MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

UNIVERSITY OF MOHEMED SEDDIK BEN YAHIA- JIJEL

Faculty of Exact Science and Computer Science

Department of Computer Science

Master thesis for obtaining a master's degree in computer science

Option: Networks and Security

Study and Analysis of a Hybrid Image Encryption Algorithm Using Chaos and Generating Functions

Directed By:

Mss .Noura Louzzani

Presented By :

Nadjib Hachehouche

ACADEMIC YEAR: 2018/ 2019

DEMOCRATIC AND POPULAR REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

UNIVERSITY OF MOHEMED SEDDIK BEN YAHIA- JIJEL

Faculty of Exact Science and Computer Science

Department of Computer Science

Master thesis for obtaining a master's degree in computer science

Option: Networks and Security

Study and Analysis of a Hybrid Image Encryption Algorithm Using Chaos and Generating Functions

Directed By:

Mss .Noura Louzzani

Presented By :

Nadjib Hachehouche

ACADEMIC YEAR:2018/2019

ABSTRACT

Scientists believed that chaos can be used in encryption to protect data because of its properties such as determinism and extreme sensitivity to initial conditions

This work is an application of chaos in digital image encryption: a new encryption algorithm for digital images based on a combination of a modified generating function of Lucas balancing numbers and the logistics map, the obtained results showed that the proposed algorithm is characterized by high security, high performance and high speed

Keywords

Chaos, dynamic systems, logistic map, generating functions, digital images, transmission, cryptography, security.

Résumé

Les scientifiques croyaient que le chaos pouvait être utilisé dans le cryptage pour protéger les données en raison de ses propriétés telles que le déterminisme et la sensibilité extrême aux conditions initiales.

Ce travail est une application du chaos dans le cryptage des images numériques: un nouvel algorithme de cryptage des images numériques basé sur une combinaison d'une fonction génératrice modifiée de suit des nombres de Lucas et la fonction logistique, les résultats obtenus ont montré que l'algorithme proposé est caractérisé par la haute niveau de sécurité, la haute performance et la grande vitesse.

Mots clés

Systèmes dynamiques, fonction logistique, fonctions génératrices, images numériques, transmission, cryptographie, sécurité.

الملخص

لطالما أمن العلماء بإمكانية إستخدام الفوضى في التشفير لحماية البيانات بسبب خصائصها الفريدة مثل الحتمية والحساسية الشديدة للظروف الأولية.

هذا العمل عبارة عن تطبيق للفوضى في تشفير الصور الرقمية: خوارزمية تشفير جديدة للصور الرقمية تعتمد على دمج بين دالة التوليد لمتتالية لوكاس المعدلة والخريطة اللوجستية ؛ أظهرت النتائج التي تم الحصول عليها أن الخوارزمية المقترحة عالية الأداء و الأمان و السرعة.

الكلمات المفتاحية:

الضوضاء, الأنظمة الديناميكية. الخريطة اللوجيستية, الدوال المولدة, الصور الرقمية, إرسال البيانات, التشفير, الحماية.

ACKNOWLEDGMENT

"الْحَمْدُ لِلَّهِ الَّذِي هَدَانَا لِهَٰذَا ومَاكُنَّا لِنَهْتَدِيَ لَوْلاَ أَنْ هَدَانَا اللَّهُ " الأعراف 43

This work not have been possible without the instructions, guidance, motivation and continuous encouragement of my supervisor: Mss Noura Louzzani, special thanks to her.

Also I would like to express my special appreciation to Ali Bousaioud for his invaluable support and advices.

Also I would like to thank the members of the jury for accepting to judge my work.

Finally, I would like to extend my appreciation to all my family members, my friends and classmates and all those who, in one way or another, offered personal support and encouragement.

Nadjib.H

DEDICATION

This master thesis is dedicated to the most important persons in my life: my beloved parents: Messaoud and Asia who have been my source of inspiration and gave me strength when I thought of giving up, who continually provide their moral, spiritual, emotional, and financial support.

Also, I would like to dedicate this work to my sister Marwa and my brothers Mansor, Aymen and Rida.

Nadjib.H

Table of Content

List of Figures

List of Tables

List of Abbreviations

- **2D:** 2 Dimensional
- **3D** :3 Dimensional
- **AES:** Advanced Encryption Standard
- **CMY:** Cyan Magenta Yellow
- **CMYK:** Cyan Magenta Yellow blacK
- **DNA:** Deoxyribonucleic Acid
- **DES:** Data Encryption Standard
- **JPEG:** Joint Photographic Experts Group
- **JFIF:** JPEG File Interchange Format
- **LMGF:** LogisticMGF
- **MGF:** Modified Generating Function
- **MGFL:** MGFLogistic
- **MVC:** Model View Controller.
- **MSE:** Means Square Error.
- **LZW:** Lempel-Ziv-Welch.
- **PSNR:** Peak Signal to Noise Ratio
- **PNG:** Portable Network Graphics
- **PWLCM :** Piece-Wise Linear Chaotic Map
- **RGB:** Red Green Blue.
- **RSA:** Rivest Shamir Adelman
- **UACI:** Unified Average Changing Intensity.

GENERAL INTRODUCTION

General introduction

Nowadays, there is a rapid growth of using visual content like images and videos, in social media networks, such as Skype, Facebook, Instagram and YouTube, visual search engines like Google images, Yahoo image search and Pinterest, cloud storage services such as Google Drive. This growth is expected to continue increasing in the next few years, may be faster, because of the coming technologies such as the Virtual Reality and 5G.

This content is transmitted over various types of mediums: wired networks such as telephone networks or wireless networks such Satellite, Wi-Fi. Etc. This medium is not only a subject of different types of noise like Gaussian noise, but also should supposed to be insecure according to Shannon's theory of communication secrecy, this require applying some mechanisms in order to transmit this content in a secure way, and encryption is probably most used mechanism.

In real world, a variety of efficient encryption schemes are available, such as: AES (Advanced Encryption Scheme), RSA (Rivest-Shamir-Adelman) and ElGamel, these algorithms are designed to encrypt textual information at the first place, and unfortunately they do not work with the same efficiency with high correlated data like images or videos, and this make securing this type of data a rising problem, and one of the non-traditional solutions of this problem is what is called "chaos-based image encryption algorithms", as the name sounds: they are cryptographic algorithms that exploit the phenomena of chaos and its characteristics such as the sensitivity to the initial conditions, ergodicity to encrypt images.

Many algorithms have been proposed by the researchers in the last few years, some of them exploit only chaotic functions in the encryption/decryption process, while others combine chaotic functions with another non-chaotic technic such as the traditional ciphers and DNA-Encoding.

In this work, we propose a solution to the previously stated problem: a hybrid image encryption algorithm that use chaotic systems and generating functions, to be more specific, it is an encryption scheme specialized to encrypt images, based on a proposed chaotic function, which is basically a combination of the well-known logistic map and a modified generating function of Lucas balancing numbers.

Outline of the thesis:

This thesis is organized as follow:

The first chapter is an introduction to our main subject: "chaos-based image encryption", we present an overview of the three related topics which are: Digital images, cryptography and chaos.

In the first topic, we shade light on basic concepts of digital image, the most common color models, and the widely used image file formats. in the second one, we present the basics of the modern cryptography and the three main types of it: symmetric ciphers, asymmetric ciphers and cryptographic hash functions. While in the chaos topic we present a short introduction to dynamic systems and chaos. At the end of this chapter we present the chaos-based image cryptography which is the relation between the previously mentioned topics.

The second chapter is the state of the art of the chaos-based image encryption schemes, in this chapter, we first present the metrics used to evaluate the security and the performance of image encryption schemes, then, we present our classification of these special type of ciphers based on the exploited chaotic system into two main classes "pure chaotic base" and "hybrid base", the first class contains ciphers that uses only chaotic systems in the encryption/decryption process, this class itself is sub-divided into other subclasses. While the second one contains ciphers that are based on a mixing of chaotic and non-chaotic systems, in this study we will refer to DNA-encoding and modern cryptosystems.

The last chapter focus on presenting our findings, first we prove the chaotic behavior in the generating function of Lucas balancing numbers, then we provide three enhanced chaotic functions, the first one is a modified generating function of Lucas balancing numbers, let's name it MGF for short, the second and the third ones are the result of combining the MGF with the well-known chaotic function: the logistic map we will refer to them as: LMGF (Logistic-MGF) and MGFL (MGF-Logistic). After that we present the conception and the implementation of our cryptosystems which are based on the new proposed chaotic functions, finally we prove the high security and high performance and his capability to withstand to all known attacks of the proposed cryptosystem using our testing platform.

At the end of this Master thesis, we will give a general conclusion, which contain a summary of this work, and an outline to future works.

CHAPTER 01

GENERALITIES: DIGITAL IMAGES, CRYPTOGRAPHY AND CHAOS

Chapter 01. Generalities: Digital Images, Cryptography and Chaos

1.1 Introduction

In this chapter we will give a brief overview in the widely used media nowadays: digital images, then we will present the most important basics of modern cryptography and its types and algorithms. After that, we will give a short overview and basic concepts of dynamic systems and chaos, and finally the shared properties between chaos and cryptography.

1.2 Digital images

Digital images have become one of the most popular media types in the internet and specifically in social media networks, according to last the statistics, there is more than 1000 shared image on Instagram per second, and visual content is 40 % more likely to be shared than any content in social media. Moreover, images are used extensively in various fields such as economics, defense, and education, and for daily life.

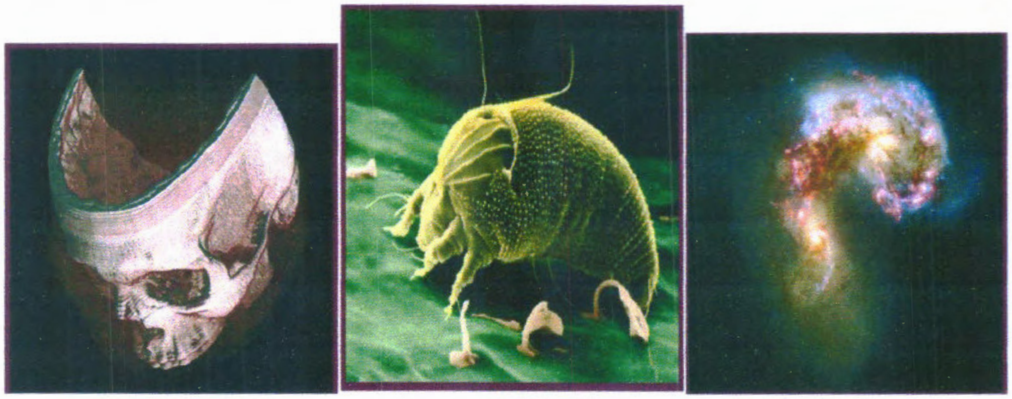


Figure 1-1 medical image, biological image and astronomical image

1.2.1 Definitions and terms

1.2.1.1 Digital image

A digital image can be defined as a “visual information that is represented in digital form.”, or it’s a “numeric representation of a 2-Dimentional image” [1] [2]

1.2.1.2 Digital image editing

Digital image editing is the manipulation of digital images using an existing software application such as” Adobe Photoshop” or “Corel Paint”. [2]

1.2.1.3 Digital image processing

Digital image processing is the conception, design, development and enhancement of digital imaging programs [3]

1.2.1.4 Pixel

Pixel is an abbreviation of "PICture ELement", it represents the smallest unit of an image. Its values are always binary words of length k so that a pixel can represent 2^k different values. The value of k is often called "depth of the image". As an example, a typical color image with three components RGB, the entire pixel is encoded in 24 bits, which is the depth of the image, as a result, this image can represent 2^{24} different color which is equal to16,777,216 different color. [1] [2]

1.2.1.5 Definition

The size of an image is the number of pixels of an image, calculated by multiplying the number of pixels of the column by the number of pixels of the row. [1]

For example, the definition of an image with 800 pixels in width and 600 pixels of height its definition is noted as 800x600.

1.2.1.6 Resolution

The resolution of an image specifies the spatial dimensions of the image in the real world and is given as the number of image elements per measurement.

for example, dots per inch (dpi) or lines per inch (lpi) for print production, or in pixels per kilometer for satellite images.

1.2.2 Color models

A color model is an abstract mathematical system for representing colors. Since color is a three dimensional entity, a color model defines three primary colors (corresponding to three dimensions or axes) from which all possible colors are derived by mixing together various amounts of these primaries. The range of colors covered by a color model is referred to as either the gamut or the color space.

Color models can be classified as either:

- an additive color models: assume that light is used to generate colors for display, as a result, the color black represents a complete lack of the primary colors while the color white corresponds to maximal and equal amounts of each of the primaries, additive models are commonly used by computer monitors and LCD projectors.
- subtractive color models: assume that pigment will be used to create colors such that a complete absence of any pigment corresponds to the color white while combining the three primaries in maximal and equal amounts yields the color black. Subtractive color models assume that when white light is projected onto pigment, the pigment will absorb power from certain wavelengths and reflect power at other wavelengths. This absorption is described as *subtraction*. In terms of electronic systems, images rendered with ink-based printers are most naturally described using a subtractive color model [2].

1.2.2.1 RGB Color Model

The RGB model is an additive color model which uses the primary colors of light: red, green, and blue as the primary colors such that any color can be obtained by combining different amounts of these three primaries.

A color within the RGB color space is defined by three numeric values (a tuple) that specify the amount of red, green, and blue that comprise the specified color.

The origin lies at <0, 0, 0> (black) while the opposite corner lies at <1, 1, 1> and corresponds to the color white. And the line connecting pure black and pure white within the RGB color space is known as the gray scale since any point lying on that line is a shade of gray [2]

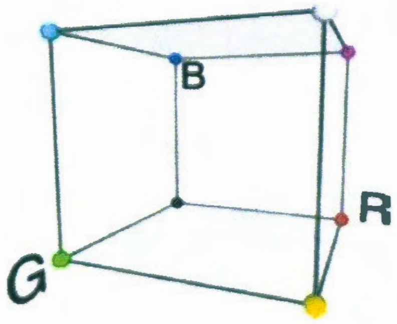


Figure 1-2 RGB color model

1.2.2.2 CMY Color Model

The CMY model is a subtractive color model that uses the primary colors of pigment: cyan, magenta and yellow as primary colors. The CMY color space can, like RGB, be viewed as a cube where a CMY color is designated by a normalized tuple of values.

The CMY color model is inversely related to the RGB color model.

$$\begin{cases} C = 1 - R \\ M = 1 - G \\ Y = 1 - B \end{cases} \quad \dots\dots\dots\dots\dots\dots\dots (1.1)$$

The CMYK color model is an extension of the CMY. The CMYK model augments the CMY by including a fourth primary component that corresponds to the amount of black in a color, this model is generally used in printers in order to reduce the used ink, as a result of this color model, it's not necessary to use the three primary colors in order to produce the black color. [2]

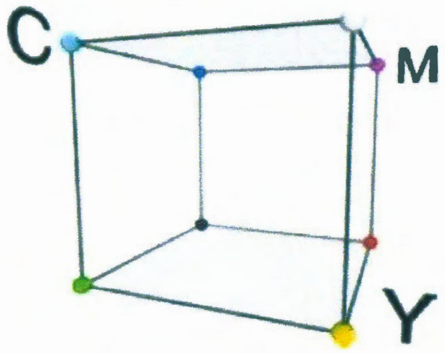


Figure 1-3 CMY color model

1.2.2.3 HSB Color Model

The HSB color model decomposes color according to how it is *perceived* rather than, as is the case with RGB, with how it is *physically sensed*. Since HSB seeks to mirror the perception of color, the three dimensions of the HSB color space are aligned with a more intuitive understanding of color than either RGB or CMY. A point within the HSB gamut (or color space) is defined by hue (the chromaticity or pure color), saturation and brightness (the intensity of the color). The HSB color space can be visualized as a cylinder.

Manipulating colors in HSB space tends to be more intuitive to artists than working with either the RGB or CMY models. In order to darken a color in HSB only the brightness value is reduced while darkening a color in RGB involves scaling each of the three primary components proportionally. The complement of a color in HSB is easily found by crossing to the opposite side of the HSB cylinder, computationally achieved by shifting the hue band by 180°.

The HSB color model is often used in image processing since it provides a way to separate color information (HS) from intensity (B). [2]

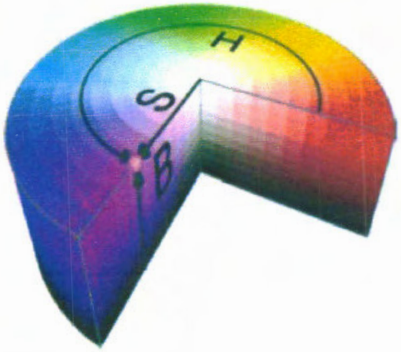


Figure 1-4 HSB color model

1.2.2.4 YUV Color Model

Another color model often used in digital image and video processing is UYV color model, which describes color perception and consists of one luminance component (Y) and two chrominance components (U and V). The YUV image representation can be created from RGB presentation using the following approximate mathematical description:

$$\begin{cases} Y = 0.6R + 0.3G + 0.1B \\ U = B - Y \\ V = R - Y \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (1.1)$$

The YUV color model consists of a luminance component Y, which captures the brightness of the pixel, and two chrominance components, U and V, that capture color information. [4]

1.2.3 Types of digital images

1.2.3.1 Grayscale images

The image data in a grayscale image consist of a single channel that represents the brightness of the image. In most cases, only positive values make sense, so typically whole integers in the range 0, ..., 2^{k-1} are used. Where 0 represents the minimum brightness (black) and 2^{k-1} the maximum brightness (white).

A typical grayscale image uses $k = 8$ bits per pixel, but for some domains such as professional photography, medicine and astronomy, 8 bits per pixel is not sufficient. Image depths of 12, 14, and even 16 bits are often used.

1.2.3.2 Binary image

Binary images are a special type of intensity image where pixels can only take on one of two values, black or white. typically encoded using a single bit (0/1) per pixel. Binary images are often used for representing line graphics, archiving documents, encoding fax transmissions. etc.

1.2.3.3 Color images

Most color images are based on the primary colors red, green, and blue (RGB), typically making use of 8 bits for each color component which means that each pixel requires $3 \times 8 = 24$ bits to encode all three components, and the range of each individual color component is [0, 255]. While color images with 30, 36, and 42 bits per pixel are commonly used in professional applications.

1.2.3.4 Indexed images

A very special class of color image. In an indexed image, the pixel values are only indices (with a maximum of 8 bits) onto a specific table of selected full color values.

1.2.3.5 Special images

Two common examples of special images are:

- Images with negative values: arise during image processing steps, such as filtering for edge detection.
- Images with floating-point values: often found in medical, biological, or astronomical applications, where extended numerical range and precision are required. [3]

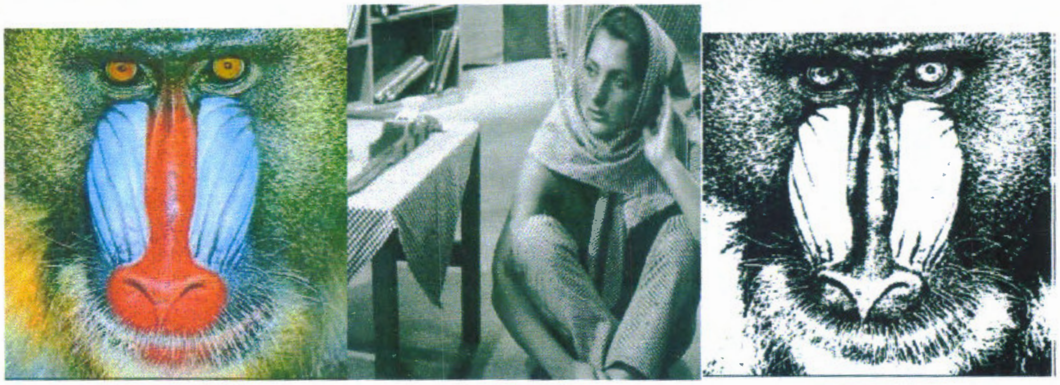


Figure 1-5 Color image, Grayscale image and binary image

1.2.4 Raster versus Vector images

There are two types of digital images:

- **raster images or bitmap images:** are images that contain pixel values arranged in a regular matrix using discrete coordinates.
- **vector graphics**: represent geometric objects using continuous coordinates, which are only rasterized once they need to be displayed on a physical device such as a monitor or printer.

