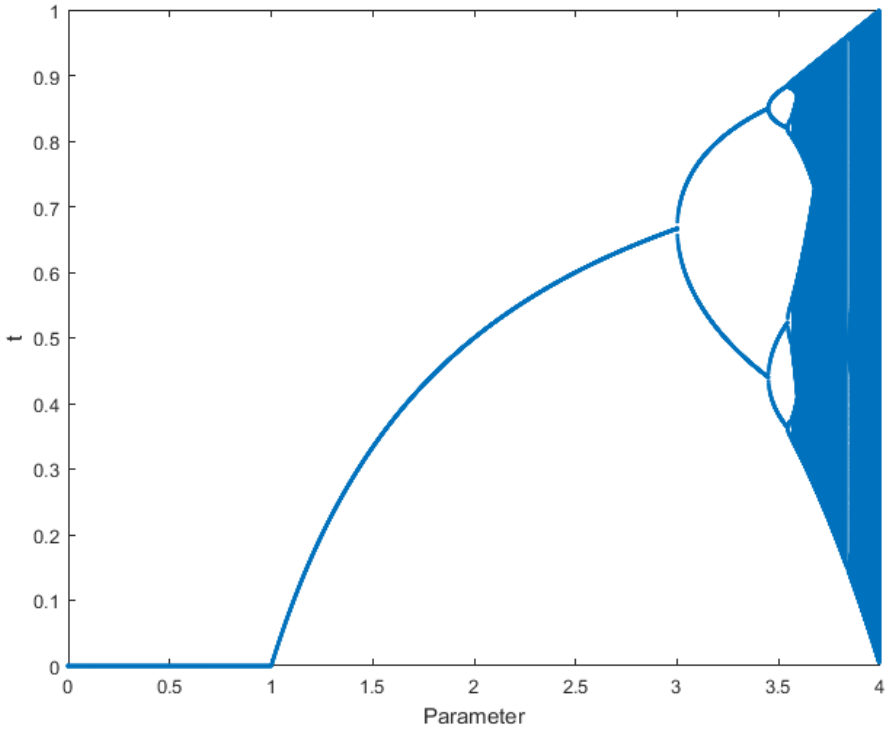

FIGURE 4.2: Bifurcation diagram for the Logistic map with iterations = 65536

4.4 Image Encryption Algorithm

In this section, S-box and logistic map based encryption algorithm is suggested for image encryption. In this algorithm, an S-box is generated by using a cubic fraction transformation. S-box generation algorithm is described in the previous Chapter 3. The gray-scale image of size $n \times m$, where n and m represent the number of rows and columns of a gray-scale image respectively. First, the image is converted into its pixel matrix. Then design of the proposed S-box based encryption algorithm is described. Then, a row and column circular permutation is applied. At the end, the scrambling process is applied by taking bit-wise `XOR` with secret key that is constructed by using the logistic map.

4.4.1 Encryption Algorithm (Gray-scale)

For the encryption purpose a gray-scale image I of size $(n \times m)$ of clock is used, where n and m are the number of rows and columns of a I respectively. The size of the gray-scale image that is used for encryption is 256×256. The encryption process is described below:

Algorithm 1: Substitution algorithm

Input: Image (I), Secret key k_0, S-box generated by CFT Chapter 3

Output: Substituted Image

1 Read the gray-scale secret image I.

2 Create a digital form of the gray-scale image I with a size of M, where $n \times m$ is the size of image matrix with integer entries lies in the range of $[0, 255]$ and n, m are the rows and columns in M respectively.

3 Using the secret key $k_0 = (u_0, v_0)$ for CFT Chapter 3 to generate an S-box 3.1 take only those values that lies in [0, 255].

4 Replace each entry of pixel value matrix m_i of the image $M = \{m_1, m_2, m_3, \ldots, m_i\}$ by $S(m_i)$ where $i \in \{1, 2, 3, \ldots, n \times m\}$. After substitution of the pixel value matrix with S-box, the resultant array is denoted by S' which is $\{s_1, s_2, s_3, \ldots, s_i\}$.

Permutation Algorithm: A suitable permutation is used to change the pixel positions of the substituted image S'. The Cubic Fractional Transformation (CFT) is used to generate the permutation array.

$$F(w) = \begin{cases} \dfrac{1}{uw^3 + v} \mod (257) & \text{if } w \in Z_n - \{43, 135\} \\ 0 & \text{if } w = 43 \\ 11 & \text{if } w = 135 \end{cases} \tag{4.3}$$

Here $u = 90$ and $v = 20$. The Equation (4.3) gives the value of $Z_n - \{0, 11\}$ when $w \in Z_n - \{43, 135\}$.

$$F(w) = \begin{cases} \dfrac{1}{uw^3 + v} \mod (257) & \text{if } w \in Z_n - \{176, 62\} \\ 0 & \text{if } w = 142 \\ 40 & \text{if } w = 62 \end{cases} \tag{4.4}$$

Here $u = 115$ and $v = 45$. The Equation (4.4) gives the value of $Z_n - \{0, 40\}$ when $w \in Z_n - \{142, 62\}$.

Algorithm 2: Permutation Algorithm

Input: S', Secret key k_1 and k_2, CFT Equations (4.3) and (4.4)

Output: Pre-encrypted Image P

1 Utilize the same algorithm as used in Chapter 3 to construct the elements of the sequences $S1 = \{s_{11}, s_{12}, s_{13}, \ldots, s_{1m}\}$ and $S2 = \{s_{21}, s_{22}, s_{23}, \ldots, s_{2n}\}$ by using Equations (4.3) and (4.4) and secret keys $k_1 = (u_1, v_1)$ and $k_2 = (u_2, v_2)$.

2 Apply a column-wise circular shift on substitution image S'. Use the value of S1 to permute the value of substitution image S'. The resulting matrix is stored in P'.

3 After applying a column-wise circular shift on image S', another row-wise circular shift is applied on image P' with the help of S2. The resulting matrix is saved in P.

After the permutation layer, a Logistic map (4.2) and a Boolean operation `XOR` is used to get final encrypted image.

Algorithm 3: Scrambling Process

Input: Pre-encrypted image P, Logistic Map (4.2) and Secret key k_3

Output: Final Encrypted image C

1 Iterate the logistic map (4.2) using initial state t_0 and control parameter γ as secret key k_3 for W times to create a chaotic sequence of length W.

2 Convert the obtained sequence by iterating Logistic map into 8-bit integer values using the following relation:

$$X = \mod(\text{floor}(x \times 10^{14}), 256)$$

3 To eliminate the transient impact, dispose off the first $(L - (n \times m))$ numbers of the chaotic sequence, then generate a new chaotic sequence of length 65536, denoted by SK.

4 Rewrite matrix P as one dimensional array $P'' = \{p''_1, p''_2, p''_3, \ldots, p''_{n\times m}\}$.

5 Then the scrambling process is performed as following:

for $i = 1$ *and* $j = 1 : n \times m$
  if $j = 1$, then
    $C(i,j) \leftarrow P''(i,j) \oplus Sk(i,j)$;
  else
    $C(i,j) \leftarrow C(i,j-1) \oplus P''(i,j) \oplus Sk(i,j)$;
  end
end

Covert the array into $n \times m$ matrix obtained from scrambling.