A number of standardized file formats exist for vector images: CGM (Computer Graphics Metafile) and SVG (Scalable Vector Graphics), as well as proprietary formats such as DXF (Drawing Exchange Format from Autodesk), AI (Adobe Illustrator), PICT (QuickDraw Graphics Metafile from Apple). [3]

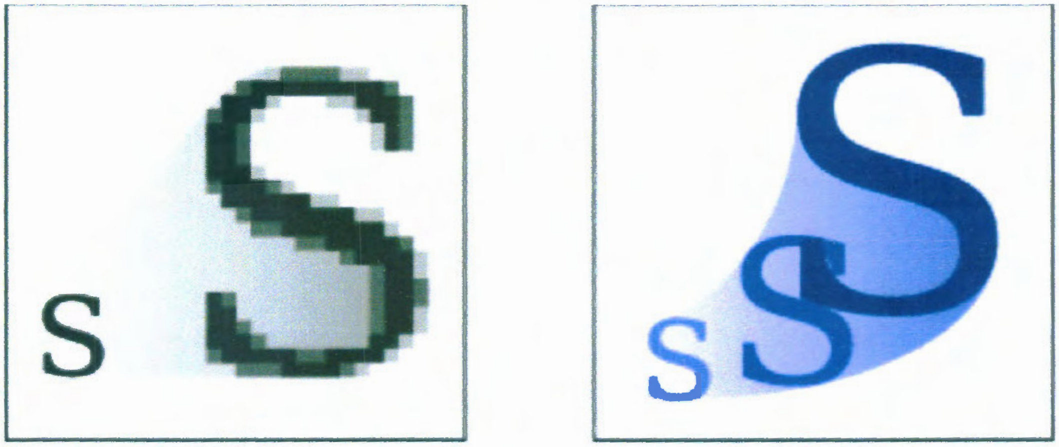


Figure 1-6 Raster image Versus Vector image

1.2.5 Image File Format

1.2.5.1 Tagged Image File Format (TIFF)

TIFF is a widely used and flexible file format designed to meet the professional needs of diverse fields such as archiving documents, scientific applications, digital photography. It was originally developed by Aldus and later extended by Microsoft and currently Adobe. It supports a range of grayscale, indexed, and true color images, and also special image types with large-depth integer and floating-point elements [3].

The TIFF specification provides a range of different compression methods (LZW, ZIP, and JPEG) and color spaces, so that it is possible, for example, to store a number of variations of an image in different sizes and representations together in a single TIFF file [3].

1.2.5.2 Graphics Interchange Format (GIF)

The Graphics Interchange Format (GIF) was originally designed by CompuServe in 1986. It has since grown into one of the most widely used formats for representing images on the Web. This popularity is largely due to its early support for indexed color at multiple bit depths, LZW compression, and ability to encode simple animations.

GIF is essentially an indexed image file format designed for color and grayscale images with a maximum depth of 8 bits and consequently it does not support true color images. It offers efficient support for encoding palettes containing from 2 to 256 colors, one of which can be marked for transparency. [3]

1.2.5.3 Portable Network Graphics (PNG)

PNG (pronounced “ping”) was originally developed as a replacement for the GIF file format when licensing issues arose because of its use of LZW compression. It was designed as a universal image format especially for use on the Internet, and, as such, PNG supports three different types of images:

- true color images (with up to 3 × 16 bits/pixel),
- grayscale images (with up to 16 bits/pixel),
- indexed color images (with up to 256 colors).

The format only supports a single image per file. The format supports lossless compression by means of a variation of PKZIP (Phil Katz's ZIP). No lossy compression is available. Ultimately, the PNG format meets or exceeds the capabilities of the GIF format in every way except GIF's ability to include multiple images in a single file to create simple animations. [3]

1.2.5.4 JPEG

Despite common usage, JPEG is not a file format, it is only a lossy method of compressing image data particularly for images produced by digital cameras, It standard defines a compression method for grayscale and color images. this format was developed by the Joint Photographic Experts Group (JPEG). [5] [3]

There are many specifications that specifies the file format:

JFIF: "JPEG File Interchange Format" (JFIF) file, JFIF specifies a file format based on the JPEG standard by defining the remaining necessary elements of a file format, it supports images with up to 65535*65535 pixels.

JPEG-2000, which is specified by an ISO-ITU standard to overcome some of the better-known weaknesses of the traditional JPEG codec. These improvements enable it to achieve significantly higher compression ratios than JPEG (up to 0.25 bits per pixel on RGB color images). Despite these advantages, JPEG-2000 is supported by only a few image-processing applications and Web browsers. [3]

1.2.5.5 Windows Bitmap (BMP

The Windows Bitmap (BMP) format is a simple, and under Windows widely used, file format supporting grayscale, indexed, and true color images. It also supports binary images, but not in an efficient manner, since each pixel is stored using an entire byte. Optionally, the format supports simple lossless, run-length-based compression.

1.2.5.6 Portable Bitmap Format (PBM)

The Portable Bitmap Format (PBM) family consists of a series of very simple file formats that are exceptional in that they can be optionally saved in a human-readable text format that can be easily read in a program or simply edited using a text editor. This format makes it easy to create and store image data without any explicit imaging API. [3]

PBM is widely used under Unix and supports the following formats: PBM (*portable bitmap*) for binary *bitmaps*, PGM (*portable graymap*) for grayscale images, and PNM (*portable any map*) for color images. [3]

1.3 Cryptography

Originally, the word cryptography is coming from the two words "cryptos" which means "hidden" and "graphien" which means code, which means the study of secret codes, it used to ensure confidentiality of information, now cryptography has a wider sense: being defined as the science of information protection against unauthorized parties by preventing unauthorized alteration of use .in other words, it's used to ensure the main three objectives of the security: Confidentiality, Integrity and Authentication, CIA for short. [6] [7]

There are three main types of cryptographic systems: Symmetric ciphers, asymmetric ciphers, and cryptographic data integrity algorithms, we will take all these types in more details in later in this chapter.

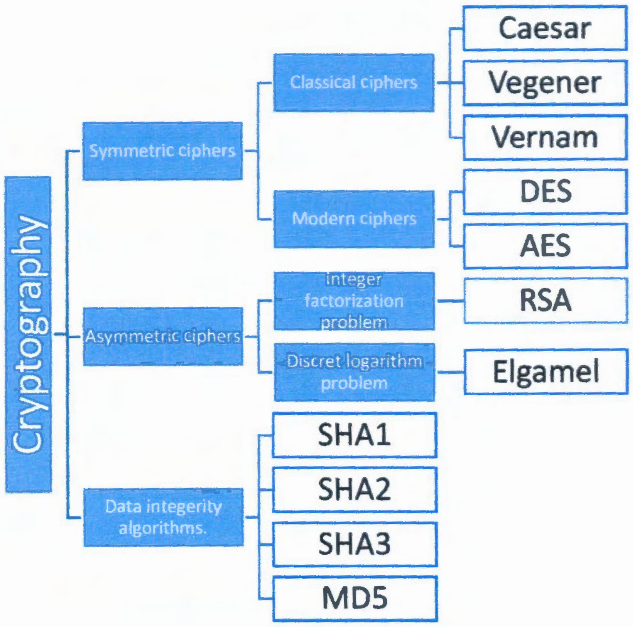


Figure 1-7 Main types of cryptographic algorithms

1.3.1 Basics of cryptography

1.3.1.1 The Shannon's theory of Secure Communication

The purpose of encryption is to ensure communication secrecy. We assume that we want to communicate, which means to transmit information through a channel. The channel is not assumed to be secure. [6]

1.3.1.2 Kerckhoffs's Law

The Kerckhoffs's law state that 'a cryptosystem should be secure even if everything about the system, except the key, is public knowledge'. So we should assume that the cryptosystem is designed by a third party, from a third company, in a third country, as a result, the design of the cipher is not a secret since it is designed and implemented in n point. [6]

1.3.1.3 The Moore's Law

named after Gordon Moor, which says that the speed of CPUs doubles every 18 months. In other words, after 18 months, CPUs will be able to perform twice as much instruction as the current

CPUs So, if a cryptosystem is designed for long-term secrecy, the secret key must be long enough to resist exhaustive search using future technologies. [6]

1.3.1.4 The Murphy Law

The murphy law states that "whatever can go wrong, will go wrong", as an application of this low in cryptography we can say that If there is a single security hole, make sure that someone will eventually find it. By extension we should keep in mind that security does not add up: systems are as secure as their weakest part. [6]

1.3.2 Symmetric ciphers

The first class is Symmetric ciphers (illustrated in Figure 1-8): Sometimes called single key or conventional cryptographic system, it was the only type of encryption in use prior to the development of public key encryption in the 1970s. It remains by far the most widely used of the two types of encryption.

In this system both of the entities (the sender and the receiver) uses the same key in the encryption/decryption process, there exist several symmetric cryptographic systems are extensively used in the last years: such as the Standardized DES, and its alternative AES.

Depending on how the plain text is encrypted, there are two different groups of symmetric ciphers, both are illustrated in Figure 1-9,

— "block ciphers", which processes one block of elements at a time producing an output block for each input block, a block cipher used in practice is a repetition of rounds (iterations), a short sequence of operations that is weak on its own but strong in number there are two main techniques to construct a round: substitution–permutation networks (as in AES) and Feistel schemes (as in DES).

— "stream ciphers", which processes one element continuously producing one element at the output at a time, as it goes along. [7] [8]

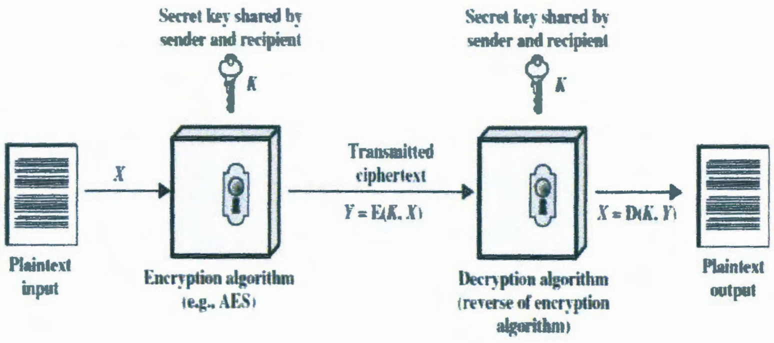

Secret key shared by sender and recipient
K
Secret key shared by sender and recipient
K
X
Transmitted ciphertext
Y = E(K, X)
X = D(K, Y)
Plaintext input
Encryption algorithm (e.g., AES)
Decryption algorithm (reverse of encryption algorithm)
Plaintext output

Figure 1-8 Symmetric cipher model

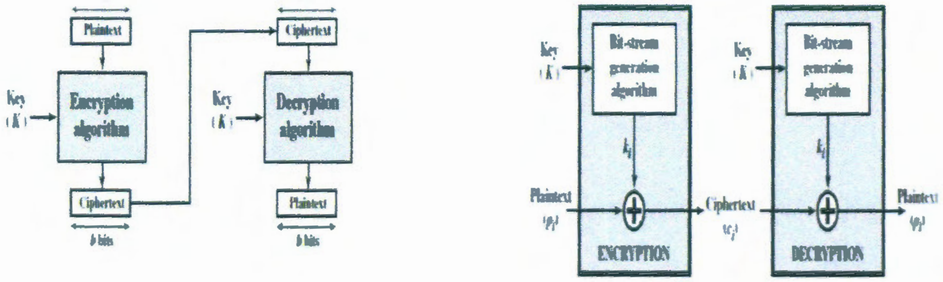


Figure 1-9 Block cipher and block ciphers

1.3.2.1 Classical ciphers

Classical ciphers are ciphers that predate computers and therefore work on letters rather than on bits, [8], there are several well-known ciphers such as:

1.3.2.1.1 Cesar cipher

It is one of the earliest and the simplest ciphers, The Caesar Cipher generalizes into the simple substitution cipher: where each letter in the plain text is shifted by a fixed secret number X to obtain the cipher text.

In the next example we will encrypt a message by shifting each letter by 1 so that the a becomes B and B becomes C and so on and so forth:

Plain text: “CHAOSBASEDCRYPTOGRAPHY”

Cipher text: “DIBPTCBTFEDSZQUPHSBQIZ”

1.3.2.1.2 Vernam cipher

Published by Gilbert Vernam in 1926, its security was formally proven by Shannon, by using the notion of perfect **secrecy**, this cipher is also known as “one-time pad” OTP for short, because the key is aimed at being used for only one plaintext. The Vernam cipher (1926) is defined by:

— the plaintext is a bitstring: an element of $\{0, 1\}^n$
— the secret key is a uniformly distributed element of $\{0, 1\}^n$
— the ciphertext is $CK(X) = X \oplus K$ where $\oplus$ is the bitwise XOR.

The main drawbacks of this cipher are:

» the key must be at least as long as the message.
» The cipher becomes insecure if a key is used twice.
» the security result makes sense only when the key source is truly random.

1.3.2.2 Modern cipher

Modern ciphers are those that uses computer CPU to perform the encryption and the decryption processes, there are several standardized modern ciphers, we will take DES, AES as examples.

1.3.2.2.1 DES

Until the introduction of the Advanced Encryption Standard (AES) in 2001, the Data Encryption Standard (DES) was the most widely used encryption scheme. DES was issued in 1977 by the National Bureau of Standards NDS (now is called the National Institute of Standards and Technology (NIST)), The algorithm itself is called the Data Encryption Algorithm (DEA). For DEA, data are encrypted in 64-bit blocks using a 56-bit key (in fact the key is represented in 8 bytes (64 bit) and the most significant bit MSB is used for parity check.)

As illustrated in Figure 1-10, DES starts by an initial bit permutation IP using the permutation matrix illustrated in Figure 1-11, performs the Feistel cipher using subkeys generated by a key schedule, and finally performs the inverse initial permutation IP using permutation matrix.

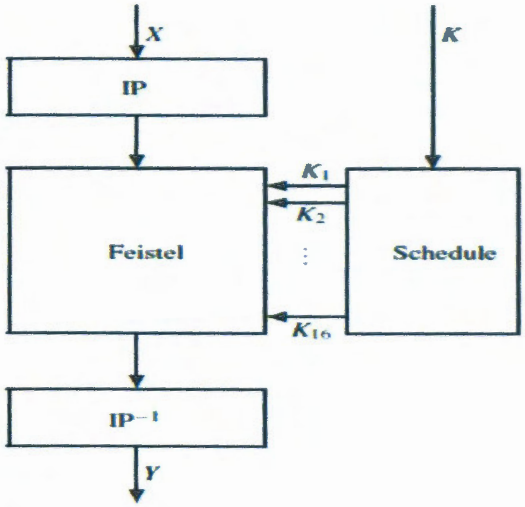


Figure 1-10 DES encryption scheme

8	7	6	5	4	3	2	1
16	15	14	13	12	11	10	9
24	23	22	21	20	19	18	17
32	31	30	29	28	27	26	25
40	39	38	37	36	35	34	33
48	47	46	45	44	43	42	41
56	55	54	53	52	51	50	49
64	63	62	61	60	59	58	57

(a)Original matrix

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

(b) IP matrix

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	27	25

(c) Inverse IP matrix

Figure 1-11 matrixes of Initial Permutation and initial permutation inverse

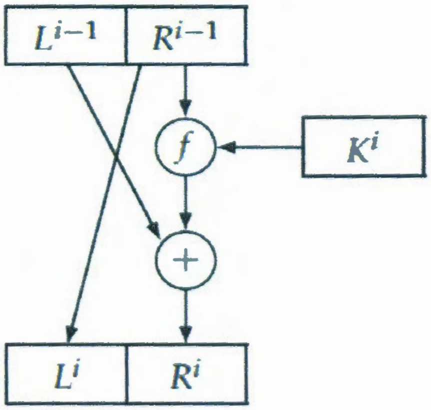


Figure 1-12 One round of DES encryption

DES was subject of many successful attacks, such as the brute force attack led by Rocke Verser, Mat Curtin, and Justin Dolske, using idle cycles of thousands of computers across the Internet in a challenge organized by RSA security [9], as a result, DES is considered to be insecure.

1.3.2.2.2 AES

AES is the most-used cipher in the universe, this symmetric cipher is standardized by NIST in 2000 as a replacement for DES, nowadays, almost modern processors integrates AES instructions such as Intel processors, AMD processors, rather than many other software products such as OpenSSL [10] [8].

unlike the majority of asymmetric encryption algorithms that its security is based on hard mathematical problems such as the discrete logarithm in the case of ElGamel, or the integer factorization in the case of RSA, AES takes it strength from the combination between the permutation and substitution, more commonly known as Substitution-Permutation Networks (SPN) informally we can say that AES is itself a hard problem, this is because many skilled people have tried to break the AES encryption and failed [11]

The block diagram of AES is illustrated in Figure 1-13, we can see that each round contain there operations: substitution (substitutive byte) , permutation (shift row and mix columns) and an XORing of the round key with the result of the previous block substitution-permutation. [8] [11]

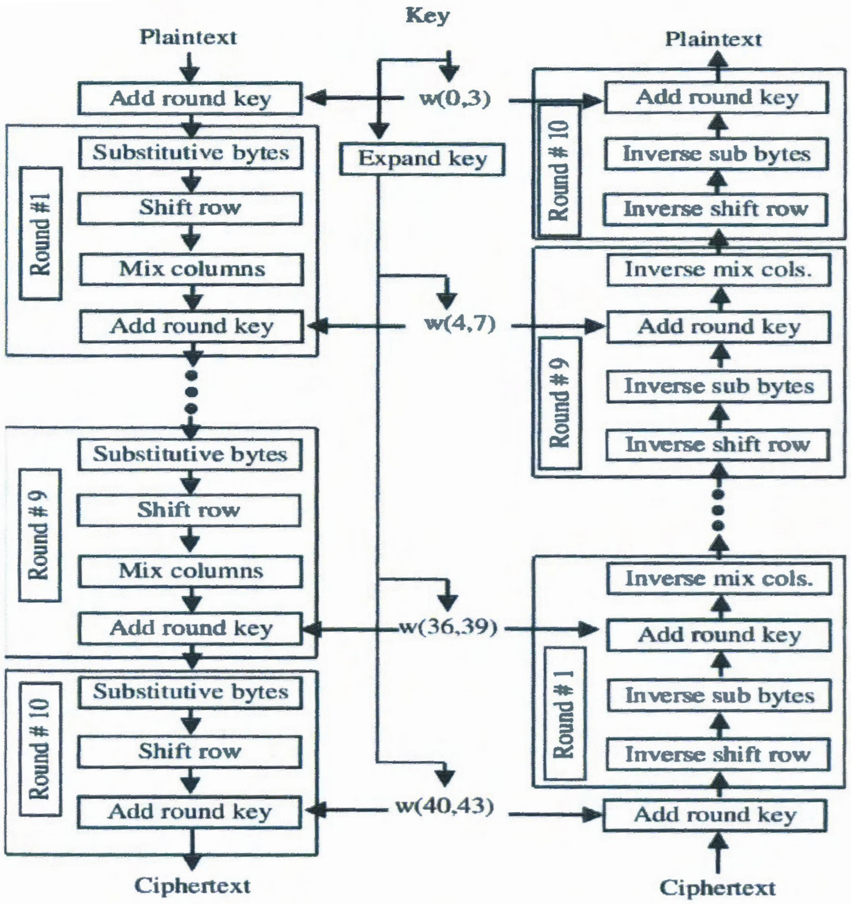


Figure 1-13 Block diagram of encryption and decryption using AES

1.3.3 Asymmetric ciphers

Invention of public-key cryptography is often attributed to Whitfield Diffie and Martin Hellman in a famous paper which was published in 1976, "new directions in cryptography" describing the requirements of the new approach, and challenging the cryptographers to come up with an algorithm that met these requirements, many algorithms have been proposed, the successful algorithm that responses to the challenge was the RSA algorithm, [6] [7] which will be taken in more detail later.