6 In the end, a cipherimage C is created by converting the resulting matrix of pixel values into an image format.

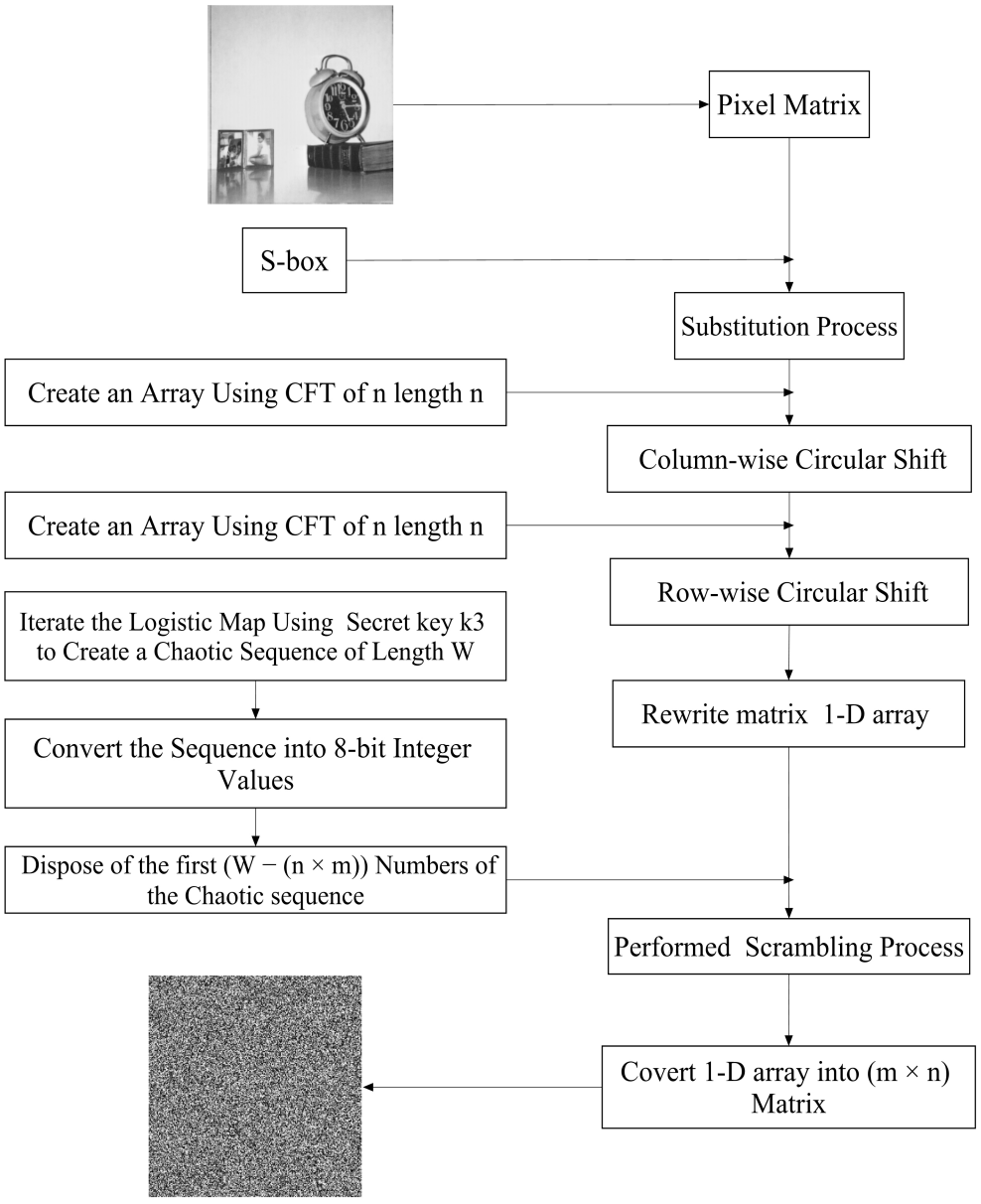

FIGURE 4.3: Flow chart of image encryption algorithm

4.5 Decryption Algorithm (Grayscale)

The decryption algorithm eridicate all encryption effects from the original image. The following steps are used to recover the plainimage I from the cipherimage C:

Algorithm 4: Decryption Algorithm

Input: Cipherimage, Secret key k_1, k_2 and k_3

Output: Original Image I

1 After receiving the cipherimage, the receiver converts the cipherimage into digital form N, where N is the pixel value matrix of $n \times m$ size with entries from 0 to 255.

2 Convert the encrypted image into one dimensional array of length $n \times m$.

3 Perform the unscrambling process as described below:

$$
\begin{aligned}
&\text{for } i = 1 \;\; and \;\; j = 1 : m \times n \\
&\quad \text{if } \; j = 1, \text{then} \\
&\qquad P''(i,j) \leftarrow C(i,j) \oplus SK(i,j); \\
&\quad \text{else} \\
&\qquad P''(i,j) \leftarrow (i, j-1) \oplus C(i,j) \oplus SK(i,j); \\
&\quad end \\
&\quad end
\end{aligned}
$$

Rewrite P'' into 2-dimension matrix P.

4 Construct the sequence $S1 = \{s_{11}, s_{12}, s_{13}, \ldots, s_{1m}\}$ and $S2 = \{s_{21}, s_{22}, s_{23}, \ldots, s_{2n}\}$ by using Equations (4.3) and (4.4) of Cubic Fractional Transformation and secret key $k_1 = (u_1, v_1)$ and $k_2 = (u_2, v_2)$.

5 Apply a reverse row-wise circular shift on P by using the element of $S2$. Then resulting matrix is stored in P'.

6 Another reverse column-wise circular shift is applied with the help of the array $S1$. Hence generated resulting matrix is stored in S'.

7 Generate an S-box 'S' by using the method mentioned in Chapter 3.

8 Construct the inverse S-box of S that is S^{-1}.

9 Replace each pixel value s'_i of the matrix S' by using inverse S-box generated in step 8,as $S^{-1}(s'_i)$ where $i \in \{1, 2, 3..., n \times m\}$. After replacing the pixel values of S' with the inverse S-box, the resultant pixel matrix is I.

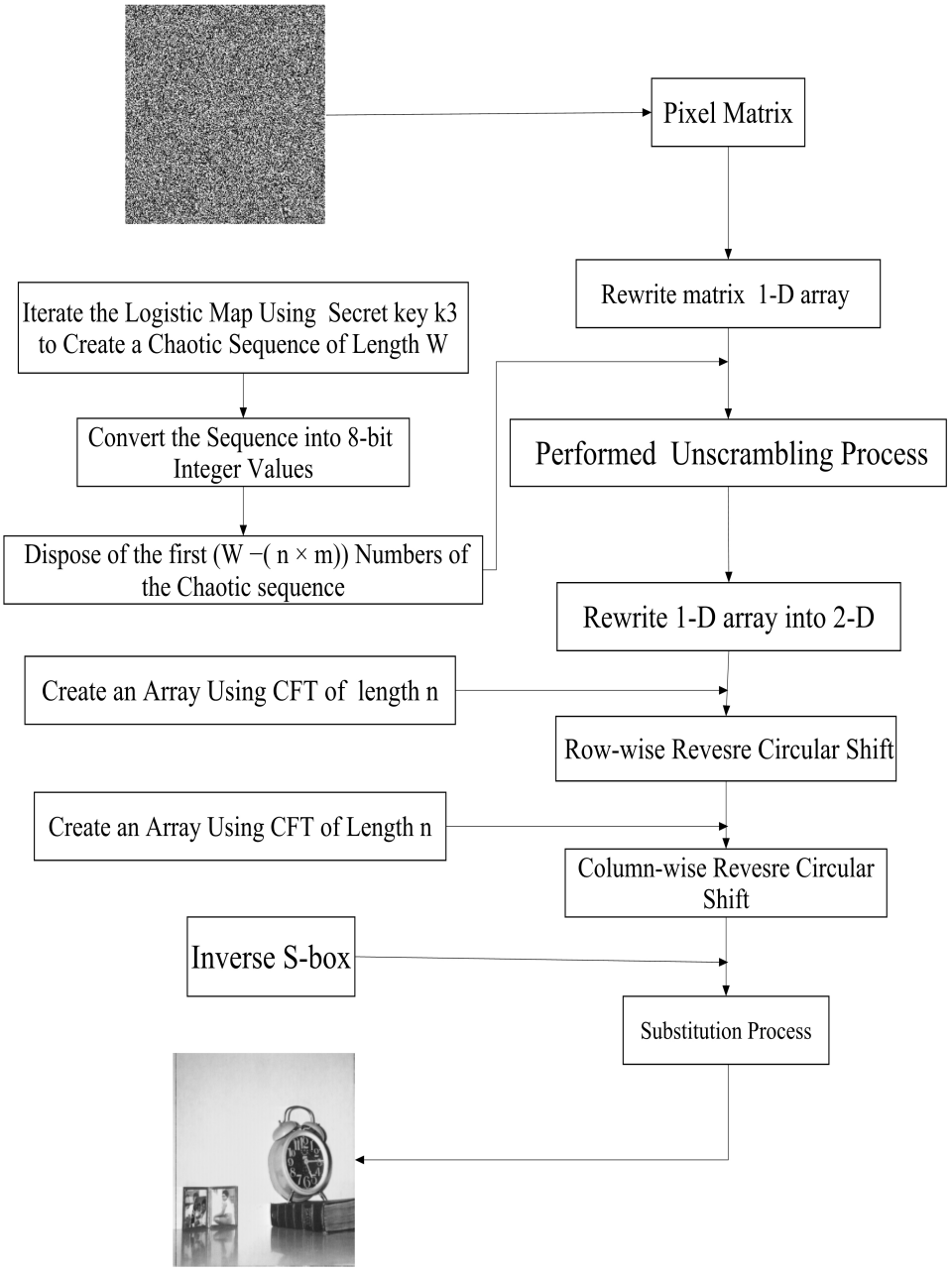

FIGURE 4.4: Flow chart of image decryption algorithm

Example 4.5.1.

The encryption process is illusted through the following toy example.

Let the pixel matrix of an image I given. For simplicity, a 4×4 hypothetical image is considered.

$$I = \begin{bmatrix} 184 & 161 & 179 & 186 \\ 175 & 118 & 118 & 120 \\ 158 & 126 & 140 & 105 \\ 137 & 123 & 98 & 82 \end{bmatrix}$$

Encryption Phase

By applying the S-box from Table 3.1 to the image I as described in Step 5 of the Algorithm (1), the resulting image S' is;

$$S' = \begin{bmatrix} 106 & 213 & 1 & 205 \\ 64 & 143 & 143 & 167 \\ 87 & 73 & 249 & 35 \\ 36 & 175 & 255 & 189 \end{bmatrix}$$

After applying S-box, the new resulting image S' is then permuted by using circular shift. Hence generated pre-encrypted image P. Using Equations (4.3) and (4.4) generate two arrays $S1$ and $S2$ of length m and n respectively. These arrays are shown as follows:

$$S1 = \{4, 1, 3, 2\}, \qquad S2 = \{3, 2, 1, 4\}$$

Then apply column-wise circular shift operation by using $S1$ array on S'. This results a p' image. Then another row-wise circular shift operation is applied by using $S2$ on P'. Hence obtained the new matrix is P. That is a pre-encrypted image.

$$P' = \begin{bmatrix} 106 & 143 & 255 & 35 \\ 64 & 73 & 1 & 189 \\ 87 & 175 & 143 & 205 \\ 36 & 213 & 249 & 167 \end{bmatrix}, \qquad P = \begin{bmatrix} 35 & 106 & 143 & 255 \\ 1 & 189 & 64 & 73 \\ 175 & 143 & 205 & 87 \\ 36 & 213 & 249 & 167 \end{bmatrix}$$

Converting this matrix P into one dimensional array P'' as shown below:

$$P'' = \{35,\ 1,\ 175,\ 36,\ 106,\ 189,\ 143,\ 213\ ,143,\ 64,\ 205,\ 249,\ 255,\ 73,\ 87,\ 167\}$$

Iterate the logistic map (4.2) to generate a random sequence SK by using secret key k_3 containing γ, t_o where $\gamma = 3.995$ and $t_o = 0.15$.

$SK = \{13,\ 86,\ 38,\ 144,\ 9,\ 233,\ 176,\ 172,\ 112,\ 230,\ 56,\ 169,\ 161,\ 110,\ 115,\ 184\}$

Then the scrambling process is performed by taking `XOR` of each element with its corresponding element of key matrix and also with its preceding element as illustrated in Step 5 of the Algorithm (3) to get cipherimage as following:

$C = \{46,\ 121,\ 240,\ 68,\ 39,\ 115,\ 76,\ 53,\ 202,\ 108,\ 153,\ 201,\ 151,\ 176,\ 148,\ 139\}$

Convert this formed array C into 2-dimension matrix form. The resulting cipher matrix is given below.

$$C = \begin{bmatrix} 46 & 39 & 202 & 151 \\ 121 & 115 & 108 & 176 \\ 240 & 76 & 153 & 148 \\ 68 & 53 & 201 & 139 \end{bmatrix}$$

Decryption Phase

The decryption process demonstrate as following.
Consider C as a pixel matrix of a cipherimage.

$$C = \begin{bmatrix} 46 & 39 & 202 & 151 \\ 121 & 115 & 108 & 176 \\ 240 & 76 & 153 & 148 \\ 68 & 53 & 201 & 139 \end{bmatrix}$$

Convert the C into one dimensional array as:

$C = \{46,\ 121,\ 240,\ 68,\ 39,\ 115,\ 76,\ 53,\ 202,\ 108,\ 153,\ 201,\ 151,\ 176,\ 148,\ 139\}$

Iterate the logistic map (4.2) to generate the random sequence SK by using secret key k_3 containing γ, t_o where $\gamma = 3.995$ and $t_o = 0.15$.

$SK = \{13,\ 86,\ 38,\ 144,\ 9,\ 233,\ 176,\ 172,\ 112,\ 230,\ 56,\ 169,\ 161,\ 110,\ 115,\ 184\}$

Apply the unscrambling process by taking `XOR` operation of each element of C with corresponding element of array SK and also with its preceding element as illustrated in Step 3 of the Algorithm (4). The obtaining array is stored in P''.

$P'' = \{35,\ 1,\ 175,\ 36,\ 106,\ 189,\ 143,\ 213\ ,143,\ 64,\ 205,\ 249,\ 255,\ 73,\ 87,\ 167\}$

Rewrite P'' into 2-dimension matrix P.

$$P = \begin{bmatrix} 35 & 106 & 143 & 255 \\ 1 & 189 & 64 & 73 \\ 175 & 143 & 205 & 87 \\ 36 & 213 & 249 & 167 \end{bmatrix}$$

Using Equations (4.3) and (4.4) to generate two array $S1$ and $S2$ of length m and n respectively. These arrays are shown as follows:

$$S1 = \{4, 1, 3, 2\}, \qquad S2 = \{3, 2, 1, 4\}$$

Then apply reverse row-wise circular shift by using array $S2$ on P. After applying reverse row-wise circular shift the resulting matrix is stored as P'. Then another reverse column-wise circular shit applied by using the element of array $S1$ on P'. Hence the resulting matrix is S'.

$$P' = \begin{bmatrix} 106 & 143 & 255 & 35 \\ 64 & 73 & 1 & 189 \\ 87 & 175 & 143 & 205 \\ 36 & 213 & 249 & 167 \end{bmatrix}, \qquad S' = \begin{bmatrix} 106 & 213 & 1 & 205 \\ 64 & 143 & 143 & 167 \\ 87 & 73 & 249 & 35 \\ 36 & 175 & 255 & 189 \end{bmatrix}$$

By using the method mentioned in Chapter 3 generate the element of an S-box S. Construct the inverse S-box of S that is S^{-1}. Then applying the inverse S-box to the image S' as described in Step 9 of the Algorithm (4) gives I as;

$$I = \begin{bmatrix} 184 & 161 & 179 & 186 \\ 175 & 118 & 118 & 120 \\ 158 & 126 & 140 & 105 \\ 137 & 123 & 98 & 82 \end{bmatrix}$$

Finally, image I can be obtained in its original form.