Formally, a public-key cryptosystem is defined by:

— a pseudorandom key generator Gen: this is a probabilistic algorithm which outputs a key pair (Kp, Ks) where Kp is a public key and Ks is a secret key,

— an encryption algorithm Enc: this is an algorithm which outputs a ciphertext Y from an input plaintext X and a public key Kp,

— a decryption algorithm Dec: this is an algorithm which outputs the plaintext X from a ciphertext Y and a secret key Ks.

1.3.3.1 RSA

RSA is the first application of the Deffi-Hellman concepts of public key cryptography, in other words, it was the first asymmetric cipher, invented in 1978, it takes his name from his inventors: R. Rivest, A. Shamir and L. Adleman.

The encryption and decryption using RSA is made as follow:

$$\left[\begin{array}{l} \boldsymbol{C = M^e \bmod n} \\ \boldsymbol{M = C^d \bmod n} \end{array}\right.$$

While the key generation algorithm of RSA is as follow:

1. Select two prime numbers p and q.
2. Calculate n=p*q.
3. Calculate $\phi(n) = (p - 1)\,(q - 1)$.
4. Select e such that e is relatively prime to $\phi\,(n)$
5. Determine d such that de K 1 (mod $\phi(n)$) and $de < \phi(n)$.

The private keys are: (d, n) while the public keys are (e, n).

1.3.3.2 ElGamel

In 1984 T. ElGamel Announced a public key scheme based on discrete logarithms, this scheme is used in a number of standards like DSS, S/MIME [7].

The key generation of ElGamel algorithm is as follow:

1. Generate a random integer X_A, such that $1 < X_A < q - 1$.
2. Compute $Y^A = a^{XA} \bmod q$.
3. A's private key is XA and A's public key is $\{q, a, Y_A\}$.

Any user B that has access to A's public key can encrypt a message as follows:

1. Represent the message as an integer M in the range $0 \leq M \leq q - 1$. Longer messages are sent as a sequence of blocks, with each block being an integer less than q.
2. Choose a random integer k such that $1 \leq k \leq q - 1$.
3. Compute a one-time key $K = (Y_A)^k \bmod q$.
4. Encrypt M as the pair of integers $(C1, C2)$ where

   $C1 = a^k \bmod q$, $C2 = KM \bmod q$

User A recovers the plaintext as follows:

1. Recover the key by computing $K = (C_1)^{XA} \bmod q$.
2. Compute $M = (C_2K^{-1}) \bmod q$.

1.3.4 Cryptographic data integrity Algorithm

1.3.4.1 Hash function

A hash function H is a function that accepts a variable-length block of data of B bit M as input and produces a fixed-size hash value of N bit: h = H(M). Where h is called a preimage of M.

Because H is a many-to-one mapping that maps 2^B different message to 2^N hash value, for any given hash value h, there will in general be multiple preimages, A collision occurs if we have

$M_1 \neq M_2$ and $H(M_1) = H(M_2)$. Because we are using hash functions for data integrity, collisions are clearly undesirable.

The average number of potential collisions is 2^{B-N}, as an example, there are 2^{256} value of 256 bits and a there is 2^{1024} different message of a message of 1024 bits, so the average number of potential collisions (distinct messages that have the same hash value) is $2^{1024-256} = 2^{768}$ collision. [8] [7]

Many algorithms can be considered as hash functions such as MD4, MD5, SHA1, SHA2, SHA3.

1.3.4.2 Cryptographic hash function

The first of the A hash function that satisfies the first two conditions is a weak hash function, if the third condition is also satisfied, then it is referred as a strong hash function. [7]

1. First preimage resistance (one-way property): For any given hash value h, it is computationally infeasible figure out the message y from its hash value h. in other words, its computationally impossible to find y such that $H(y) = h$.
2. Second preimage resistance (week collision resistance): For any given block x, it is computationally infeasible to find two different messages (x, y) with the same hash value h: Such that $y \neq x$ with $H(y) = H(x)$.
3. Collision resistance (strong collision resistance): It is computationally infeasible to find any pair of messages (x, y) such that $H(x) = H(y)$.

1.3.4.3 Message Authentication Code

Also known as keyed hash functions, which is a function that takes a pre-shared key and a data block as an input, it's used to protects a message's integrity and authenticity. And only who has the key can check the validity of the message.

As an example, we will calculate a message authentication code to the same word 'nadjib' with a slight different in the key, which will be in uppercase in the first one and lowercase in the second, we can see clearly that this slight change in the key changes the whole hash value.

SHA256('nadjib','H')=
43721bb8bd6aacb0752c82e25131bb5a11b79a2949e562144faa0b626d19e565

SHA256('nadjib','h')=
498a74efbf731ec83a431631dee59aa3600a9e785903986af59eb29ede7ee27a

1.3.4.4 Use of cryptographic hash function

Cryptographic Hash functions are used in a variety application such as:

- Message authentication: is achieved using Message Authentication Code (MAC).
- Digital signatures: The hash value of the message is encrypted with the user's private key, so that anyone who knows the user's public key can verify the integrity of the message.
- One-way password files: in most cases the hash value of passwords are stored instead of the password itself, this protects passwords even if someone gains access to the files that contains passwords.
- Virus detection: a hash value of each files is stored securely, later, it is possible to determine whether a file has been modified or not by simply recalculating the hash value of the file and comparing the new value with the securely stored one.

1.4 Dynamic systems

Dynamics is primarily the study of the time-evolutionary process and the corresponding system of equations is known as dynamical system. Generally, a system of n first-order differential equations in the space R^n is called a dynamical system of dimension n.

The evolutionary processes of a dynamic system may be deterministic or non-determinacy: A process is deterministic if its entire future course and its entire past are uniquely determined by its state at the present time. Otherwise, the process is nondeterministic.

1.4.1 Discreet time dynamic systems:

The discrete-time process is by difference equations (or maps) given only at equally spaced points of time (integer valued time), it can be mathematically expressed as follow [12]:

$$x_{n+1} = F(x_n)$$

$$F: R^n. Z \rightarrow R^n$$

An example of discrete time dynamical system is the logistic map which represent the growth of a population, It can be expressed mathematically using the following difference equation:

$$x_{n+1} = r * x * (1 - x)$$

Where r is a control parameter in range [0,4].

1.4.2 Continuous time dynamic systems:

The continuous-time process is represented by differential equations: expressed mathematically as follow: [12]

$$x(t) = \frac{dx}{dt} = F(x(t))$$

$$F: R^n. R^n \rightarrow R^n$$

Where the variable t is usually interpreted as time.

If the function F is explicitly time independent, then the system is called autonomous, otherwise, if F is explicitly time dependent, then the system is called non-autonomous. While the dynamical system itself is often called a flow.

A good example of continuous time dynamical systems is the Lorenz system. Which simulate weather patterns.

$$\begin{cases} x = 10(x - y) \\ y = 28x - y - xz \\ z = xy - \left(\frac{8}{3}\right)z \end{cases}$$

The state x(0) or x_0 is called initial conditions, while the set of all points starting from this state are called a trajectory or an orbit. In the case of a flow, the orbit is continuous curve, and a set of disconnected points in the case of a map.

1.4.3 Chaos

In Greek mythology, 'Chaos' is defined as an infinite formless structure, it's almost similar to the common actual usage which means "a state of disorder". Whereas in dynamical systems, chaos can be defined as phenomena that appears in deterministic non-linear dynamic systems, which is characterized by the following five characteristics: [12]

- **Dynamic instability**: Also mentioned as butterfly effect, or sensitivity to initial conditions, where two randomly closed initial conditions with 10^{-14} or smaller have very different orbits.
- **Topological mixing:** Also mentioned as topological transitivity, means that the system evolves over time so that any given region or of its phase space eventually overlaps with any other given region.
- **Aperiodicity**: the system progresses in an orbit that on no occasion replicates itself, that is, these orbits are never periodic.
- **Dense periodic orbits**: it explains that the system follows a dynamic that can diligently approach every potential asymptotic state in random.

Ergodicity: a statistical measurement which gives the same result of the behavior averaged over time and over the space of the chaotic attractor.

—

1.4.4 Attractors

The attractor is the dominant concept in chaos scheme. The word attractor mentions to the elongated behavior of the orbits, and it signifies the region of phase space where the orbits of the system come together after the transient. The attractor A is a dense region where all orbits come together and where the system gets confined

An attractor can be a point, a curve, a manifold, or even a complex set with a fractal structure identified as a strange attractor from geometrical point of view. A transitory explanation of them is as follows:

1.4.4.1 Fixed point:

it relates to a stationary state of the system.

A fixed point x_* of the map $x_{n+1} = f(x_n)$ is such that $f(x_*) = x_*$. If there exists $n > 0$ such that $fn(x_*) = x_*$ and $f k(x_*)\_=x_*$ where $0 \leq k \leq n$ then we say that the point x_* is a periodic point of period n.

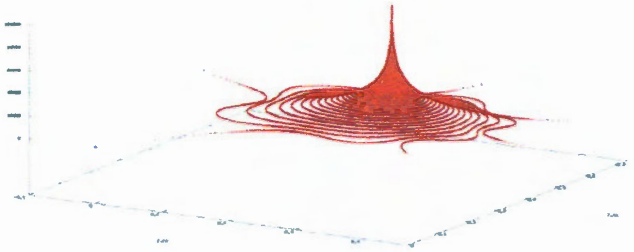


Figure 1-14 fixed point

1.4.4.2 Limit cycle

which is related with a periodic conduct of the system. Once the system arrives with in the attractor the states of the system starts periodic recurrence.

a **limit cycle** is a closed trajectory in phase space having the property that at least one other trajectory spirals into it either as time approaches infinity or as time approaches negative infinity.

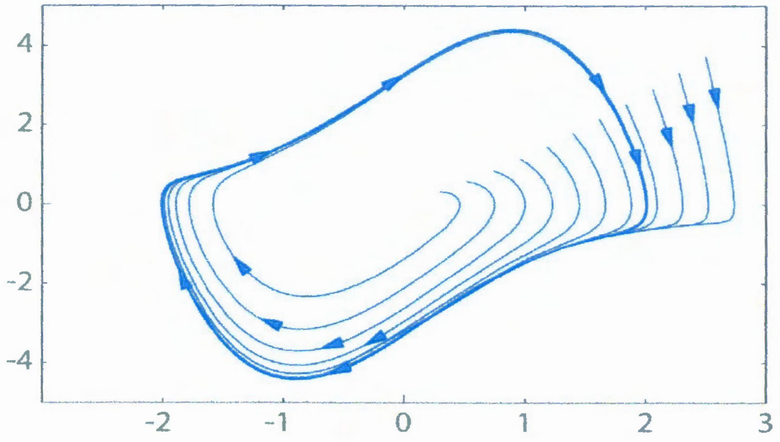


Figure 1-15 limit cycle

1.4.4.3 Manifold

where there is more than one frequency in the periodic trajectories of the system. For example, in the case of two frequencies, the attractor is a 2D-torus.

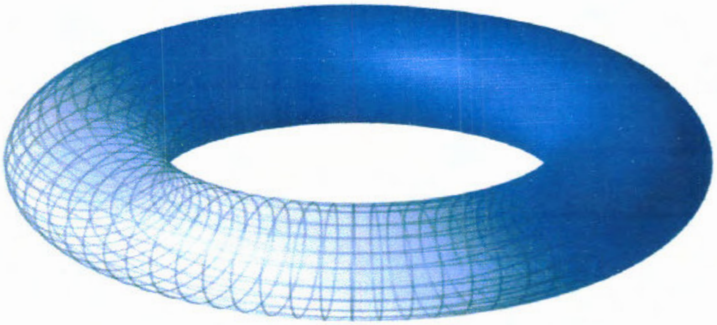


Figure 1-16 Torus

1.4.4.4 Strange attractor

it is informally said to have a complex geometric shape with non- integer dimension. Any state in the attractor evolves within it and never converges to a fixed point, limit cycle or manifold. The dynamics on this attractor is normally chaotic, but there exist also strange attractors that are not chaotic.

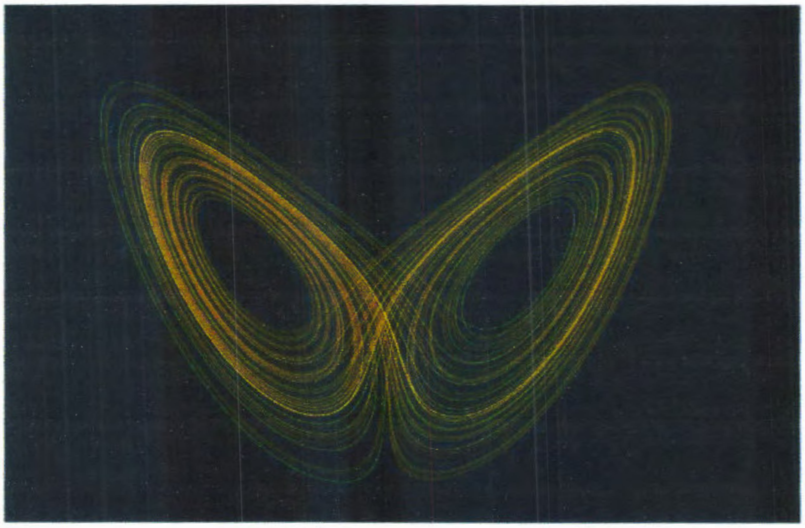


Figure 1-17 Lorenz attractor

1.4.5 Bifurcation

Bifurcation is a qualitative change in the dynamics of a given dynamical system as a control parameter is varied.

We say that a bifurcation occurs if the phase portrait of a dynamical system changes for some parameter r. In other words, if the dynamics of the system for r1 = r−ψ, ψ > 0 are no longer the same as those for r2 = r+ψ, ψ > 0, we say that for that distinct value of r a bifurcation occurred, resulting in qualitatively different phase portraits for the respective values of r. [12]

1.4.5.1 Tent map:

A discreet dynamical system, it expressed using the following formula:

$$X_{n+1} = \begin{cases} rX_n & , X_n < 0.5 \\ r(1 - X_n), & X_n \geq 0.5 \end{cases}$$

The bifurcation diagram of tent map is illustrated in the following figure:

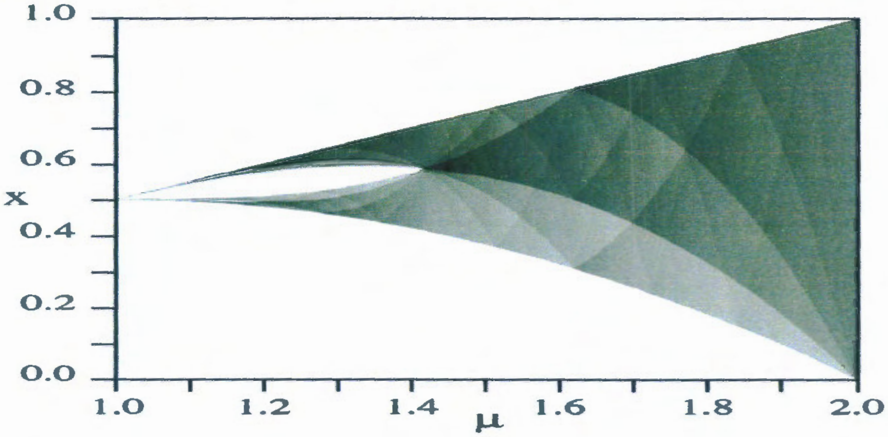


Figure 1-18 Bifurcation diagram of tent map

1.4.5.2 Henon map:

It's one of the most student discreet time dynamical system, expressed mathematically as follow: [12]

$$\begin{cases} X_{n+1}=1-\alpha X_n^2+Y_n \\ Y_{n+1}=\beta X_n \end{cases}$$

This map is chaotic when α=1.4 and β=0.3, as shown in the bifurcation diagram.

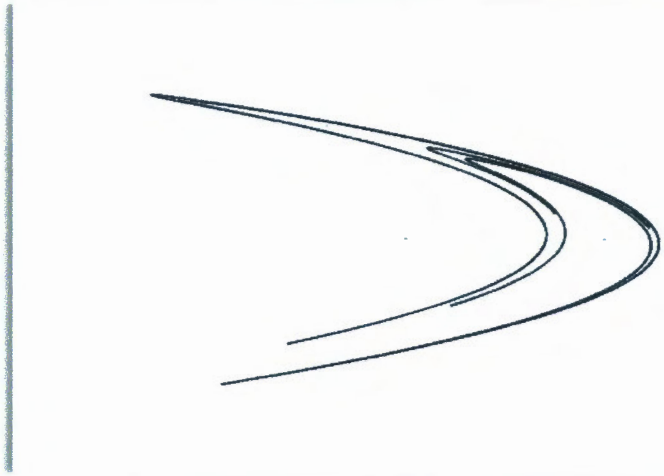


Figure 1-19 Henon map attractor

1.4.5.3 Lorenz system:

Lorenz system is a continuous time dynamical system which represent the atmospheric motions, this system is defined as follow:

$$\begin{cases} \dfrac{dx}{dt} = \sigma(y - x) \\ \dfrac{dy}{dt} = x(\rho - z) - y \\ \dfrac{dz}{dt} = xy - \beta z \end{cases}$$

This system has a chaotic behavior when α=10, ρ=28,β=8/3,

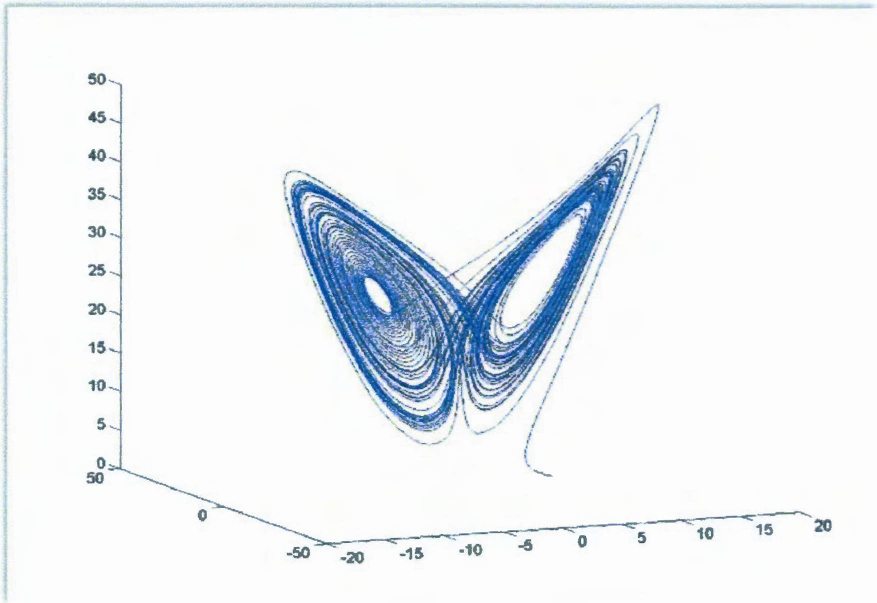


Figure 1-20 Lorenz attractor

1.5 Mixing all together: Chaos-based image encryption:

Chaos-based cryptography is the application of the mathematical chaos theory to the practice of the cryptography, and this is possible because of the shared properties between classical cryptographic systems and chaotic systems, the following table illustrates this property in detail [12].

In contrast with chaos-based cryptography, chaos-based image encryption or cryptography is a special type of chaos-based cryptography which deal with pixels rather than bytes.