4.6 Results and Discussions

The tests are performed to demonstrate the efficiency and validity of the suggested algorithm. This algorithms (1), (2) and (3) have been applied on different images such as Clock, House, Girl and Chemical Plant using MATLAB R2017a. These images are from the http://sipi.usc.edu/database/, known as the USC-SIPI open image repository.

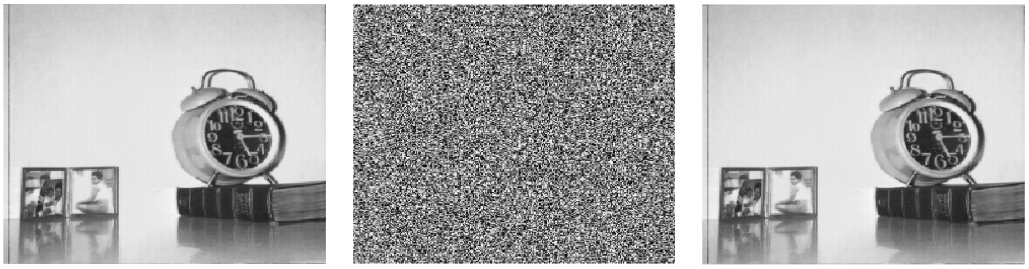

FIGURE 4.5: Results of Clock Image encryption and decryption algorithm: (a) plainimage, (b) Encrypted Image, (c) Decrypted Image

The standard gray image of Clock (256×256) is used in first example shown in Figure 4.5(a). After applying the encryption algorithm the cipherimage is depicted in Figure 4.5(b). The decryption algorithm (4) is then used for decoding the cipherimage, as shown in Figure 4.5(c). The decryption result demonstrates that the proposed algorithm is effective in recovering the original image.

In the next example a Chemical Plant image of sizes (256×256) is chosen. The image is displayed in Figure 4.6(a), the encryption result is shown in 4.6(b). The

decryption algorithm is then used to perform the decoding the cipherimage, as shown in Figure 4.6(c). The decryption result demonstrates that the proposed algorithm is effective in recovering the plainimage.

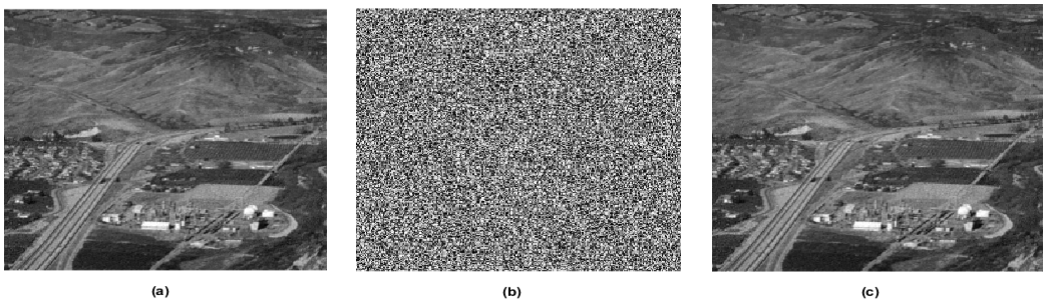

FIGURE 4.6: Results of Chemical Plant image encryption and decryption algorithm: (a) plainimage, (b) Encrypted Image, (c) Decrypted Image

4.6.1 Security Analysis

There are some security assessments that examine the quality of encryption these includ key space, key sensitivity, differentioal analysis correlation coefficients, entropy and histogram analysis.

4.6.1.1 Key Space

Any cryptosystem considers key space as an important feature. It must be large enough to withstand brute force attacks. The key space $10^{30} \approx 2^{100}$ [92] is the recommended for high security and protection from brute-force attacks. One common attack is the brute-force attack, in which an assailant endeavours to figure out the right security keys by exorbitantly looking of an encryption algorithm excessively. As a result, an adequate and enormous key space makes sure that the encryption algorithm is resistant to a brute-force attack. In encryption algorithm, a Cubic Fractional Transformation (CFT) and Logistic map are used. CFT involve u and v as parameter and Logistic map involves γ as control parameters and t as a variable. The key is a comprising of 4 secret keys $k = (k_0, k_1, k_2, k_3)$. The number of

possible key combinations is 2^{147}. The resulting keyspace exceeds the minimum key size requirement.

4.6.1.2 Key Sensitivity Analysis

The secret key should be well known to the image encryption scheme, and changing a single bit of the secret key should result in a completely different encrypted result. An effective image encryption scheme must be key sensitive in order to prevent unauthorized preliminary attacks. Even if the encryption key changes slightly, the cipher image should not be decrypted accurately. For example, by adding 0.000000000000005 to the parameter "γ" of the secret key in the encryption scheme, the parameter's new value will be 3.995000000000005. The original image will not be obtained by using this to decrypt.
The plainimage of Clock is represented in Figure 4.7(a), the encrypted image in Figure 4.7(b), and the decryption results are achieved with a slightly different key in Figure 4.7(c). This shows that the algorithm is extremely sensitive to key.

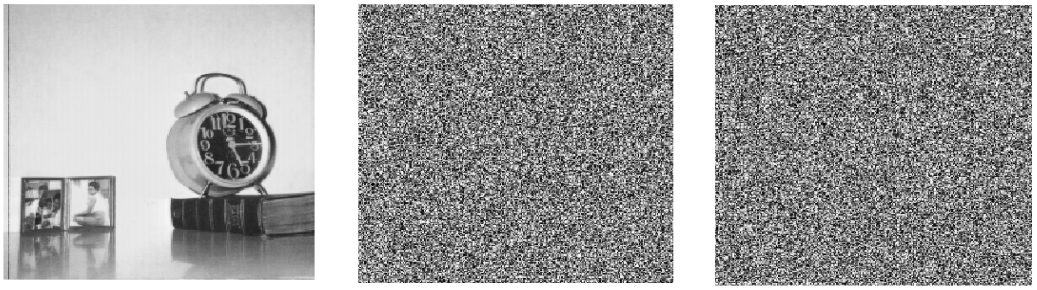

FIGURE 4.7: Key sensitivity test for Clock: (a) Plainimage (b) Encrypted image (c) Decrypted image by slightly changed key

4.6.1.3 Differential Analysis

Images encrypted with a high-performance image encryption method should appear completely different from plainimages; this is one of its key characteristics. Therefore the number of pixel change rate (NPCR) and unified average changing intensity (UACI) are used to compare an encrypted image to the original image

and the encrypted image after changing one pixel in the original image. The NPCR and UACI values can be calculated as;

$$\text{NPCR} = \frac{\sum_{i,j} M(i,j)}{w \times h} \times 100 \tag{4.5}$$

$$\text{UACI} = \frac{1}{w \times h}\left[\sum_{i,j} \frac{|Y(i,j) - Y'(i,j)|}{255}\right] \times 100 \tag{4.6}$$

Here w and h represent the encrypted image's width and height, respectively. Y and Y' are cipherimages generated from plainimages and by plane images with a difference of one pixel, respectively. If $Y = Y'$ then $M(i,j) = 0$ otherwise 1. The values of NPCR and UACI should be high and close to their ideal values to withstand differential attacks. Table 4.1 show the NPCR and UACI value performance for the proposed schemes is perfect. As a result, it will providegreat resistance to "known plaintext attacks" and "chosen plaintext attacks".