Chaotic property	Cryptographic property	Description
Ergodicity	Confusion	The output has the same distribution for any input
Sensitivity to initial conditions	Diffusion with a small change in one plain-text or secret key	A small deviation in the input can cause a large change at the output

Mixing property	Diffusion with a small change in one plain-block or whole plain text	A small deviation in the local area can cause a large change in the whole space
Deterministic dynamics	Deterministic PR	A deterministic process can cause a pseudo-random behavior
Structure complexity	Algorithm (attack)complexity	A simple process has a very high complexity

Table 1-1 shared characteristics between chaotic systems and cryptographic systems

1.6 Conclusion:

In this first chapter we have presented an overview on the three key elements that are directly related to our work: images, cryptography and chaos, first we have listed it basic concepts, some of the most used color models, and the most used file formats, then, we have listed some basics of modern cryptography, the three main types of cryptography symmetric and asymmetric ciphers in addition to cryptographic hash functions. After that we have presented an overview in dynamical systems and chaos, and finally we have ended this chapter by the relation between the chaos and cryptography and images, or what is more commonly known as chaos-based image encryption.

The next chapter will be a state of the art of image encryption schemes, we will list the various image encryption schemes, and we will classify them based on the type of the used chaotic system.

CHAPTER 02

CHAOS-BASED IMAGE ENCRYPTION: STATE OF THE ART

A double precision number is represented in 48 bits (6 bytes), which means that the total space is 2^{48}, which is approximately equal to 10^{14}.

As an example, a cipher which contains 2 different keys, k1 falling in range [0,5] and the other falling in the range [0,1], the total key space of the cipher is:

S= $5*2^{48}*1*2^{48} = 2^{2.32}*2^{96} = 2^{98.32}$

2.3.1.2 Key sensitivity

A good cryptosystem should be very sensitive to the state of its secret keys [14], which means that a slight change in the keys should produce almost totally different cipher images for the same image. In other words, the result of encrypting a plain image with two keys with a slight difference between them (one bit) should be almost different, likewise, a slight change in the decryption key should prevent a successful decryption and produce an almost different plain image.

Generally, a metric named Cipher-text Difference Rate (CDR) is used in order to investigate the sensitivity to secret keys. The CDR is calculated using the following formula [14]:

$$CDR = \frac{Diff(Y,Y_1)+Diff(Y,Y_2)}{2*W*H} * 100\% \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(2.2)}$$

$$Diff(A,B) = \sum_{i=0}^{W-1} Diffp(A(i,j),B(i,j))$$

$$Diff(A(i,j),B(i,j) = \begin{cases} 1, A(i,j) \neq B(i,j) \\ 0, A(i,j) = B(i,j) \end{cases}$$

$$Y = C(I,K)$$

$$Y_1 = C(I,K+\Delta K)$$

$$Y_2 = C(I,K-\Delta K)$$

Where:

- C is the encryption function.
- Y is an encrypted image I using a key K.
- Y_1 and Y_2 are two encrypted images of the same plain image I with a slight change in the key($+\Delta K$ and $-\Delta K$ respectively).
- Diff (A, B) is the sum of different pixels of two given images A and B.

2.3.2 Statistical analysis

2.3.2.1 Information entropy analysis

Entropy of a system is interpreted as an indicator to measure and characterize the amount of disorder in the system. In other words, it represents the necessary information to define the states of the system. The entropy is defined as:

$$H(m) = \sum_{i=0}^{2^N-1} p(m_i) Log_2 \frac{1}{p(m_i)})$$

where *$p(m_i)$* is the probability of the symbol *m_i*, and *N* is the number of intensity levels.

Generally speaking, the maximum value of the entropy is equal to N, so the maximum value of an image component (color level) encoded in 8 bits is equal to 8, while an image composed of 3 components for which component is encoded in 8 bits, like RGB image is calculated as follow:

$$H(RGB) = \frac{H(Red)+H(Green)+H(Bleu)}{3} \quad \text{.. (2.3)}$$

2.3.2.2 Histogram analysis (statistical attack)

The most popular and effective way to test the uniformity of the values (invulnerability to statistical attacks) is the histogram analysis. So, for an encrypted image, it's necessary for an encrypted image to have a uniform distribution of pixel values on both axes [13], in other words a good cipher should produce encrypted images having a uniform histogram as much as possible.

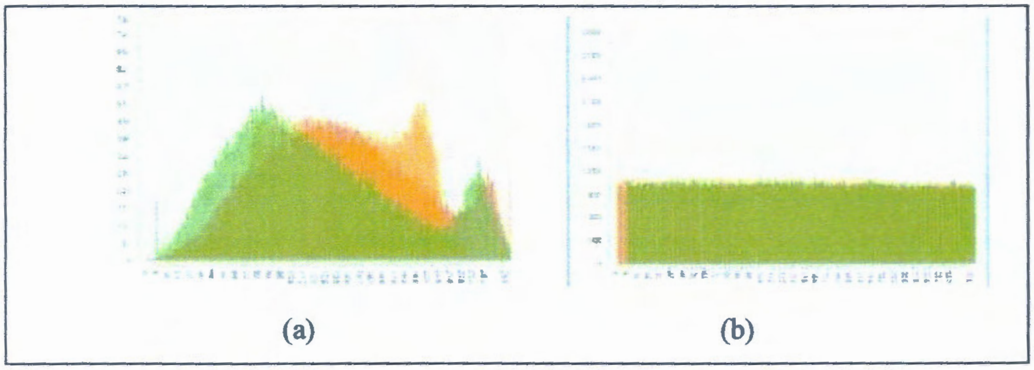


Figure 2-2 (a) non-uniformly distributed histogram (b) uniformly distributed histogram

2.3.2.3 Correlation analysis

No doubt that neighboring pixels of an image are generally highly-correlated, A good cipher should break this correlation . [14], as a result, the encrypted images should have an extremely low correlation among neighboring pixels. The correlation coefficient is calculated using the following formula:

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)D(y)}} \quad \text{.. (2.4)}$$

$$cov(x,y) = \frac{1}{N}\sum_{i=0}^{N}(x_i - E(x))(y_i - E(y))$$

$$D(x) = \frac{1}{N}\sum_{i=0}^{N}(x_i - E(x))^2$$

$$E(x) = \frac{1}{N}\sum_{i=0}^{N}x_i$$

where x and y represent the intensity values of two adjacent pixels. N is the number of pixels in the current analysis sample. $D(x)$ and $E(x)$ stand for the variance and expectation of the current sample.

The correlation value falls in range [-1,1] where 0 represents no correlation and 1 represents a full correlation (images are identical), generally, a correlation up to 0.8 represent a strong correlation.

Generally speaking, the smaller correlation value between adjacent pixels is, the better performance of the encryption algorithm is [15].

2.3.3 Robustness analysis

Images are inevitably contaminated by noise or have data loss during storage and transmission over networks, especially when using unreliable protocols (such as UDP).

Encrypted images should have the ability to resist both noise and data loss attacks, and in order to test this ability, the ability to restore the image after applying different levels of noise or data loss, the peak signal to noise ratio (PSNR) metric is generally used, this metric is calculated by the following formula:

$$PSNR = 10.log\frac{255^2}{MSE}\ (dB) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.5)$$

$$MSE = \frac{1}{W * H}\sum_{i=0}^{W-1}\sum_{j=0}^{H-1}(I_p(i,j) - I_D\,(i,j))^2$$

where IP and ID are the original plain image and the decrypted image, respectively.

The larger the PSNR value is, the higher restoring ability of the decrypted image's is, generally, it's very hard to differentiate the real original image and decrypted image when PSNR is over 35 dB.

2.3.4 Speed analysis

Run time of an encryption algorithm is as important as its security level, especially for real time applications.

Two metrics are commonly used to test the speed performance for a cipher, Encryption Throughput (ET) and Number of Cycles Per Byte (NCPB), both are expressed in the following formula [13]:

$$ET = \frac{Image\_size(byte)}{Encryption_{time}(s)} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.6)$$

$$NCPB = \frac{CPU\_speed}{ET} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.7)$$

2.4 Classes of chaos-based image encryption schemes

We can group the existing chaotic systems based on the type of the chaotic systems used by the cipher into two groups as follow:

- Ciphers that are based on "pure chaotic systems", in other words they are ciphers that exploits only chaotic systems in the encryption/decryption process, this group can be itself divided into the following sub-groups:
  - Homogeneous chaotic systems.
  - Enhanced chaotic systems.
  - Cascaded chaotic systems.
  - Heterogeneous chaotic systems.
- Systems that are based on a "hybrid base", in other terms: they are ciphers that exploits chaotic systems and non-chaotic systems. this group also can be divided into:
  - Combination of chaotic systems and DNA-encoding.
  - Combination of chaotic systems and non-chaotic ciphers.

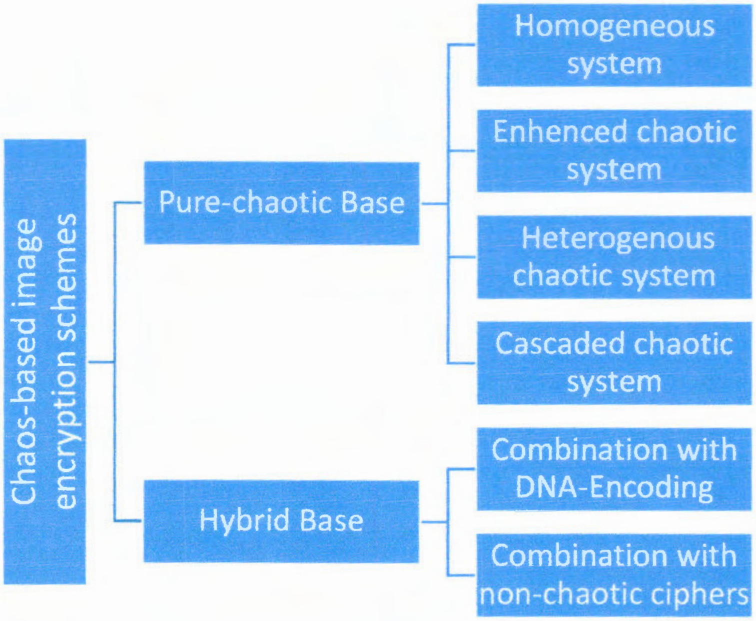

Chaos-based image encryption schemes
Pure-chaotic Base
Hybrid Base
Homogeneous system
Enhenced chaotic system
Heterogenous chaotic system
Cascaded chaotic system
Combination with DNA-Encoding
Combination with non-chaotic ciphers

Figure 2-3 classification of image encryption schemes

2.4.1 Pure chaotic base

2.4.1.1 homogeneous chaotic systems

The homogeneous chaotic systems are systems that exploit one or more chaotic systems of the same type. cipher that exploits the 1-D Logistic map or two distinct Sine maps (in this case, one map is used using different keys.) are examples for homogeneous chaotic systems.

many homogeneous ciphers have been proposed, in the literature, Authors of [16] have proposed a new image encryption technique using a simple chaotic number approach based on confusion and diffusion. the chaotic map offers a simple computation and achieves good bifurcation and high Lyapunov Exponent.

The proposed confusion method exploits the chaotic map for scrambling or confusing the pixel position by making the image as a vector then generating and then sorting a chaotic sequence, this sequence is then used to figure out the new pixel position of the image sequence. On the other sıde, the proposed diffusion process exploits three chaotic sequences generated by three different keys.

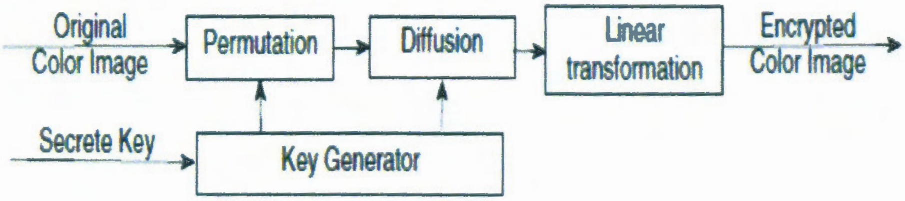


Figure 2-4. The block diagram of the proposed scheme in [16]

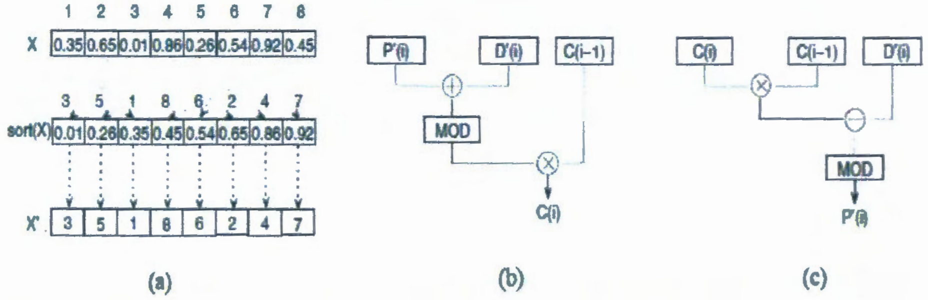


Figure 2-5. The diagram of (a) generating the permutation position matrix , (b) diffusion process in encryption , (c) diffusion process in decryption proposed in [16]

The statistical analysis shows that the proposed cipher produces encrypted images having uniformly distributed histograms which means that this cipher will resist against histograms attack. And it (the cipher) produces pixels with a low correlation coefficient (close to zero) in addition to an entropy value 7.9989 which is close to the ideal value for 256 bit 8. The plain image sensitivity shows that a pixel changes in the plain image result a changing at 99.6108% of pixels in the encrypted image and the UACI =33.4635%.

Authors of [17] have proposed a new chaotic encryption scheme for color images based on a proposed chaotic system which is a combination of three well know 1D chaotic maps (Sine, Logistic and Chebyshev maps) , the combinations are Sine-Sine map (SSM), Logistic-Logistic Map (LLP) and Chebyshev-Chebyshev Map (CCM), the authors show that their system is an effective chaotic system with better chaotic performance and larger chaotic range [0,4]. The results of experiments also showed that the proposed encryption scheme has an excellent performance in image encryption.

The encryption process of this cipher is as follow:

1. Convert the RGB image with the size M*N to M*3N then to a 1D pixel matrix *P*.
2. then generate then sort (ASC) a chaotic sequence *X* using the new chaotic system to obtain the permutation position matrix *X'*,
3. the permuted image pixel matrix P' is obtained using *X'* and *P*.
4. Calculate the Diffusion Matrix *D'*.
5. Calculate the encrypted image pixel matrix *C* from *D'* and *P'*

6. Obtain a new encrypted image pixel matrix *C'* by rotating *C* to the left by the amount of l_p.
7. Convert *C'* into the RGB color image with the size of M*N.

The key analysis shows that the system has a large enough key space to withstand the brute force attack (2^{138}) in addition to a key sensitivity to tiny differences so that a (10^{-14}) change can result almost different images (more than 99.6%) , the statistical analysis in the other hand shows that the system produces encrypted images with a good uniform distribution of pixels regarding to the histogram analysis, with a low correlation among the neighboring pixels(near to 0) due to the correlation analysis in the three directions of a 1000 pairs of adjacent pixels from the plain image and their corresponding in the encrypted image. Also they proved that the algorithm is resistant against both data loss and data noise attacks, by performing data loss attack of 64x64 block size, and 3% Salt&Pepper against the encrypted image, The decrypted image contains most of the visual elements.

2.4.1.2 Enhanced chaotic systems

In this class, a new chaotic system is created by enhancing the characteristics of one or more chaotic systems in terms of randomness of the output and chaotic range. The output of the new system is then used by the cipher.

Many image encryption schemes based on an enhanced chaotic systems have been proposed : authors of [18] proposed a new image encryption scheme for color and grayscale images using a proposed Enhanced Skew Tent Map (ESTM),The new system is made simply by adding a multiplication and applying arithmetic modular (mod1) to the Skew Tent Map as follow:

$$X_{n+1} = \begin{cases} \dfrac{X_n}{b}\ , for\ 1 < X_n \leq b \\ \dfrac{1 - X_n}{1 - b}\ , for\ b < X_n \leq 1 \end{cases}$$

(a) Tent map

$$X_{n+1} = \begin{cases} mod\left(\left(\left(\dfrac{X_n}{b}\right) * 10^5\right), 1\right), for\ 1 < X_n \leq b \\ mod\left(\left(\left(\dfrac{1 - X_n}{1 - b}\right) * 10^5\right), 1\right), for\ b < X_n \leq 1 \end{cases}$$

(b) Modified Tent Map

Where $X_{n+1}\epsilon[0,1]$ is the state of the chaotic system, $b\epsilon[0,1]$ is the control parameter, and $X_n\epsilon[0,1]$.

The comparison of the Pseudo Randomness properties of the both maps in terms of distribution density, Lyapunov exponent and bifurcation, demonstrate the superiority of the new system ESTM compared with STM, and as a result, the ESTM is able to generate a better chaotic stream.

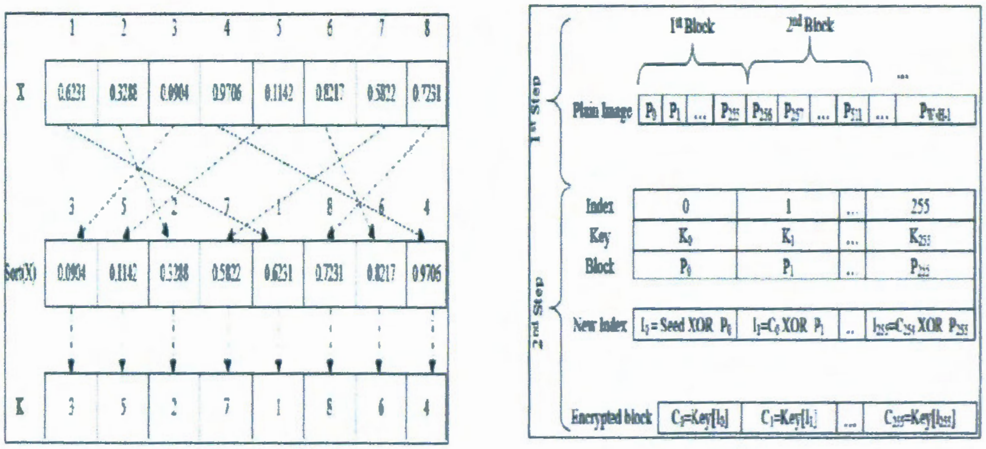


Figure 2-6.Encryption scheme of [18]

Then they proposed an encryption scheme for images based on the previous chaotic system and confusion/diffusion Shanon principles, this system has a large key space (2^{128}) which ensure the resistance against brute force attack, and (the cipher) produces encrypted images with a uniformly distributed histograms and entropy value of 7.999 which means that the proposed cipher will resist against statistical attacks. whereas the UACI and NPCR values (33.5515% and 100% respectively) means that this scheme is immune to a differential attack.

2.4.1.3 Cascaded chaotic systems

A cascaded chaotic system is the result of cascading two or more chaotic systems, in other words: is the result of combining two more chaotic systems in which the output of the first chaotic system becomes the input of the second one, and so on and so forth. The output of the new system is then used to encrypt/decrypt the images.

[19] is an example of encryption schemes that are based on cascaded chaotic systems, authors of this paper have proposed a cascaded 2D chaotic system by combining the Sin and Logistic maps, called 2D-SLM, and based on this new system they have proposed a new image encryption algorithm capable to encrypt various kind of images (grayscale images, color images, biometric images).