TABLE 4.1: UACI and NPCR values of encrypted images

Images	Clock	House	Girl	Chemical Plant	Tree
NPCR	99.6154	99.6292	99.6688	99.5910	99.6047
UACI	33.3538	33.2600	33.4429	33.3423	33.2128

4.6.1.4 Correlation Coefficients Analysis (CCA)

Correlation coefficients is a powerful tool for determining the rate of success of an attack on an encrypted image. Correlation is used to connect adjacent pixels that have correlation coefficients. A good encryption algorithm minimizes the correlation coefficient between pairs of encrypted neighboring pixels in the vertical, horizontal, and diagonal directions. The Correlation Coefficients (CC) is calculated as following:

$$C_{a,b} = \frac{cov(a,b)}{\sqrt{D(a)} \times \sqrt{D(b)}} \tag{4.7}$$

where a and b are the grayscale values of two pixels from the input image. $D(a)$ and $D(b)$ are the variances of a and b, respectively. $cov(a, b)$ represents the a and b covariance. The following equations are used to calculate the correlation coefficients:

$$cov(a,b) = \frac{1}{K}\sum_{j=1}^{K}(a_j - X(a))(b_j - X(b))$$

$$X(a) = \frac{1}{K}\sum_{j=1}^{K} a_j$$

$$D(a) = \frac{1}{K}\sum_{j=1}^{K}(a_j - X(a))^2$$

The test outcomes are shown in Table 3.5. The data show that the CC in the cipherimage produced by above mentioned formula's. It can be seen from Table 4.2 that correlation coefficient of cipherimage are closer to zero.

TABLE 4.2: Two adjacent pixels' correlation coefficient in a plain and cipher image.

Images	Clock		House		Girl	
Direction	Original	Cipher	Original	Cipher	Original	Cipher
Horizontal	0.9740	-0.0057	0.9529	-0.0065	0.9656	0.0053
Vertical	0.9564	0.0036	0.9782	-0.0072	0.9740	-0.0057
Diagonal	0.9389	-0.0054	0.9360	-0.0027	0.9515	-0.0018

4.6.1.5 Entropy Analysis

Data entropy was first presented by Shannon in 1948 [93]. In cryptography, the term "entropy" refers to the randomness that a system accumulates for use in techniques that uses random data. Lack of entropy can make a cryptosystem susceptible and prevent it from securely encrypting data.

This component of examination estimates the randomness in a coded image. It additionally lets us know the typical measure of data conveyed by the coded image.

If β is an encrypted image then entropy of β can be determined by the following equation.

$$E(\beta) = \sum_{j=0}^{2^N-1} P(\beta_j) \log_2 \frac{1}{P(\beta_j)} \tag{4.8}$$

$P(\beta_j)$ in equation 4.8 represents the probability of symbol β_j appearing in cipher image β. For a (256×256) grayscale image, the ideal value of the entropy is 8. Table 4.3 shows the entropy values of different images encrypted using the proposed schemes. The result shows that the resulting entropy of the proposed encryption algorithm is very close to the ideal entropy.

TABLE 4.3: Entropy analyses

Images	Clock	House	Girl	Chemical Plant	Tree
Plainimage	6.7056	6.4961	7.0524	7.0817	7.3102
Cipherimage	7.9973	7.9977	7.9975	7.9969	7.9975

4.6.1.6 Histogram Analysis

An image histogram usually refers to a histogram of pixel intensity values. Histogram of the image is a graph which shows the pixel value at each of the different intensity values found in that image. 256 different intensities are possible for a 8-bit grayscale image, so the histogram will graphically display 256 numbers showing the distribution of pixels amongst those grayscale values.

For a high-security image encryption algorithm, the encrypted image must have a histogram with a uniform distribution. Figure 4.8 shows the plain and cipherimage histograms for the image clock. It is clear that, there is no hint of a statistical attack on the cipherimage.

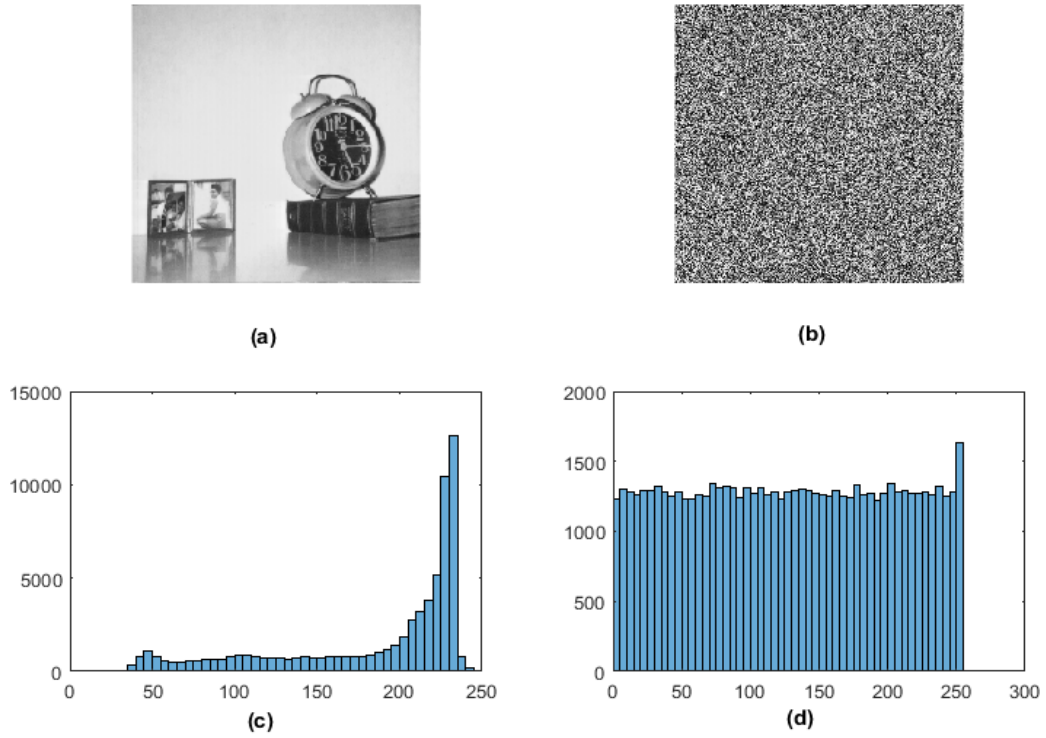

FIGURE 4.8: The following is the encryption result for the grayscale image of Clock: (a) Plainimage (b) Encrypted image (c) Histogram of plainimage (d) Histogram of encrypted image