The proposed chaotic system in [19] is expressed mathematically using the following formula

$$\begin{cases} x_{i+1} = \alpha(3\sin(\pi y_i) + 1)x_i(1 - x_i) \\ y_{i+1} = \alpha(3\sin(\pi x_i) + 1)y_i(1 - y_i) \end{cases} \quad \text{..} \quad (2.8)$$

Where α is the control parameter within the range of [0,1]. The block diagram of the proposed encryption scheme is as follow:

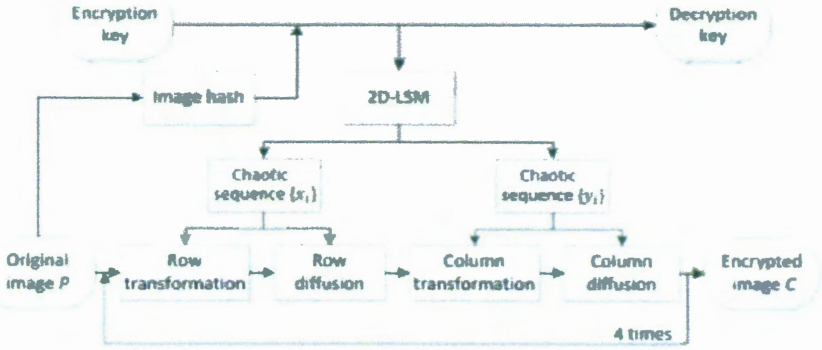


Figure 2-7 The general structure of the cipher proposed in [19]

According to the obtained results, the encrypted images by the proposed cipher has a uniformly distributed histograms, and low correlation between adjacent pixels.

Another example is the encryption scheme proposed by authors of [20], which is an image encryption technique based on bit-plane operation by using PWLCM (2.9) (Piece-Wise Linear Chaotic Map) and cascaded system called Logistic-Adjusted-Sine map (2.10) and using the SHA-256 hash function.

The proposed algorithm consists at first a bit-plane operation using PWLCM system, this operation not only confuses the pixels but also diffuses them simultaneously, then a confusion step by performing row-shuffling and column-shuffling using the 2D Logistic-Adjusted-Sine map. The hash function is used to update the secret keys to make the system resistant against known plaintext attack and chosen-plaintext attack, because a tiny difference in the plain image change its hash value, and as a result changes the key of encryption, this makes the cipher very sensitive to the plain image.

$$X_{n+1} = \begin{cases} \frac{X_n}{\mu} \; if \; 0 \leq X_n < \mu \\ \frac{X_n - \mu}{0.5 - \mu} \; if \; \mu \leq X_n < 0.5 \\ 1 - X_n \; if \; 0.5 \leq X_n < 1 \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2.9)$$

$$\begin{cases} X_{n+1} = Sin\big(\pi * r * (Y_n + 3) * X_n * (1 - X_n)\big) \\ Y_{n+1} = Sin\big(\pi * r * (X_{n+1} + 3) * Y_n * (1 - Y_n)\big) \end{cases} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2.10)$$

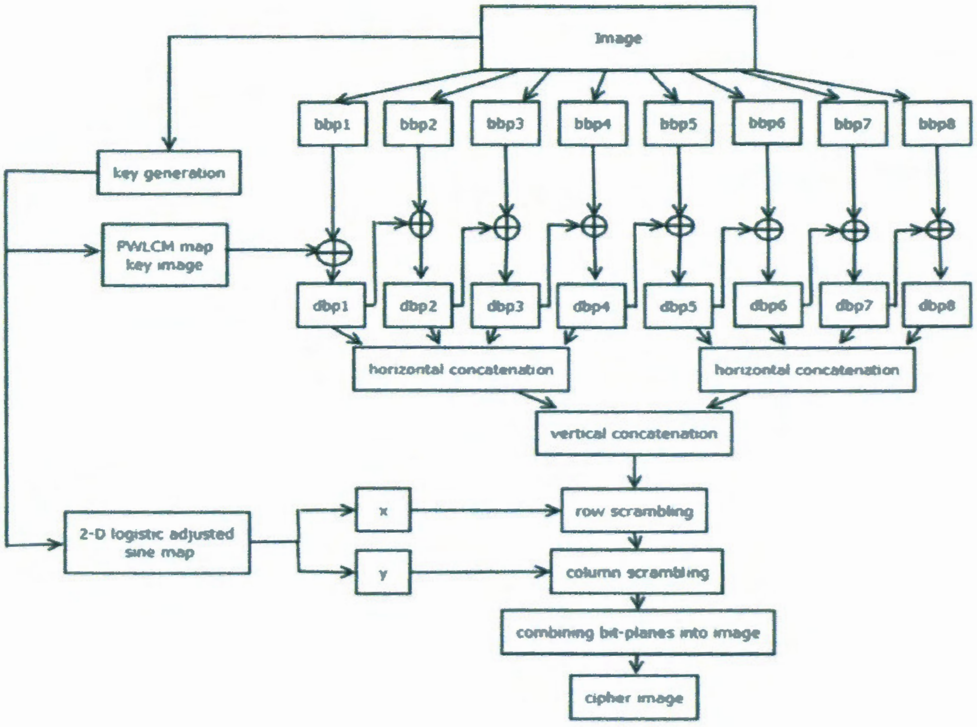


Figure 2-8.The diagram of the encryption scheme based on bit plane operation

The key analysis of the proposed encryption scheme shows that it has a key with a large space equals to 1.11038*2^{377} which is large enough to evade brute-force attack and sensitivity to tiny changes so that a 10^{-14} change in the encryption/decryption key cause a change the value of 99% of the image pixels. Whereas the statistical analysis shows that the proposed cipher produces encrypted images with a uniformly distributed histogram with a correlation coefficient for the three directions (Horizontal, Vertical and Diagonal) near to zero and an information entropy value close maximum theoretical value 8 (7.9973), this proves the resistance of the proposed cipher against histogram attacks, statistical attacks and information entropy respectively.

2.4.1.4 Heterogeneous chaotic systems

Are ciphers that uses the output of different chaotic maps in the cryptosystem in order to encrypt/decrypt the image, for example a cipher that uses the output of a chaotic system in the confusion phase and output of another one in the diffusion one etc.

Authors of [15] proposed a new chaotic system based on Henon map and Lu chaotic map, random overlapping blocks partition and double spiral scans (double spiral scan is illustrated in Figure 2-10.

The input image is first divided into overlapping blocks and pixels of every block are scrambled via double spiral scans. During spiral scans, the start-point is randomly selected using the output of Henon map. Next, image content based secret keys are generated and used to control the Lu chaotic map for calculating a secret matrix with the same size of input image. Finally, the encrypted image is obtained by calculating XOR operation between the corresponding elements of the scrambled image and the secret matrix.

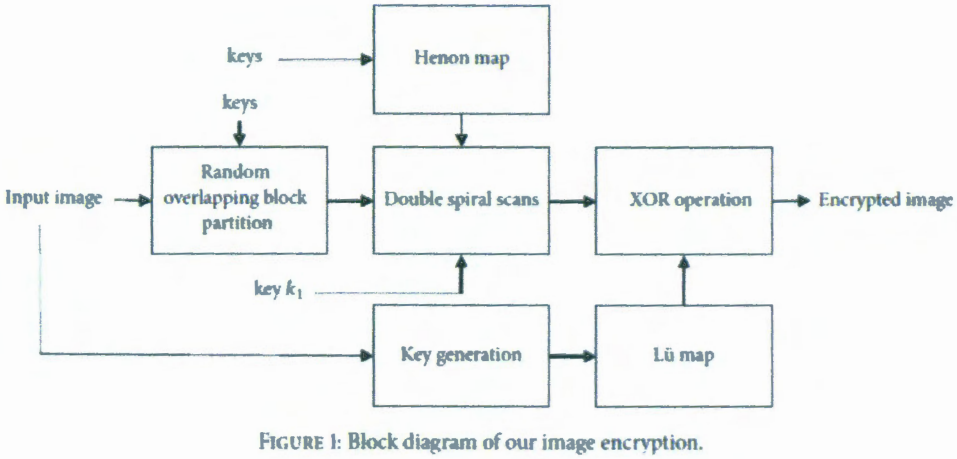


Figure 2-9 The general structure of the cipher proposed in [15]

Experiment results showed that the proposed encryption scheme has a key with a large space equals to 2^{192} which is large enough to resist against brute-force attack, The ciphers produce encrypted images having an average entropy value up to 7.99 and a uniform histogram, and an average correlation among neighboring pixels in the three directions (vertical, horizontal and diagonal) equals to 0.06,

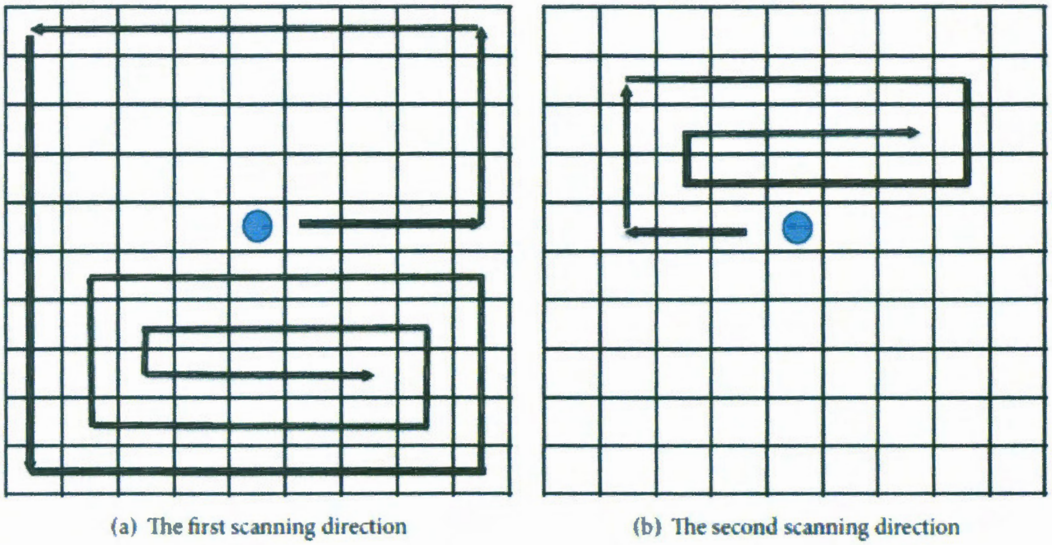

(a) The first scanning direction
(b) The second scanning direction

Figure 2-10 Diagram of double spiral

2.4.2 Hybrid base

2.4.2.1 Combination of chaotic systems and DNA encoding

This class of systems uses DNA encoding with chaotic systems in cipher to encrypt the images.

using DNA encoding technique to encrypt images is not secure, it can be combined with another technique such as chaos. [21]

Before presenting the related algorithms based on DNA encoding and chaotic systems, we will present some basics of DNA encoding and how it can be used to encode bit sequence.

There are four different nucleic acids in a DNA sequence which are named A (adenine), T (thymine), C (cytosine), and G (guanine). Regarding the rules of base pairing, (A) always pairs with (T), and (C) always pairs with (G). It can be concluded that (A) and (T) are complementary, (G) and (C) are also complementary. These relationships are often called the rules of Watson-Crick base pairing.

Similarly, in the binary system, (0,1) are complementary which means that (00,11) and (10,01) are also complements. So, if we use the four deoxynucleotides "A," "T," "G," and "C" to represent the binary numbers "00," "11," "01," and "10," respectively, then each pixel can be encoded into a string of nucleotides, as an example, $(11000101)_2$ is the binary format could be encoded using Rule 3 as (ATGG).

There are 24 types of combinations for the four nucleotides. However, only eight coding combinations are suitable for the principle of complementarity. These rules are summarized in Table 2-1

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

Table 2-1 Encoding and decoding map rule of DNA Sequence of [19]

In the last few years, many ciphers have been proposed [22] [23] [24], authors of [22] proposed a chaotic system based on 3D logistic map and DNA encoding, The 3D logistic map is an extension of the original 1D logistic map using the following formulas:

$$\begin{cases} X_{n+1} = RX_n(1 - X_n) + \beta Y_n^2 X_n + \alpha Z_n^3 \\ Y_{n+1} = RY_n(1 - Y_n) + \beta Z_n^2 Y_n + \alpha X_n^3 \\ Z_{n+1} = RZ_n(1 - Z_n) + \beta X_n^2 Z_n + \alpha Y_n^3 \end{cases}$$

This nonlinear system presents chaotic behavior when:

$0.53 < R < 3.81$, $0 < \beta < 0.022$, $0 < \alpha < 0.015$ and X_0, Y_0 and Z_0 are in [0,1].

The proposed encryption algorithm in [22] is can be written as follow:

1. Determine the value of X_0, Y_0 and Z_0 from the key.
2. Convert the RGB pixels into a vector of 1 Dimension.
3. Generate location_Pi :a random number between [0,SIZE].

4. Select the pixel Pi.
5. Generate 3 random numbers using the 3D logistic map Xi,Yi and Zi,
    - *Yi* is scaled between 1 and 8 to choose one of the DNA rules in order to encode *Pi* to DNA sequence,
    - Zi is scaled between [0, 255] and [0,8], after that, the first number is encoded using the corresponding DNA rule of the second number.
    - Xi is used to decode the result of DNA XORing the encoded *Pi* and encoded random number.
6. The new diffused number is XORed with pervious diffused pixel, then positioned in location_P using a permutation algorithm.

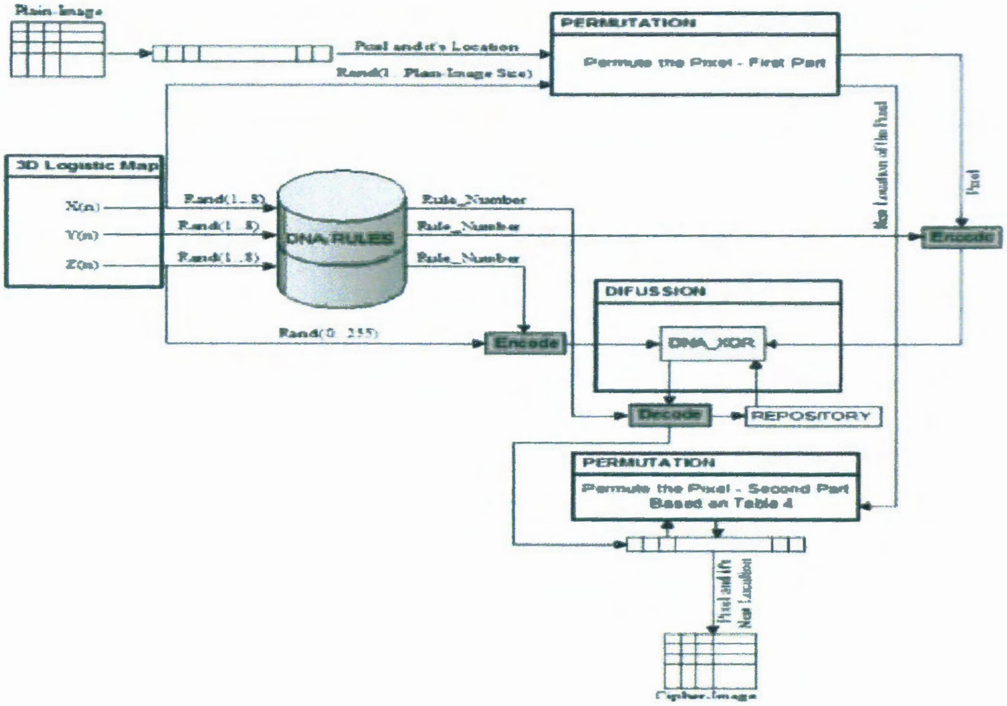


Figure 2-11 the general structure of the cipher proposed in [22]

The proposed encryption scheme has a key with a large space equals to 2^{240} which is large enough to evade brute-force attack, sensitive to tiny changes so that a 10^{-14} change. The cipher produces encrypted images with a uniformly distributed histogram having almost no correlation among neighboring pixels, and an information entropy value close to 8 (7.99).

2.4.2.2 Combination of chaotic and non-chaotic cryptosystems:

This class of ciphers uses a combination of the traditional non-chaotic ciphers (AES, ElGamel, etc.) and chaotic systems in the cipher.

Authors of [25] proposed a new color encryption scheme using a combination of two efficient chaotic maps: Henon map and the logistic map, and the well-known standard symmetric cipher AES.

The scheme first decomposes the image I of size M*N into three matrixes of size M*N with R G B, then permutes the intensity values of the pixels using 2D chaotic Henon map, and then using the 1D chaotic logistic map, after that the pixel values of the three matrices are altered using AES, the three matrices are then rearranged to construct the encrypted image. The decryption process is the reverse process of the encryption.

The key analysis shows that the system has large key space ($3.4 *10^{122}$) with a great sensitivity to its initial conditions (the keys) so that a tiny variation (10^{-14}) make totally different results which means that the system is resistive against brute force attacks. While the statistical analysis shows that the encrypted images histogram is uniformly distributed which means that the encryption scheme is resistant against statistical attacks. they also proved that the encryption scheme will resist against to differential attacks due to the NPCR and UACI values which are greater than 99.6% and close to 33.3% respectively, and robust against different level of Salt&Pepper attack (1%, 5%,10%) and data loss attack by removing different areas from each layer of the cipher image.

Combinations of chaotic systems and elliptic Curve ElGamel scheme are also proposed [26].

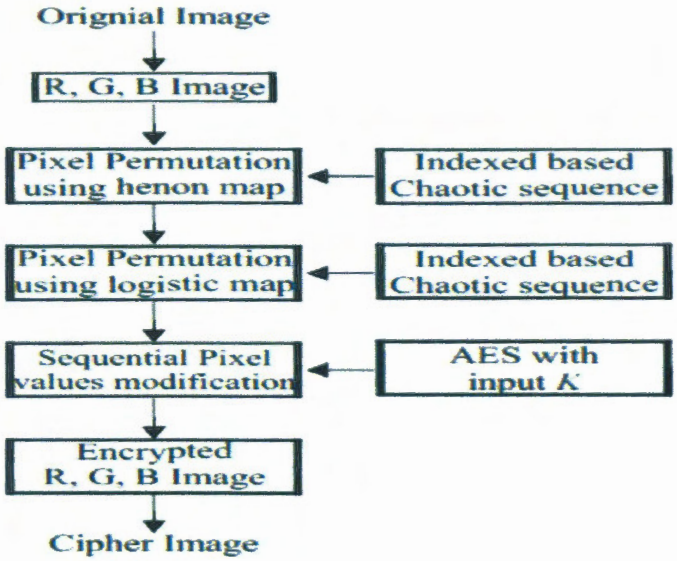

Orignial Image
R, G, B Image
Pixel Permutation using henon map
Indexed based Chaotic sequence
Pixel Permutation using logistic map
Indexed based Chaotic sequence
Sequential Pixel values modification
AES with input K
Encrypted R, G, B Image
Cipher Image

Figure 2-12.The proposed encryption scheme in [25]

2.5 Conclusion:

Based on the previous classification we can conclude that: homogeneous ciphers have generally Simple structure and as a result they are faster and easy to implement and their security mainly depends on the security of the exploited chaotic map which is at almost cases not very high.

On the other hand, encryption schemes based on an enhanced chaotic system and a cascading of chaotic systems are similar to the first class but they have an improved chaotic system and as a result they are more secure compared with the first class. While heterogeneous systems are more complicated and their security level depends on the security of multiple chaotic maps, and as a result they are more secure than the previous classes.

Despite encryption schemes based on only DNA encoding, and balancing numbers are reported to be insecure, combining them with chaotic systems is a promising approach, and it can improve the security of and the performance of the ciphers.

In the next chapter, and in light of the two previous chapters, we will present our encryption scheme, which is a hybrid chaotic system that combine a modified generating function of Lucas balancing numbers with the well-known 1D logistic map.

CHAPTER 03

THE PROPOSED HYBRID CHAOTIC CRYPTOSYSTEM

Chapter 03. The Proposed Hybrid Chaotic Cryptosystem for Image Encryption

3.1 Introduction

In this chapter we will present at the first place some details about the two chaotic functions that we have used in this work : the well-known logistic map, and a modified generating function of a Lucas balancing numbers, we will refer to it as MGF for short, this last has also a chaotic behavior as we will demonstrate later using its bifurcation diagram , further, we will present the proposed hybrid chaotic system which is a combination of the two previously mentioned chaotic systems, then we will present our cryptosystem for color and grayscale images, which is based on the proposed hybrid chaotic system, and a java implementation of the cryptosystem, finally we will prove the security and the high performance of the proposed cryptosystem on both noisy and non-noisy networks using simulation.