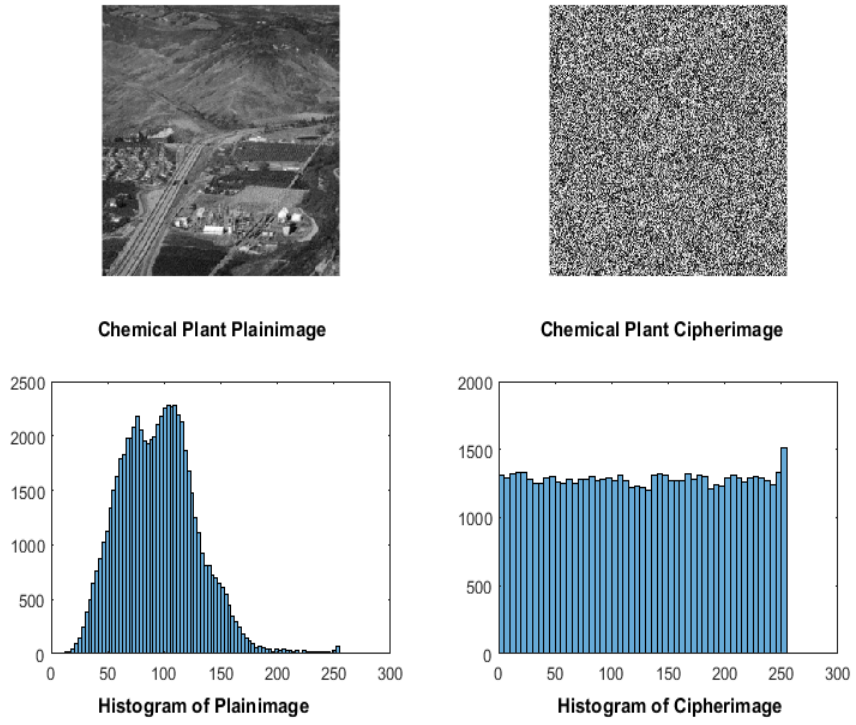

FIGURE 4.9: Histogram analysis of chemical plant plainimage and cipherimage

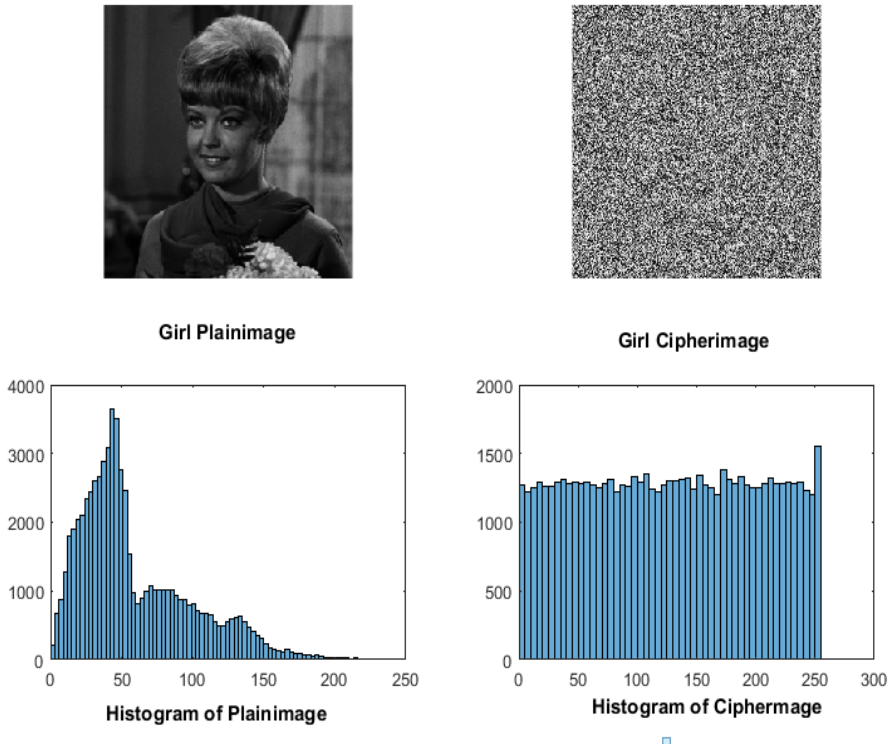

FIGURE 4.10: Histogram analysis of girl plainimage and cipherimage

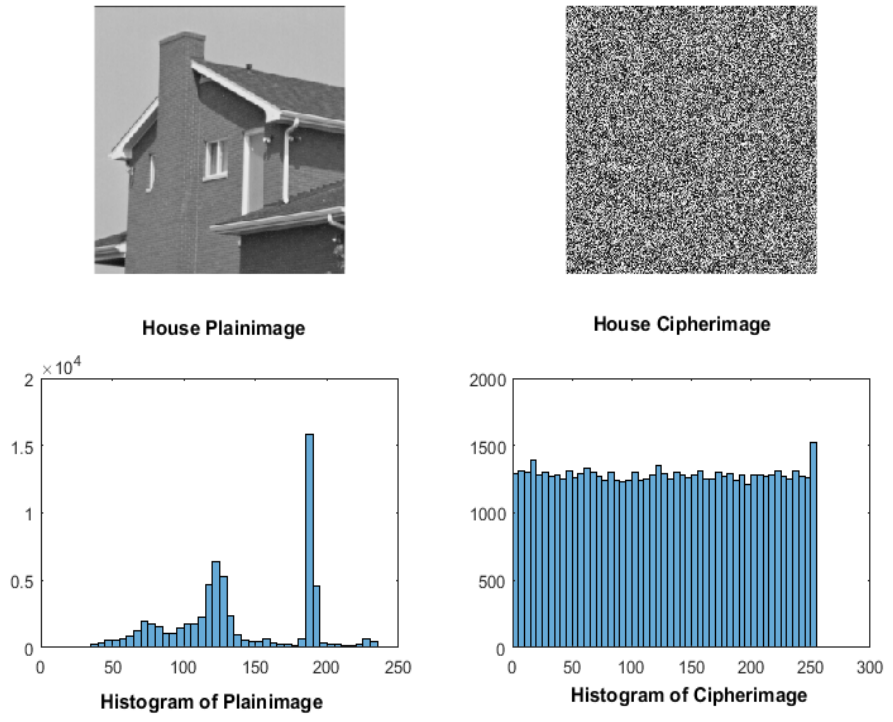

FIGURE 4.11: Histogram analysis of house plainimage and cipherimage

Figures 4.9, 4.10 and 4.11 shows histogram analysis of different plainimage and cipherimage.

Chapter 5

Conclusion

An S-box generation scheme based on cubic fractional transformation is reviewed. The method used to construct an S-box is simple and effective. The constructed S-box is highly non-linear from some old S-boxes [74, 77–86]. It has low linear probability (LP) and differential probability (DP), so the proposed S-box resists differential and linear cryptanalysis attacks. Security analysis shows that the proposed S-box is very useful in modern block cipher. The base of best knowledge it can be said that this method is the first method to create an S-box from the cubic fractional transformation (CFT). The construction procedure is completed on MATLAB and both an S-box and inverse S-box are produced.

As such, S-box ia an important component of the block encryption algorithm. They play an important role in creating confusion and diffusion. After creating the S-box, it is used in the image encryption algorithm for substitution purpose. After this, different circular shifts are performed to create more confusion. Finally, a chaotic map is used to generate a random sequence and it is used for scrambling the pixels of the image by applying the XOR operation. Entropy analysis, key sensitivity analysis, key space analysis, statistical analysis, and differential analysis are all tools used in security analysis. Security analysis shows that it is not possible to obtain the original image through cryptographic attacks. Which tells that this encryption algorithm is a secure algorithm and can be used for image encryption algorithm. This algorithm can also be used for encrypting images of different types and size.

Bibliography

[1] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *Handbook of applied cryptography*. CRC press, 2018.

[2] C. J. Monico, *Semirings and semigroup actions in public-key cryptography*. University of Notre Dame, 2002.

[3] T. Satoh and K. Araki, "On construction of signature scheme over a certain non-commutative ring," *IEICE transactions on fundamentals of electronics, communications and computer sciences*, vol. 80, no. 1, pp. 40–45, 1997.

[4] B. Acharya, S. K. Panigrahy, S. K. Patra, and G. Panda, "Image encryption using advanced hill cipher algorithm," *International Journal of Recent Trends in Engineering*, vol. 1, no. 1, pp. 663–667, 2009.

[5] W. Stallings, *Cryptography and network security, 4/E*. Pearson Education India, 2006.

[6] S. L. Garfinkel, "Public key cryptography," *Computer*, vol. 29, no. 6, pp. 101–104, 1996.

[7] C. E. Shannon, "Communication theory of secrecy systems," *The Bell system technical journal*, vol. 28, no. 4, pp. 656–715, 1949.

[8] Y. Desmedt and J.-J. Quisquater, "Public-key systems based on the difficulty of tampering (is there a difference between des and rsa?)," in *Conference on the Theory and Application of Cryptographic Techniques*, pp. 111–117, Springer, 1986.

[9] G. F. Elkabbany, H. K. Aslan, and M. N. Rasslan, "A design of a fast parallel-pipelined implementation of aes: Advanced encryption standard," *arXiv preprint arXiv:1501.01427*, 2015.

[10] W. Diffie and M. E. Hellman, "Special feature exhaustive cryptanalysis of the nbs data encryption standard," *Computer*, vol. 10, no. 6, pp. 74–84, 1977.

[11] A. Ullah, S. S. Jamal, and T. Shah, "A novel scheme for image encryption using substitution box and chaotic system," *Nonlinear Dynamics*, vol. 91, no. 1, pp. 359–370, 2018.

[12] J. Fridrich, "Image encryption based on chaotic maps," in *1997 IEEE international conference on systems, man, and cybernetics. Computational cybernetics and simulation*, vol. 2, pp. 1105–1110, IEEE, 1997.

[13] A. P. Parameshwaran and S. Wen-Zhan, "Encryption algorithms for color images: a brief review of recent trends," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 10, 2016.

[14] Y. Wang, K.-W. Wong, X. Liao, and T. Xiang, "A block cipher with dynamic s-boxes based on tent map," *Communications in Nonlinear Science and Numerical Simulation*, vol. 14, no. 7, pp. 3089–3099, 2009.

[15] G. Tang, X. Liao, and Y. Chen, "A novel method for designing s-boxes based on chaotic maps," *Chaos, Solitons & Fractals*, vol. 23, no. 2, pp. 413–419, 2005.

[16] G. Tang and X. Liao, "A method for designing dynamical s-boxes based on discretized chaotic map," *Chaos, solitons & fractals*, vol. 23, no. 5, pp. 1901–1909, 2005.

[17] X. Zhang, Z. Zhao, and J. Wang, "Chaotic image encryption based on circular substitution box and key stream buffer," *Signal Processing: Image Communication*, vol. 29, no. 8, pp. 902–913, 2014.

[18] M. Khan, T. Shah, and M. A. Gondal, "An efficient technique for the construction of substitution box with chaotic partial differential equation," *Nonlinear Dynamics*, vol. 73, no. 3, pp. 1795–1801, 2013.

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

[20] M. Khan, "A novel image encryption scheme based on multiple chaotic s-boxes," *Nonlinear Dynamics*, vol. 82, no. 1, pp. 527–533, 2015.

[21] M. Khan, T. Shah, and S. I. Batool, "A new implementation of chaotic s-boxes in captcha," *Signal, Image and Video Processing*, vol. 10, no. 2, pp. 293–300, 2016.

[22] M. Khan and T. Shah, "A novel image encryption technique based on hénon chaotic map and s 8 symmetric group," *Neural Computing and Applications*, vol. 25, no. 7, pp. 1717–1722, 2014.

[23] M. Khan and T. Shah, "A copyright protection using watermarking scheme based on nonlinear permutation and its quality metrics," *Neural Computing and Applications*, vol. 26, no. 4, pp. 845–855, 2015.

[24] M. Khan, T. Shah, H. Mahmood, M. A. Gondal, and I. Hussain, "A novel technique for the construction of strong s-boxes based on chaotic lorenz systems," *Nonlinear Dynamics*, vol. 70, no. 3, pp. 2303–2311, 2012.

[25] M. T. Tran, D. K. Bui, and A. D. Duong, "Gray s-box for advanced encryption standard," in *2008 international conference on computational intelligence and security*, vol. 1, pp. 253–258, IEEE, 2008.

[26] D. R. Stinson, *Cryptography: theory and practice*. Chapman and Hall/CRC, 2005.

[27] K. M. Ali and M. Khan, "Application based construction and optimization of substitution boxes over 2d mixed chaotic maps," *International Journal of Theoretical Physics*, vol. 58, no. 9, pp. 3091–3117, 2019.

[28] I. R. Dragomir and M. Lazăr, "Generating and testing the components of a block cipher," in *2016 8th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pp. 1–4, IEEE, 2016.

[29] M. Ahmad, N. Mittal, P. Garg, and M. M. Khan, "Efficient cryptographic substitution box design using travelling salesman problem and chaos," *Perspectives in Science*, vol. 8, pp. 465–468, 2016.

[30] E. Al Solami, M. Ahmad, C. Volos, M. N. Doja, and M. M. S. Beg, "A new hyperchaotic system-based design for efficient bijective substitution-boxes," *entropy*, vol. 20, no. 7, p. 525, 2018.

[31] C. Adams and S. Tavares, "The structured design of cryptographically good s-boxes," *Journal of cryptology*, vol. 3, no. 1, pp. 27–41, 1990.

[32] F. Özkaynak and A. B. Özer, "A method for designing strong s-boxes based on chaotic lorenz system," *Physics Letters A*, vol. 374, no. 36, pp. 3733–3738, 2010.

[33] S. Farwa, T. Shah, and L. Idrees, "A highly nonlinear s-box based on a fractional linear transformation," *SpringerPlus*, vol. 5, no. 1, pp. 1–12, 2016.

[34] X. Wang, L. Liu, and Y. Zhang, "A novel chaotic block image encryption algorithm based on dynamic random growth technique," *Optics and Lasers in Engineering*, vol. 66, pp. 10–18, 2015.

[35] V. Patidar, N. Pareek, G. Purohit, and K. Sud, "Modified substitution–diffusion image cipher using chaotic standard and logistic maps," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 10, pp. 2755–2765, 2010.

[36] M. François, T. Grosges, D. Barchiesi, and R. Erra, "A new image encryption scheme based on a chaotic function," *Signal Processing: Image Communication*, vol. 27, no. 3, pp. 249–259, 2012.

[37] 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 Applications*, vol. 76, no. 2, pp. 2933–2953, 2017.

[38] M. Khan, T. Shah, and S. I. Batool, "Construction of s-box based on chaotic boolean functions and its application in image encryption," *Neural Computing and Applications*, vol. 27, no. 3, pp. 677–685, 2016.

[39] S. Farwa, N. Muhammad, T. Shah, and S. Ahmad, "A novel image encryption based on algebraic s-box and arnold transform," *3D Research*, vol. 8, no. 3, pp. 1–14, 2017.

[40] L. Lidong, D. Jiang, X. Wang, L. Zhang, and X. Rong, "A dynamic triple-image encryption scheme based on chaos, s-box and image compressing," *IEEE Access*, vol. 8, pp. 210382–210399, 2020.

[41] T. S. Ali and R. Ali, "A new chaos based color image encryption algorithm using permutation substitution and boolean operation," *Multimedia Tools and Applications*, vol. 79, no. 27, pp. 19853–19873, 2020.

[42] T. S. Ali and R. Ali, "A novel color image encryption scheme based on a new dynamic compound chaotic map and s-box," *Multimedia Tools and Applications*, pp. 1–25, 2022.

[43] F. Lafitte and M. F. Lafitte, "Package boolfun,"

[44] F. Lafitte, "The boolfun package: Cryptographic properties of boolean functions," 2012.

[45] R. C. Team *et al.*, "R: A language and environment for statistical computing," 2013.

[46] W. Stein, "Sage mathematics software," *http://www. sagemath. org/*, 2007.

[47] J. A. Alvarez-Cubero and P. J. Zufiria, "A c++ class for analysing vector boolean functions from a cryptographic perspective," in *2010 International Conference on Security and Cryptography (SECRYPT)*, pp. 1–9, IEEE, 2010.

[48] S. Picek, L. Batina, D. Jakobović, B. Ege, and M. Golub, "S-box, set, match: a toolbox for s-box analysis," in *IFIP International Workshop on Information Security Theory and Practice*, pp. 140–149, Springer, 2014.

[49] A. H. Zahid, M. J. Arshad, and M. Ahmad, "A novel construction of efficient substitution-boxes using cubic fractional transformation," *Entropy*, vol. 21, no. 3, p. 245, 2019.

[50] A. Mousa and A. Hamad, "Evaluation of the rc4 algorithm for data encryption.," *Int. J. Comput. Sci. Appl.*, vol. 3, no. 2, pp. 44–56, 2006.

[51] R. J. McEliece, "A public-key cryptosystem based on algebraic," *Coding Thv*, vol. 4244, pp. 114–116, 1978.

[52] R. Singh and S. Kumar, "Elgamals algorithm in cryptography," *International Journal of Scientific & Engineering Research*, vol. 3, no. 12, pp. 1–4, 2012.

[53] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[54] D. Rosenthal, D. Rosenthal, and P. Rosenthal, "The fundamental theorem of arithmetic," in *A Readable Introduction to Real Mathematics*, pp. 31–34, Springer, 2014.