3.2 Chaotic functions

3.2.1 Logistic map

The logistic map is one of the most well-known 1-D chaotic maps, a discrete-time dynamical system that represents an idealized population model, defined as:

$$X_{n+1}=rx_n(1-x_n) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.1)$$

Where X_{n+1} and X_n are the actual and the previous outputs of the map respectively, both of them are falling in the range of [0,1] whereas r is the control parameter in the range of [0,4], The logistic map suffers from various problems: first, the limited range of the chaotic behavior which is limited to [3.57,4] which leads to a limited key space, the second problem is the non-uniformity of the outputs of this function, this can reduce the distribution of encrypted image data, also because of its simple structure, generated orbits may be estimated and its initial values (secret key) can be predicted which decrease its security level. [27]

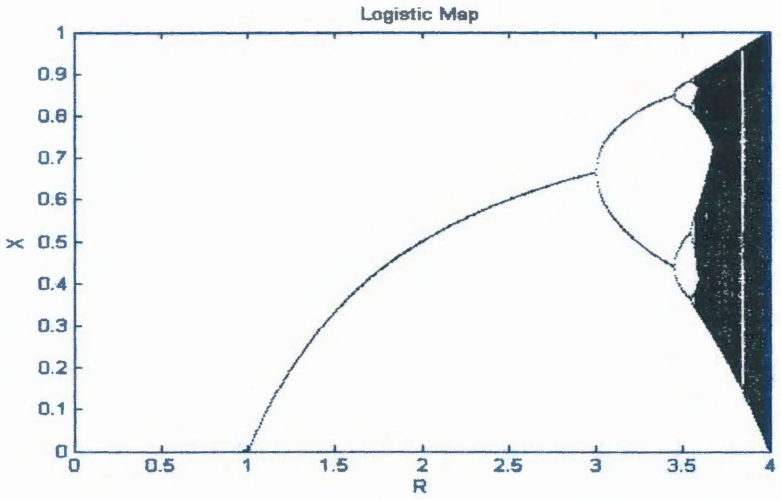


Figure 3-1 Logistic Map

3.2.2 Generating function of Lucas balancing numbers

Recently, Behera and Panda [28] introduced balancing numbers $n \in Z_+$ as solutions of the Diophantine equation

$$1+2+...+(n-1)=(n+1)+(n+2)+...+(n+r). \qquad (3.2)$$

for some positive integer r which is called balancer or cobalancing number. For example $6;35;204;1189$ and 6930 are balancing numbers with balancers $2;14;84;492$ and 2870, respectively. If n is a balancing number with balancer r, then from (3.2) has:

$$\frac{n(n+1)}{2} = rn + \frac{r(r+1)}{2}$$

and so

$$r = \frac{-(2n+1)+\sqrt{8n^2+1}}{2} \; and \; n = \frac{2r+1+\sqrt{8r^2+8r+1}}{2}$$

Let C_n denote the n^{th} Lucas- balancing number. Then

$$\begin{cases} C_{n+1} = 6C_n - C_{n-1}, n \geq 1 \\ C_0 = 1, C_1 = 3 \end{cases}$$

3.2.2.1 The proposed generating function

The terms of the Lucas-balancing numbers satisfy the recurrence relation:

$$\begin{cases} C_{n+1} = 6C_n - C_{n-1}, n \geq 1 \\ C_0 = 1, C_1 = 3 \end{cases} \quad \ldots\ldots\ldots\ldots\ldots \; (3.3)$$

And since we have

$$\sum_{n=0}^{\infty} C_n t^n = \frac{1-3t}{1-6t+t^2},$$

then the generating function for the generalized Lucas-balancing numbers is given by

$$f(t) = \frac{1-3t}{1-6t+t^2}$$

The well-known Binet's formula in the Fibonacci numbers theory [29] [30] allows us to express the generalized Lucas-balancing number in function of the roots r_1 and r_2 of the characteristic equation, associated to the recurrence relation (3.3):

$$r^2 = 6r - 1. \qquad (3.4)$$

Proposition (Binet's formula)

The C_n is the n^{th} Lucas-balancing numbers is given by

$$C_n(x) = \frac{r_1^n + r_2^n}{2}$$

where r_1, r_2 are the roots of the characteristic equation (3.4) whereas $r_1 > r_2$

Proof:

The roots of the characteristic equation (3.4) are $r_1 = 3 + 2\sqrt{2}$ and $r_2 = 3 - 2\sqrt{2}$.

Note that , we have

» $r_2 < 0 < r_1$ and $|r_2| < |r_1|$,

» $r_1 + r_2 = 6$ and $r_1 . r_2 = 1$.

The Lucas-balancing numbers defined above may be generalized by considering the following recurrence relation

$$\begin{cases} U_n = pU_{n-1} - qU_{n-2}, \ n \geq 2 \\ U_0 = 1, \ U_1 = b \end{cases}.$$

where a, b ϵ R and p and q are real numbers.

Furthermore, letting $b = 3$, the generalized Lucas-balancing numbers are obtained for p=6,q=1

Proposition

Let p,q,b ϵ R , a generating function of the generalized Lucas- balancing numbers is derived as follows:

$$g_{p,q}(t) = \frac{1 + (b - p)t}{1 - pt + qt^2}$$

Proof. The generalized Lucas-balancing numbers is given by

$$\begin{cases} U_n = pU_{n-1} - qU_{n-2}, \ n \geq 2 \\ U_0 = 1, \ U_1 = b \end{cases}.$$

The ordinary generating function associated is defined by:

$$g_{p;q}(t) = \sum_{n=0}^{+\infty} U_n t^n = U_0 + U_1 t + \sum_{n=2}^{+\infty} U_n t^n.$$

Using the initial conditions, we get:

$$\begin{aligned} g_{p;q}(t) &= \sum_{n=0}^{+\infty} U_n(x)t^n = U_0 + U_1 t + \sum_{n=2}^{+\infty} U_n t^n \\ &= 1 + bt + \sum_{n=2}^{+\infty} \left[pU_{n-1} - qU_{n-2} \right] t^n. \end{aligned} \qquad (3.5)$$

Considering the fact that j=1-2 and p=n-1 then Eq. (3.5) can be written by

$$\begin{aligned} g_{p;q}(t) &= 1 + bt + pt \sum_{p=0}^{+\infty} \left[U_p(x) \right] t^p - qt^2 \sum_{j=0}^{+\infty} U_j(x) t^j \\ &= 1 + bt + pt \sum_{p=0}^{+\infty} (U_p(x)t^p - 1) - qt^2 \sum_{j=0}^{+\infty} U_j(x) t^j, \end{aligned}$$

which is equivalent to

$$\left(1 - pt + qt^2\right) g_{p;q}(t) = 1 + (b - p)t,$$

Therefore:

$$g_{p,q}(t) = \frac{1+(b-p)t}{1-pt+qt^2} \;\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.6)$$

where p and q are system parameters.

3.2.2.2 Quantifying chaotic behavior in the proposed generating function

In this section, we will investigate the dynamical behavior of the proposed generating function by using numerical simulations. Thus, the proposed generating function may be rewritten in the following recurrence form:

$$x_{p,q}(n) = \frac{1+(b-q)t}{1-pt(n-1)+qx^2(n-1)} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.7)$$

Where p, q and b are system parameters.

Using graphical simulations illustrated in Figure 3-2, we explore the dynamical behavior of x(n) for different values of the parameter q in the range and letting p=6, b=3

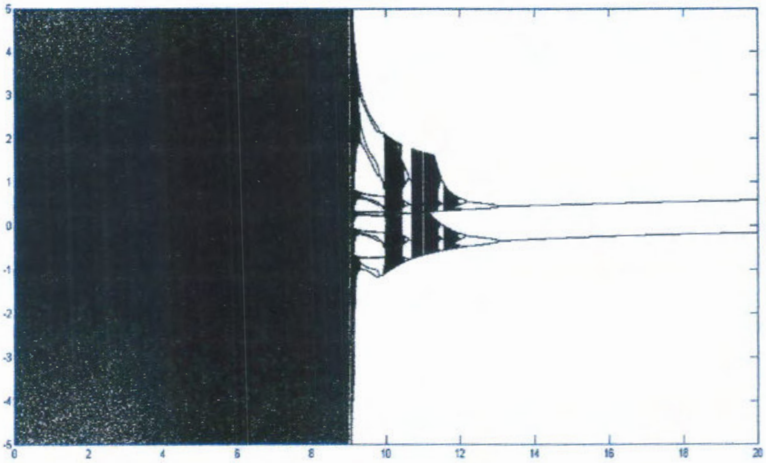


Figure 3-2 Bifurcation diagram of the generating function of the generating function of Lucas balancing numbers

3.2.3 The proposed functions

The main drawbacks of the previous generating function are the non-uniformity of its output, in other words, the chance of getting a number in the output is not equal for all the numbers, in addition to the limited chaotic range of this function which falls in [0,9]

In this part we will try to extend the chaotic range of this generating function, by keeping only the fractional part of the generated numbers, so the proposed modified function is expressed as:

$$\boldsymbol{MGF(x) = |Fract(\frac{1-3x}{1-6x+rx^2})|} \text{ ....(3.8)}$$

Where Fract(x) is a function that returns the fractional number of a given number x, and r is a system parameter. It's clear that the output of this function will be in the range of [0,1], but the most important characteristic of this function is the chaotic behavior and its wider range compared with the original function, which is demonstrated in bifurcation diagram of this function.

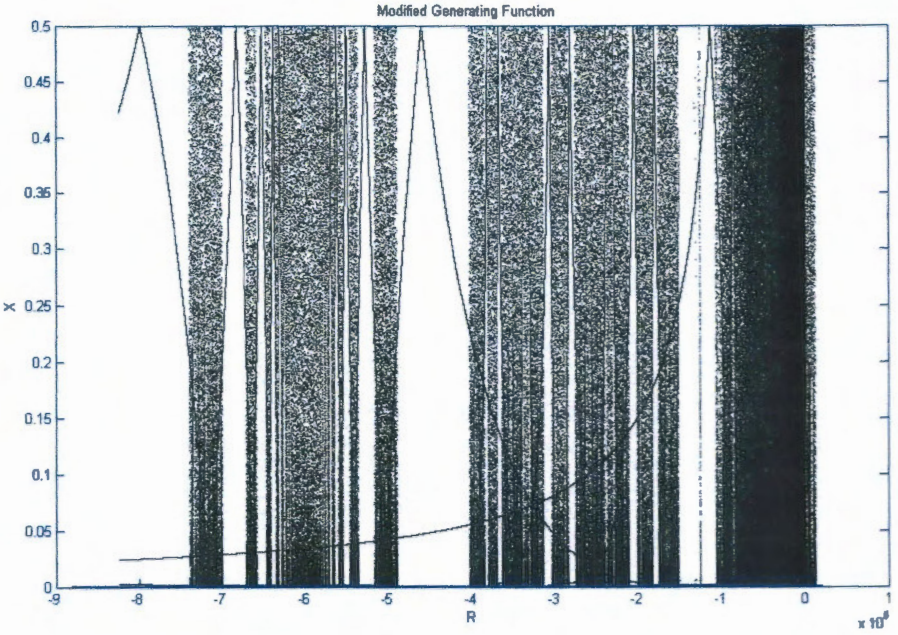


Figure 3-3 Bifurcation of MGF

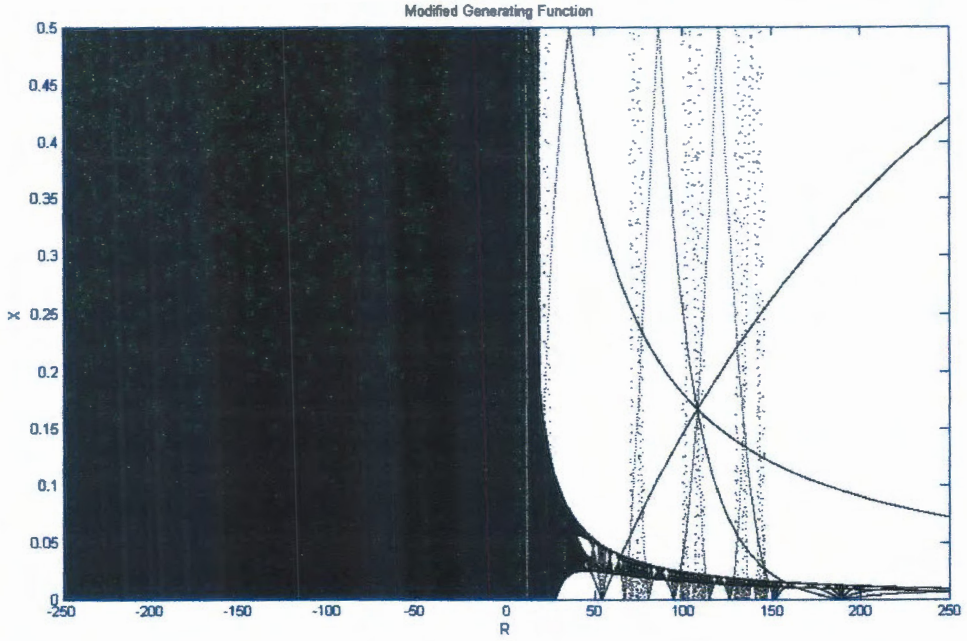


Figure 3-4 MGF in range [-250,250]

As its clear from the bifurcation diagram, this modified generating function has a much more wider chaotic range compared with the original function, the range of first functions is [0,9] while the range of the modified generating function is [-8*10^5,0].

3.2.4 Combining MGF with the logistic map

The main drawback of the previous modified generating function is the non-uniformity of its outputs, which can decrease its security level, and in order to improve it, we will combine it another chaotic function, the logistic map. This combination results two other functions, these functions can be expressed mathematically as follow:

— $MGF(L(x)) = |Fract(\frac{1-3*L(x)}{1-6*L(x)+r*L(x)^2})|$ ....(3.9)

— $L(MGF(x)) = r$*MGF(x)*(1-MGF(x)) ....(3.10)

The bifurcation diagram of these functions which is illustrated in the Figure 3-5 and Figure 3-7 respectively , not only shows a better distribution of the output, but also an infinite chaotic range.

3.2.4.1 The first combination: L (MGF(x))

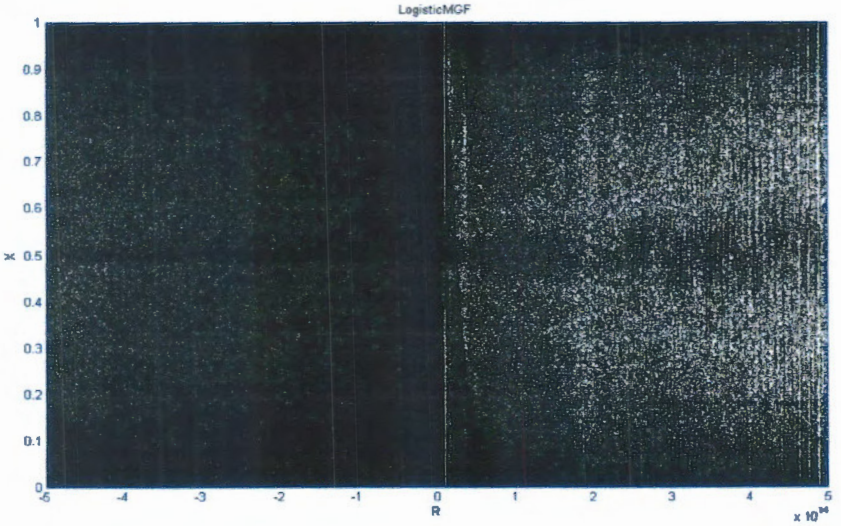


Figure 3-5 Bifurcation of the LogisticMGF function

Figure 3-5 is the bifurcation diagram of the MGF(L(x)) function expressed in (3.9) in the range [$-5*10^{-5}$, $5*10^{-5}$] while Figure 3-6 is the bifurcation diagram of the same function but in a smaller range: [-250 , 250].

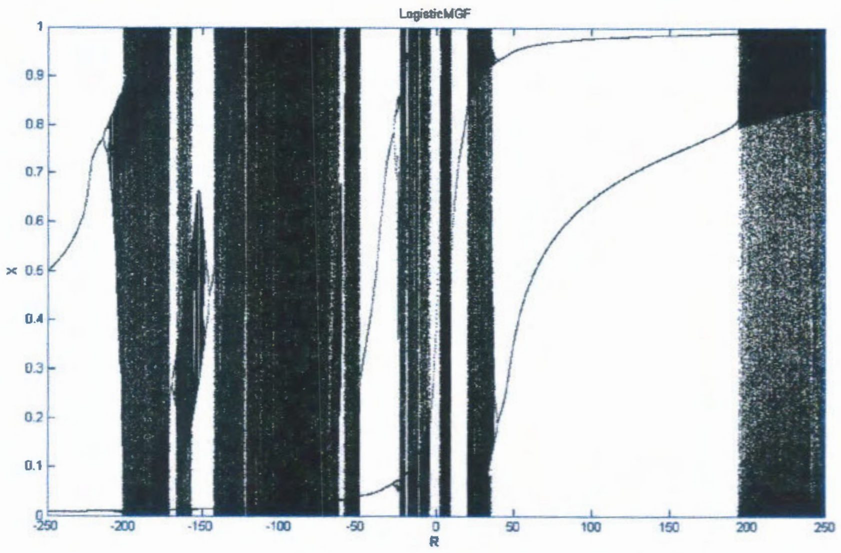


Figure 3-6 Bifurcation of the logisticMGF function in range [-250,250]

3.2.4.2 The second combination: MGF(L(x))

The following diagrams are belong to the L(MGF(x)) function expressed in(3.9): Figure 3-7is the bifurcation diagram of the function in the range [-5*10^{-5} , 5*10^{-5}] while Figure 3-8 is it the bifurcation diagram in: [-250 , 250].

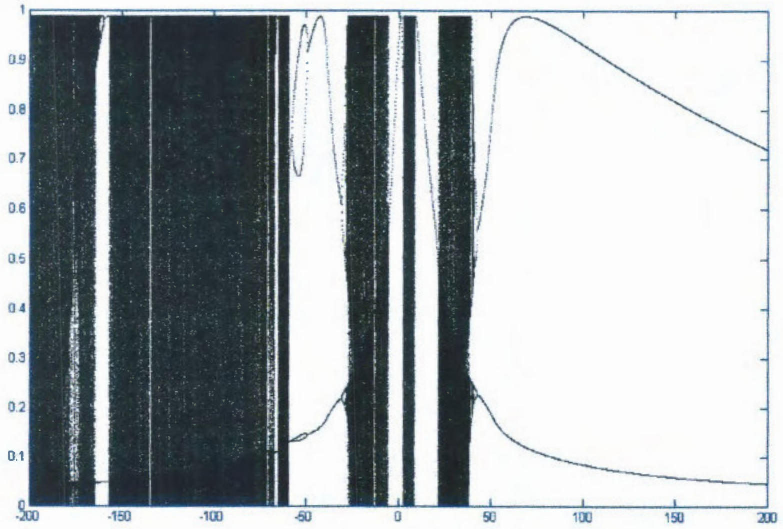


Figure 3-7 Bifurcation of the MGFLogistic function

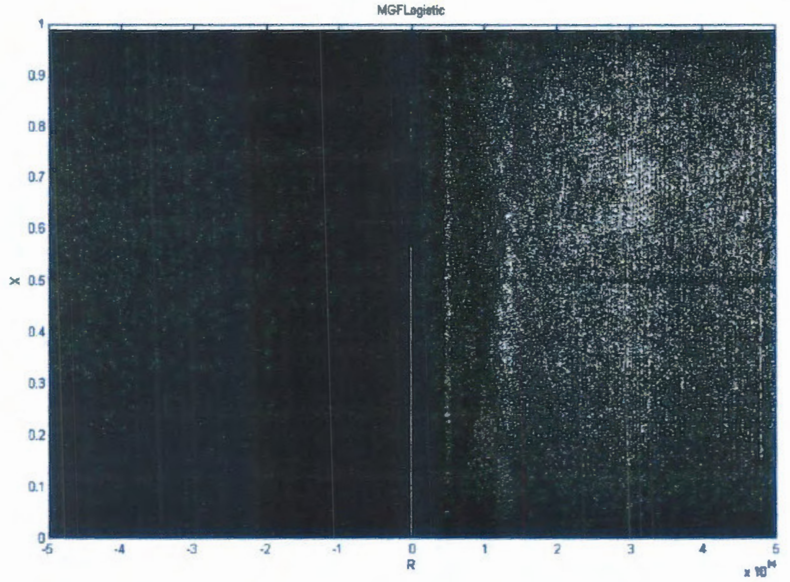


Figure 3-8 Bifurcation of the MGFLogistic function in range [-250,250]

3.3 Cryptosystem conception

The proposed cryptosystem is based on the Shannon's Confusion/diffusion principals, the confusion is just a rearrangement of the image pixels with no modification in the pixel values, the main goal of this phase is breaking the high correlation among the pixels, whereas The diffusion phase is masking the plain pixels by XORing them with a secret chaotic sequence.