[55] C. J. Benvenuto, "Galois field in cryptography," *University of Washington*, vol. 1, no. 1, pp. 1–11, 2012.

[56] N. P. Smart *et al.*, *Cryptography: an introduction*, vol. 3. McGraw-Hill New York, 2003.

[57] N. Tokareva, *Bent functions: results and applications to cryptography*. Academic Press, 2015.

[58] J. A. Clark, J. L. Jacob, S. Maitra, and P. Stanica, "Almost boolean functions: the design of boolean functions by spectral inversion," in *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.*, vol. 3, pp. 2173–2180, IEEE, 2003.

[59] J. A. Armario, "Boolean functions and permanents of sylvester hadamard matrices," *Mathematics*, vol. 9, no. 2, p. 177, 2021.

[60] A. Hedayat and W. D. Wallis, "Hadamard matrices and their applications," *The Annals of Statistics*, pp. 1184–1238, 1978.

[61] V. Rijmen and J. Daemen, "Advanced encryption standard," *Proceedings of Federal Information Processing Standards Publications, National Institute of Standards and Technology*, pp. 19–22, 2001.

[62] A. Al-Wattar, "Dynamic key-depending s-boxes inspired by biological dna," *Int. J. Electr. Comput. Sci. IJECS*, vol. 15, no. 04, pp. 48–53, 2015.

[63] O. S. Rothaus, "On bent functions," *Journal of Combinatorial Theory, Series A*, vol. 20, no. 3, pp. 300–305, 1976.

[64] A. Webster and S. E. Tavares, "On the design of s-boxes," in *Conference on the theory and application of cryptographic techniques*, pp. 523–534, Springer, 1985.

[65] I. VERGİLİ and M. D. Yücel, "Avalanche and bit independence properties for the ensembles of randomly chosen n\times n s-boxes," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 9, no. 2, pp. 137–146, 2001.

[66] P. Sarkar and S. Maitra, "Construction of nonlinear boolean functions with important cryptographic properties," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 485–506, Springer, 2000.

[67] T. Siegenthaler, "Correlation-immunity of nonlinear combining functions for cryptographic applications (corresp.)," *IEEE Transactions on Information theory*, vol. 30, no. 5, pp. 776–780, 1984.

[68] W. Meier, E. Pasalic, and C. Carlet, "Algebraic attacks and decomposition of boolean functions," in *International conference on the theory and applications of cryptographic techniques*, pp. 474–491, Springer, 2004.

[69] N. Li, L. Qu, W.-F. Qi, G. Feng, C. Li, and D. Xie, "On the construction of boolean functions with optimal algebraic immunity," *IEEE Transactions on Information Theory*, vol. 54, no. 3, pp. 1330–1334, 2008.

[70] Y. Zhou, "On the distribution of auto-correlation value of balanced boolean functions," *Advances in Mathematics of Communications*, vol. 7, no. 3, p. 335, 2013.

[71] Q. Lu, C. Zhu, and G. Wang, "A novel s-box design algorithm based on a new compound chaotic system," *Entropy*, vol. 21, no. 10, p. 1004, 2019.

[72] M. Asim and V. Jeoti, "Efficient and simple method for designing chaotic s-boxes," *ETRI journal*, vol. 30, no. 1, pp. 170–172, 2008.

[73] S. S. Jamal, A. Anees, M. Ahmad, M. F. Khan, and I. Hussain, "Construction of cryptographic s-boxes based on mobius transformation and chaotic tent-sine system," *IEEE Access*, vol. 7, pp. 173273–173285, 2019.

[74] I. Hussain, T. Shah, M. A. Gondal, M. Khan, and W. A. Khan, "Construction of new s-box using a linear fractional transformation," *World Appl. Sci. J*, vol. 14, no. 12, pp. 1779–1785, 2011.

[75] A. Altaleb, M. S. Saeed, I. Hussain, and M. Aslam, "An algorithm for the construction of substitution box for block ciphers based on projective general linear group," *AIP Advances*, vol. 7, no. 3, p. 035116, 2017.

[76] M. Sarfraz, I. Hussain, and F. Ali, "Construction of s-box based on mobius transformation and increasing its confusion creating ability through invertible function," *International Journal of Computer Science and Information Security*, vol. 14, no. 2, p. 187, 2016.

[77] G. Vaicekauskas, K. Kazlauskas, and R. Smaliukas, "A novel method to design s-boxes based on key-depdendent permutation schemes and its quality analysis," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 4, pp. 93–99, 2016.

[78] E. M. Mahmoud, A. Abd, T. A. E. El Hafez, and T. A. El Hafez, "Dynamic aes-128 with key-dependent s-box," 2013.

[79] A. H. Alkhaldi, I. Hussain, and M. A. Gondal, "A novel design for the construction of safe s-boxes based on tderc sequence," *Alexandria Engineering Journal*, vol. 54, no. 1, pp. 65–69, 2015.

[80] G. Chen, "A novel heuristic method for obtaining s-boxes," *Chaos, Solitons & Fractals*, vol. 36, no. 4, pp. 1028–1036, 2008.

[81] A. Belazi, R. Rhouma, and S. Belghith, "A novel approach to construct s-box based on rossler system," in *2015 international wireless communications and mobile computing conference (IWCMC)*, pp. 611–615, IEEE, 2015.

[82] S. Mahmood, S. Farwa, M. Rafiq, S. M. J. Riaz, T. Shah, and S. S. Jamal, "To study the effect of the generating polynomial on the quality of nonlinear components in block ciphers," *Security and Communication Networks*, vol. 2018, 2018.

[83] N. Siddiqui, U. Afsar, T. Shah, and A. Qureshi, "A novel construction of s16 aes s-boxes," *Int. J. Comput. Sci. Inf. Secur*, vol. 14, pp. 811–818, 2016.

[84] I. Hussain, T. Shah, M. A. Gondal, and Y. Wang, "Analyses of skipjack s-box," *World Appl. Sci. J*, vol. 13, no. 11, pp. 2385–2388, 2011.

[85] I. Hussain, T. Shah, M. A. Gondal, W. A. Khan, and H. Mahmood, "A group theoretic approach to construct cryptographically strong substitution boxes," *Neural Computing and Applications*, vol. 23, no. 1, pp. 97–104, 2013.

[86] I. Hussain, T. Shah, H. Mahmood, M. A. Gondal, and U. Y. Bhatti, "Some analysis of s-box based on residue of prime number," *Proc Pak Acad Sci*, vol. 48, no. 2, pp. 111–115, 2011.

[87] H. Pan, Y. Lei, and C. Jian, "Research on digital image encryption algorithm based on double logistic chaotic map," *EURASIP Journal on Image and Video Processing*, vol. 2018, no. 1, pp. 1–10, 2018.

[88] R. Stoop and P. Meier, "Evaluation of lyapunov exponents and scaling functions from time series," *JOSA B*, vol. 5, no. 5, pp. 1037–1045, 1988.

[89] H. A. Abdullah and H. N. Abdullah, "Secure image transmission based on a proposed chaotic maps," in *Multimedia Security Using Chaotic Maps: Principles and Methodologies*, pp. 81–109, Springer, 2020.

[90] M. Murillo-Escobar, C. Cruz-Hernández, L. Cardoza-Avendaño, and R. Méndez-Ramírez, "A novel pseudorandom number generator based on pseudorandomly enhanced logistic map," *Nonlinear Dynamics*, vol. 87, no. 1, pp. 407–425, 2017.

[91] M. I. Mihailescu and S. L. Nita, "Cryptography and cryptanalysis in matlab," *Apress, Berkeley, CA*, 2021.

[92] G. Alvarez and S. Li, "Some basic cryptographic requirements for chaos-based cryptosystems," *International journal of bifurcation and chaos*, vol. 16, no. 08, pp. 2129–2151, 2006.

[93] C. Shannon, "Une théorie mathématique des télécommunications," *Bell. Syst. Techn. J.*, vol. 27, no. 3, pp. 379–423, 1948.