3.3.1 The chaos-based Pseudo Number Generator:

Using the logistic map and the Modified Generating Function (MGF for short) we can create four pseudo number generators to generate a sequence of numbers in range [0,1]:

1. Logistic(x , r) : based on the logistic map to generate the chaotic sequence,
2. MGF(x , r) : based on the modified generating function where x is a number in [0,1] and r in range [0 , 10]
3. Logistic(MGF(x,r1) ,r2) : a composition of the two previous functions , x in range[0,1] and r1 in range [3,8] r2 in range [3.57,4]
4. MGF(logistic(x,r1),r2). x in range[0,1] and r1 in range [3,8] r2 in range [3.57,4]

Note: in fact, r1 in the case of the third have no limit range, but this the proposed chaotic function produces uniform output in this particular range, other ranges may exist, this is the same case of r2 in the fourth function.

These functions generate a sequence of numbers in range [0,1] , this range can be easily extended to be [M,N] by simply multiplying by K and then adding N to the sequence

Demonstration:

— Let : $0 \leq X \leq 1$
— Multiplying By K we get: $0 \leq X \leq K$
— Adding N we get: $N \leq X+K \leq K+N$

— Replace (X+K) by X’ and (K+N) by M : N ≤ X’≤ M.

For example, to extend a generated number to be in range [0,255] we simply multiply this number by 255.

3.3.2 Encryption process:

The general flowchart of the encryption is illustrated in Figure 3-9 and explained in following algorithm , While Figure 3-10 , Figure 3-14, illustrates the confusion and the diffusion steps respectively in more details.

Encryption Algorithm:

1. Choose an image Pi of W with and H height.
2. Define the initial values of the chaotic functions (the secret keys).
3. Generate two chaotic sequences using the chaos-based PRNG:
   a. rShift[] : used by the row shift function to scramble the image pixels vertically, its length is equal to W and its maximum value equal to H.
   b. cShift[]: used by the column shift function to scramble the image pixels horizontally, its length is equal to H and its maximum value equal to W.
4. Repeat N times :
   a. Shift each row of the image using row shift function.
   b. Shift each column of the image using column shift function.

5. Generate another chaotic sequence “Mask” using the PRNG , this sequence should has a length equal to 3*W*H and a maximum value equal to 255 ,
6. Calculate the encrypted image Ei by XORing each color of each pixel with a number of the generated sequence.

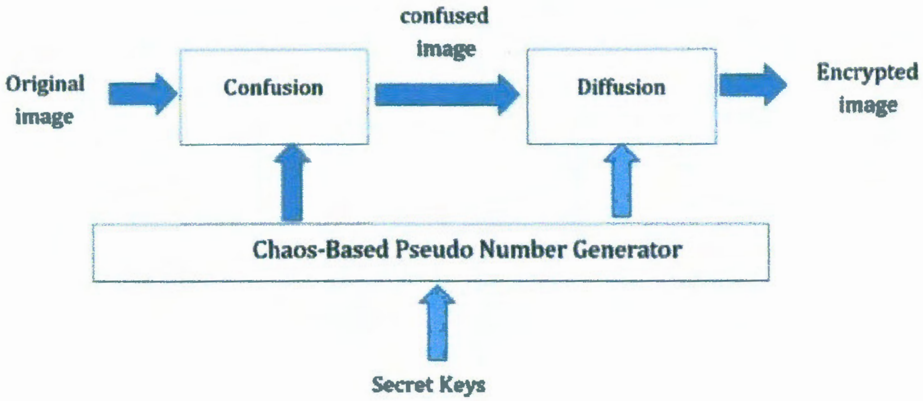


Figure 3-9 The encryption flowchart

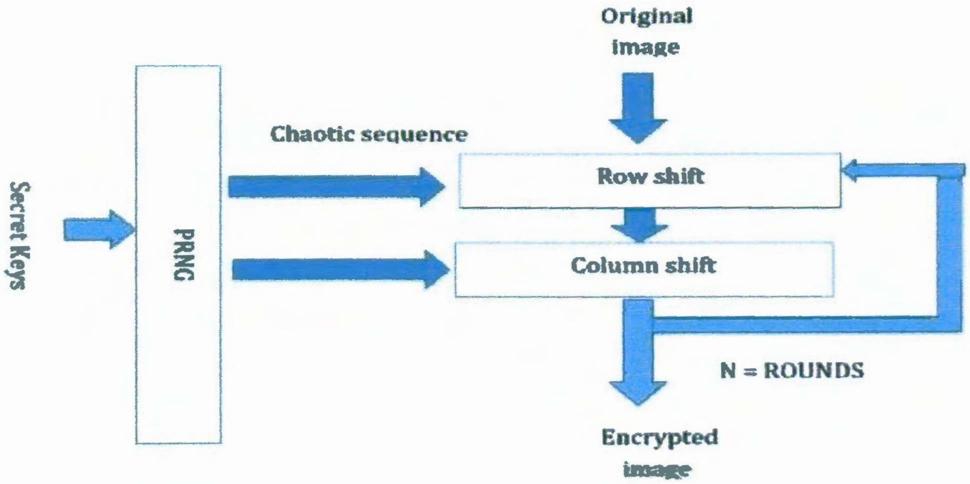


Figure 3-10 Confusion flowchart

Where row Shift is a function that shifts each row of the image down by number Ni generated using the chaos based PRNG. Alike column Shift is a function that shifts each column of the image to the right by number Nj generated using the chaos based PRNG. Whereas ROUNDS is the numbers of rounds.

The confusion process is illustrated in the following flowchart:

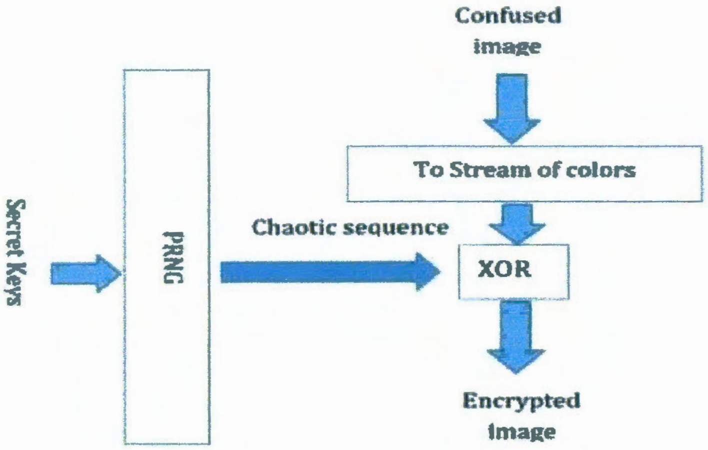


Figure 3-11 Diffusion flowchart

3.3.3 Decryption process

The general flowchart of the decryption process of the proposed cryptosystem is illustrated in Figure 3-13 and explained in following algorithm , While Figure 3-14 illustrates the reverse confusion step, the diffusion step in the decryption process is the same of the encryption process.

Decryption Algorithm:

1. Choose an encrypted image Ei of W width and H height.
2. Define the initial values of the chaotic functions (the secret keys).
3. Generate a chaotic sequence "Mask" using the PRNG , this sequence should has a length equal to 3*W*H and a maximum value equal to 255 ,
4. Calculate the encrypted image Ei by XORing each color of each pixel with a number of the generated sequence.
5. Generate two chaotic sequences using the chaos-based PRNG:
    a. rShift[] : used by the reverseRowShift function to scramble the image pixels vertically, its length is equal to W and its maximum value equal to H.
    b. cShift[]: used by the reverseColShift function to scramble the image pixels horizontally, its length is equal to H and its maximum value equal to W.
6. Repeat N times to get the decrypted image :
    a. Shift each column of the image using reverseColumnShift function.
    b. Shift each row of the image using reverseRowShift function.

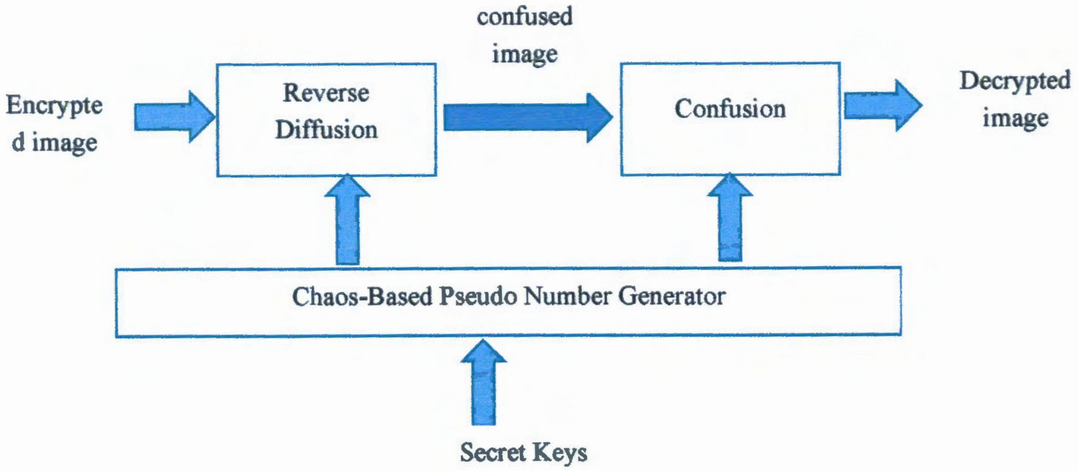


Figure 3-13 Decryption flowchart

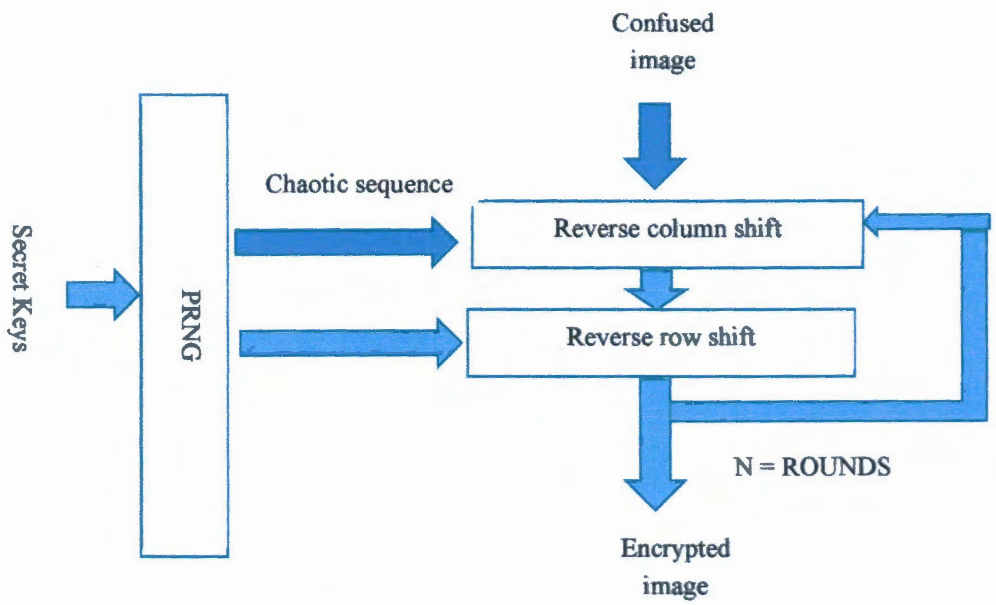


Figure 3-14 Reverse confusion flowchart

Where reverse row shift and reverse column shift are the inverse function of row shift and column shift respectively which shifts each row respv column of the image down respv left by number Ni respv Nj generated using the chaos based PRNG.

3.4 Cryptosystem Implementation

3.4.1 Hardware

This application has been developed using a Laptop Dell Inspiron N-5010 which has the following characteristics:

- RAM : 4 GB
- Processor : Intel ® Core^{TM} i7-2670QM_CPU @ 2.20GHz octa Cores
- Operating system :Windows 10 Pro N 64-bit

3.4.1.1 Software

The bifurcation diagrams of the proposed functions are made using Matlab. while the application is written using Java as a programming language.

The java code and the graphical interface are written using NetBeans IDE v8.2 and SceneBuilder respectively, the proposed tools by Oracle to develop java application and javaFX GUI: Graphical User Interfaces for short.

Also we used the following libraries:

- jFeonix: is an open source java library, that implements Google Material Design using java components [31]

- controlsFX: is an open source project for JavaFX that aims to provide a high quality user interface controls and other tools to complement the core JavaFX distribution [32]

3.4.2 Data

In this work, we focused on using standard test images like "Baboon.jpg", "Lena.jpg" and "Barbara.jpg", These images are downloaded from the following websites:

- Image database of University of Waterloo:
  - » http://links.uwaterloo.ca/Repository.html
- Image database of University of Wisconsin-Madison:
  - » https://homepages.cae.wisc.edu/~ece533/images

3.5 The application:

The application is simply a security and performance testing platform of our encryption scheme, this application is based on the MVC model, (Model View and Controller), illustrated in the following figure:

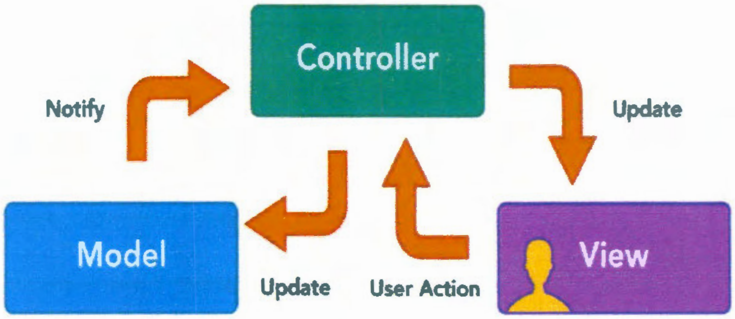


Figure 3-15 Diagram of MVC model

3.5.1 Model:

Generally, the model contains data structures It directly manages the data, logic of the application. in our case it contains the ciphers implementation in addition to statistic methods and performance and analysis functions.

3.5.2 View

- contain the representational information or the graphical user interfaces, in our case it contains the following interfaces

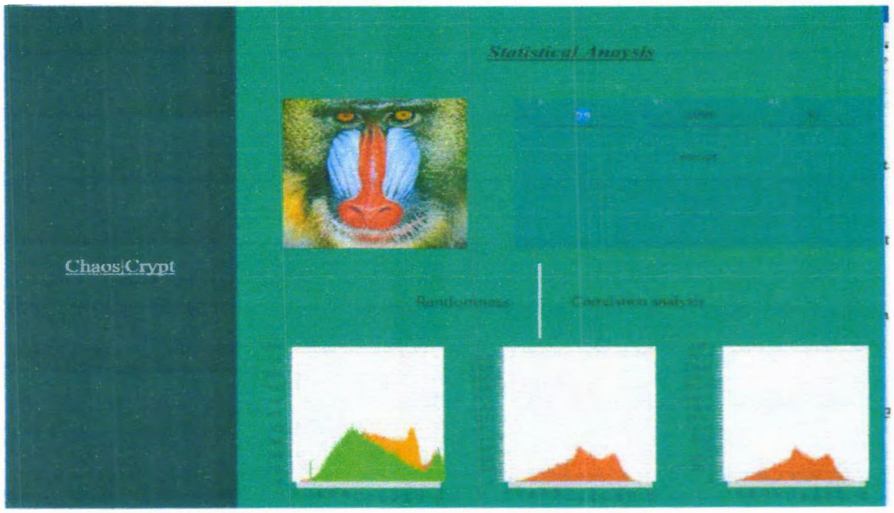


Figure 3-16 statistical analysis interface

This interface show some statistical information about a selected image, basically, the histogram of the image, and pixel correlation graphs in the three directions, vertical, horizontal and diagonal, it also allows the encryption of the selected image and of course showing its statistics

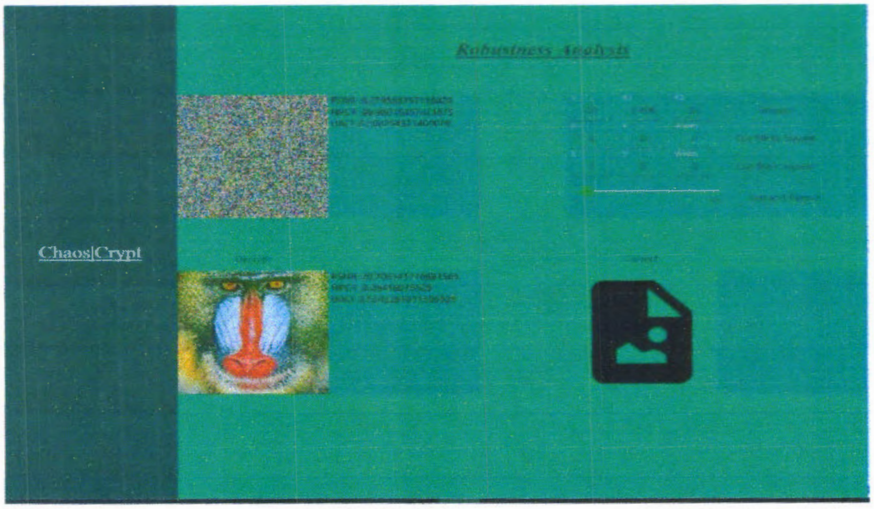
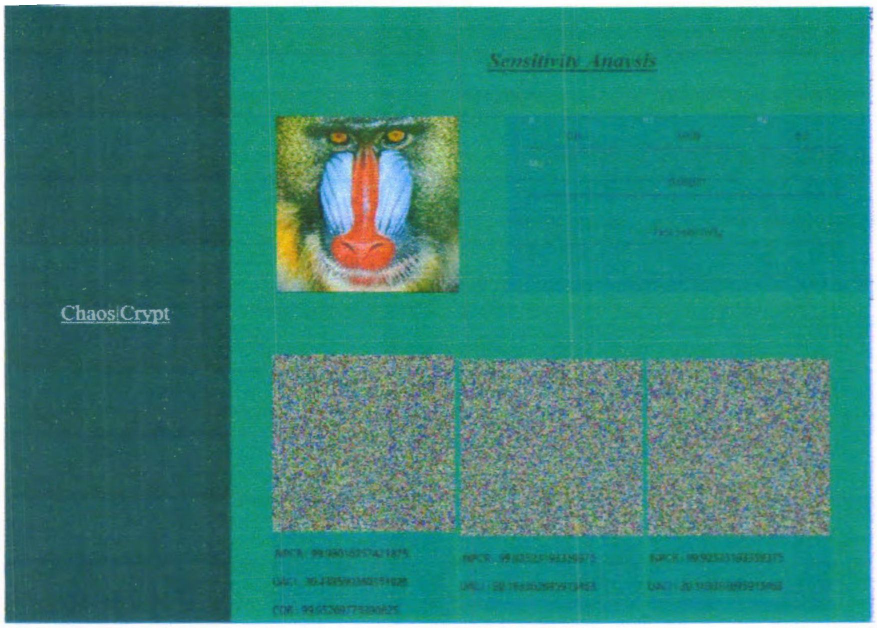


Figure 3-17 Robustness analysis interface

This interface contains robustness tests of the cipher, it allows the user to add different types of noise and data loss to a selected image, and testing the restoring ability, by showing both the decrypted image and useful information like the PSNR, UACI and NPCR.



NPCR : [illegible]
UACI : [illegible]
CDR : [illegible]

NPCR : [illegible]
UACI : [illegible]

NPCR : [illegible]
UACI : [illegible]

Figure 3-18 Sensitivity analysis interface

In this interface, the user can test the sensitivity of the encryption key, it shows the three encrypted images using the three different keys (K1, K1+μ, K1-μ) and the necessary statistics about the image, typically, NPCR, UACI and the CDR values.

3.5.3 Controller

- Responsible for converting the user actions into commands to the model and the outputs of the model into commands to the view, in our case it contains the controllers of the previously mentioned interfaces.

3.6 Experimental results

In this section we will present the security and performance results of the java implementation of the proposed cryptosystems described in the previously.

3.6.1 Key analysis

3.6.1.1 Key space analysis

The proposed CS has 3 keys: X in range [0,1] , R1 in range [3.57,4] and R2 in range [3,8], which means it's key space according to the formula (2.1) is equal to:

We have:

$K_1=2^{48}$, $K_2=0.47*2^{48}$, $K_2=5*2^{48}$.

So:

$S=Sk_1*SK_2*SK_3$

$S=2^{48}*0.47*2^{48}*5*2^{48}$

$S=2^{146}$

Which is larger than 2^{100}, this proves that the proposed cryptosystem is resistant against brute force attacks.

3.6.1.2 Key sensitivity

In order to test the sensitivity of the key, we will encrypt a test image using a key K_1, then we will change one bit of the key (10^{-14}) in other words, we will encrypt the image using the following keys: $K_2 = K_1 + \mu$ and $K_3 = K_1 - \mu$. After that we will encrypt this test image again using the new key, and finally we will calculate the difference between the two encrypted images using CDR metric (described in formula (2.2)). Table 3-1 contains the obtained results.

Altered key	Value	CDR %
X	+ μ	99.902915%
	- μ	99.964714%
R1	+ μ	99.93267%
	- μ	99.980926%
R2	+ μ	99.975585%
	- μ	99.975585%

Table 3-1 CDR values of 'baboon.jpg'

As we can see, the proposed cryptosystem is very sensitive to slight changes ($\mu = 10^{-15}$) in any one of the three keys, as a result, this slight change cause more than 99.9% change in the encrypted image. In other words, changing only one bit change in the key results 99 % encryption/decryption results, this means that only one bit change in the key will prevent a successful encryption/decryption.

3.6.2 Statistical analysis

3.6.2.1 Information entropy analysis

The following table gives the information entropy of several test images of various sizes and the corresponding entropy value of three levels Red, Green and Blue.

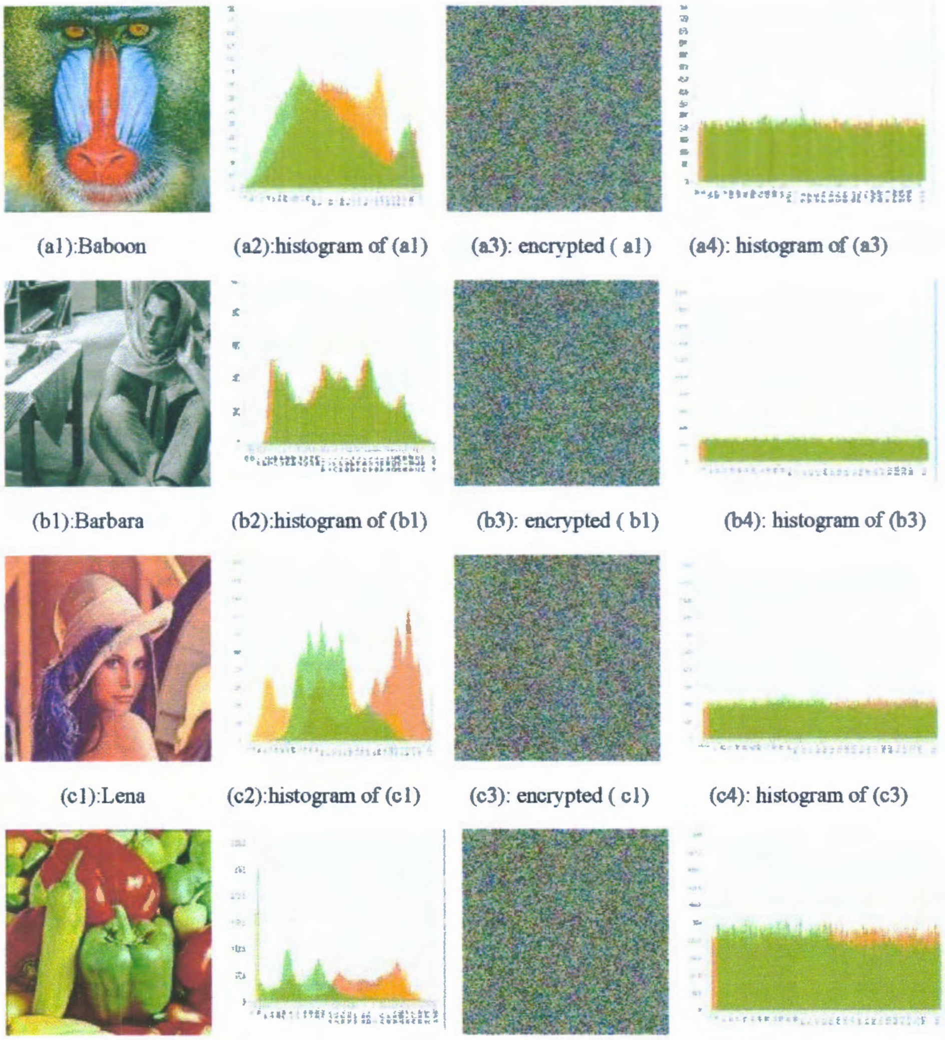
Test image	Image size	Plain Image entropy	Encrypted image entropy			
			Red	Green	Bleu	(R+G+B)/3
Baboon	256x256	7.6588	7.996775	7.997443	7.996775	7.996998
	512x512	7.6463	7.999215	7.999181	7.999215	7.999204
Barbara	256x256	7.6399	7.997429	7.997457	7.997429	7.997438
	512x512	7.6337	7.999266	7.999255	7.999266	7.999262
Lena	256x256	7.3185	7.996686	7.997163	7.996686	7.996845
	512x512	7.2722	7.999175	7.999279	7.999175	7.999209
Pepper	256x256	7.3829	7.997665	7.99696	7.997665	7.99743
	512x512	7.3697	7.99925	7.999341	7.99925	7.99928

Table 3-2 information entropy of test images and their encrypted correspondings

It's clear that all the entropy value of all encrypted images is near to the ideal value 8, which means that proposed cryptosystem produces random images.

3.6.2.2 Histogram analysis (statistical attack):

In this experiment we have chosen four test images of size 512x512: figures: a1, b1, c1, d1 While figures a2, b2, c2, d2 represent their corresponding histograms. encrypted images in the other hand and their histograms are illustrated in a3, b3, c3, d3 and a4, b4, c4, d4 respectively.



(a1):Baboon (a2):histogram of (a1) (a3): encrypted (a1) (a4): histogram of (a3)

(b1):Barbara (b2):histogram of (b1) (b3): encrypted (b1) (b4): histogram of (b3)

(c1):Lena (c2):histogram of (c1) (c3): encrypted (c1) (c4): histogram of (c3)

(d1):Pepper (d2):histogram of (d1) (d3): encrypted (d1) (d4): histogram of (d3)

We can clearly see: the encrypted images histogram has a uniform distribution of pixel values (all pixels has the same chance of appearance), this is proved that the cryptosystem is not vulnerable to histogram attack.

3.6.2.3 Correlation analysis

In this experiment we randomly chosen 2000 pixels of the plain image and their corresponding in the encrypted image and then calculate the correlation coefficient using formula (2.4).

Direction	Horizontal direction		Vertical direction		Diagonal direction	
Test Image	Original image	Encrypted image	Original image	Encrypted image	Original image	Encrypted image
Baboon	0.917	0.012	0.868	0.005	0.861	0.004
Barbara	0.953	0.005	0.972	-0.0019	0.967	-0.0240
Lena	0.975	0.006	0.969	-0.001	0.882	0.002
Pepper	0.961	-0.002	0.964	0.005	0.958	0.007

Table 3-3 Correlation coefficient in the horizontal, vertical and diagonal

We can clearly notice that the correlation coefficient of the plain images is up to 0.9 and close to 1, this means that the plain images are high correlated, This is not the case in the encrypted images, which their values are close to 0, this means that they have almost no correlation among the adjacent pixels.

The low correlation among the adjacent pixels proves that proposed algorithm can break the correlation between them.

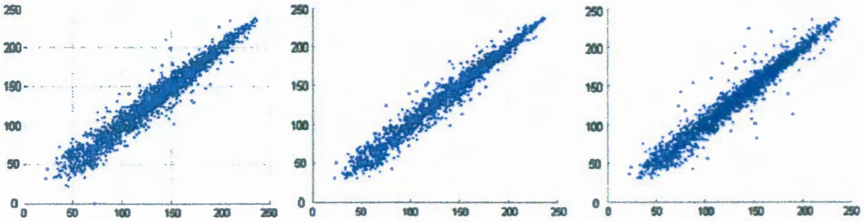


Figure 3-19 correlation of neighboring pixels of the plain image in the vertical horizontal and diagonal directions

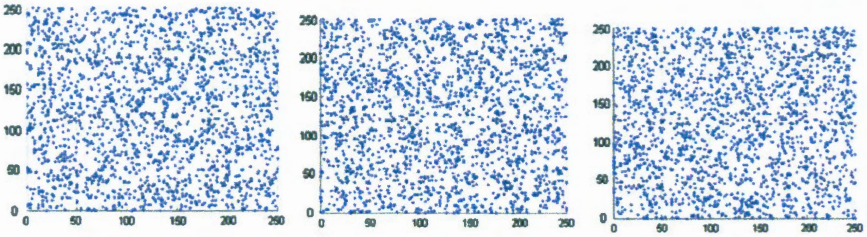


Figure 3-20 correlation of neighboring pixels of the encrypted image in the vertical horizontal and diagoanl directions

3.6.3 Robustness analysis

In order to test the robustness of the proposed algorithm against data loss and data noise attacks, we will apply a Salt&Pepper attack, white square cut and black square cut on several encrypted images and then we calculate the restoring ability of the encrypted images, Table 3-4 and Table 3-5 contains the restoring ability after applying Salt&Pepper noise attack , white square cut and black square data loss attack respectively.

Test Image	Size	Noise Salt&Pepper	PSNR dB
Baboon	512x512	1%	29.1197
		2%	26.1803
		5%	22.847
		10%	19.9142
Barbara	512x512	1%	29.1294
		2%	26.0186
		5%	22.7702
		10%	19.8815
Lena	512x512	1%	28.9243
		2%	26.0334
		5%	22.6630
		10%	19.7973
Pepper	512x512	1%	28.5041
		2%	25.5770
		5%	22.1864
		10%	19.2623

Table 3-4 The restoring ability after applying different levels of Salt&Pepper noise

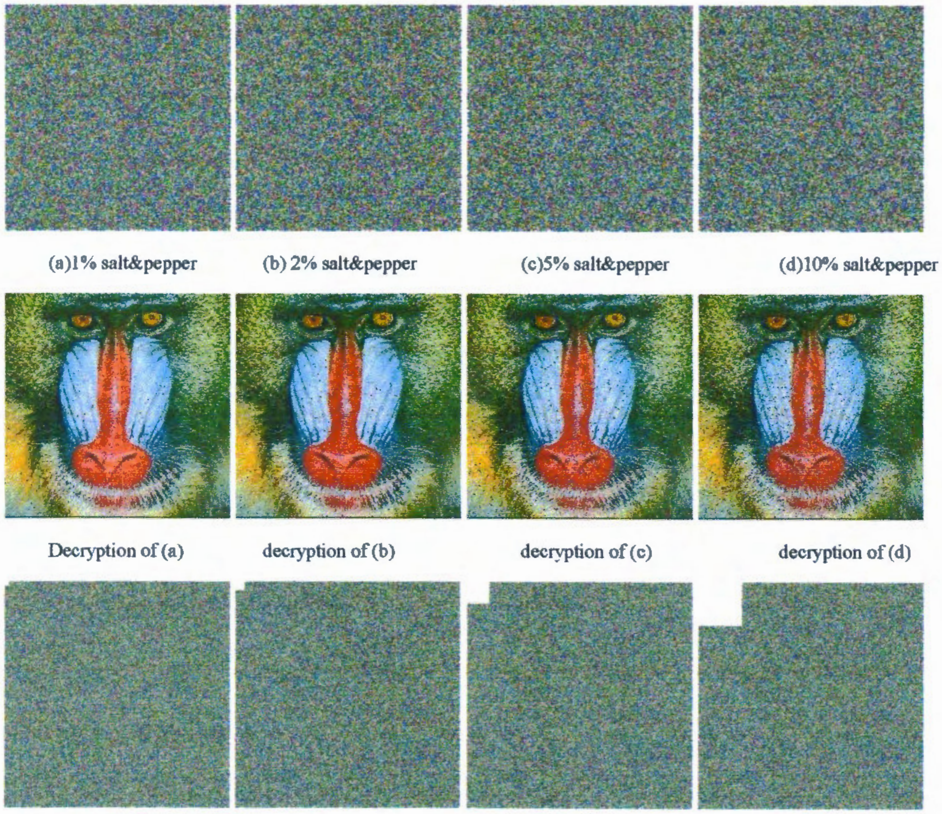
As we can see, the image is almost the same after applying less than 2% of Salt&Pepper noise because the PSNR value is almost 30. Moreover, the image still has an acceptable quality (PSNR is up to 19) and it still keeping its visual content even if 10 % of its data is contaminated by Salt&Pepper noise or it was a subject to data loss of 100x100 block size

Image name	Size	Data loss type	Block size	PSNR
Baboon	512x512	White Square cut	10x10	43.5048
			20x20	37.7003
			50x50	29.9762
			100x100	24.0076
		Black Square cut	10x10	44.3207
			20x20	38.2238
			50x50	25.1333
			100x100	24.0682
Barbara	512x512	White Square cut	10x10	43.7195
			20x20	37.7751
			50x50	29.8807
			100x100	24.0179
		Black Square cut	10x10	43.6133
			20x20	37.9364
			50x50	25.1277
			100x100	24.0027

Lena	512x512	White Square cut	10x10	43.8814
			20x20	37.6247
			50x50	29.9207
			100x100	23.8959
		Black Square cut	10x10	43.5978
			20x20	37.8592
			50x50	25.0417
			100x100	23.9054
Pepper	512x512	White Square cut	10x10	42.7315
			20x20	36.8522
			50x50	29.1542
			100x100	23.4313
		Black Square cut	10x10	43.6425
			20x20	37.5457
			50x50	24.5568
			100x100	23.2886

Table 3-5 The restoring ability after applying data loss

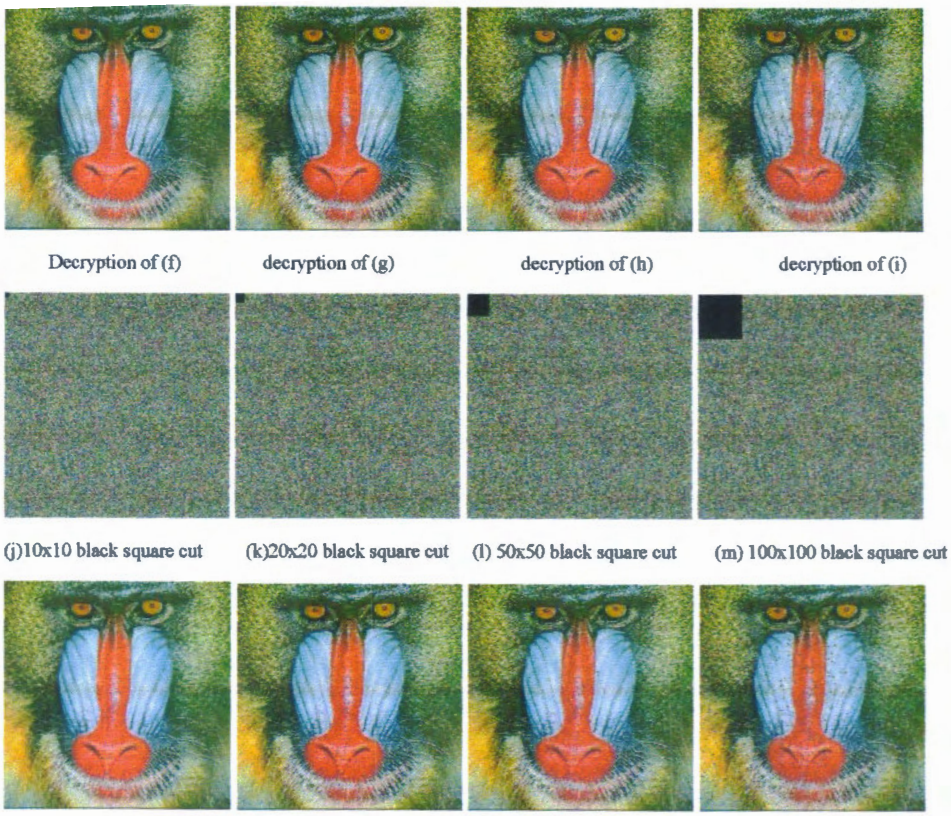
As we can clearly see the in the following images, the images is almost the same after applying less than 2% of Salt&Pepper noise Moreover, the image still has its visual content even if 10 % of its data is contaminated by Salt&Pepper noise or with a square data loss of 100x100 in size



(a)1% salt&pepper (b) 2% salt&pepper (c)5% salt&pepper (d)10% salt&pepper

Decryption of (a) decryption of (b) decryption of (c) decryption of (d)

(f)10x10 white square cut (g)20x20 white square cut (h) 50x50 white square cut (i) 100x100 white square cut



Decryption of (f) decryption of (g) decryption of (h) decryption of (i)

(j)10x10 black square cut (k)20x20 black square cut (l) 50x50 black square cut (m) 100x100 black square cut

Decryption of (j) decryption of (k) decryption of (l) decryption of (m)

3.6.4 **Speed analysis**

Test Image	Size	Execution Time	Decryption Time
Baboon	512x512	0.219 Sec	0.250 Sec
Barbara	512x512	0.281 Sec	0.219 Sec
Lena	512x512	0.219 Sec	0.203 Sec
Pepper	512x512	0.188 Sec	0.218 Sec

Table 3-6 Encryption time of several test images

Obviously, the proposed cryptosystem is very fast, and as a result it can be used to encrypt images of real time application.

3.7 Comparative study:

In this section, we will perform a comparative study between the different possible ciphers, in other words, we will keep the same structure of the ciphers, and we will change the chaotic PRNG and then we will calculate the different performance metrics and we will compare between them based on the results of the different metrics.

The existing PRNGs are the following:

— Logistic PRNG.

— MGF PRNG.

— MGFLogistic PRNG.

— LogisticMGF PRNG.

The following results are obtained using the baboon.JPG test image of size 512x512.

Test image	Metric	Used PRNG			
		Logistic PRNG	MGF	LogisticMGF	MGFLogistic
Baboon.jpg	Key space	2^{96}	2^{99}	2^{146}	2^{146}
	Key sensitivity (CDR)	99.98%	99.99%	99.999%	99.98%
	Entropy	7.983179	7.998	7.998	7.999
	NPCR	99.5704%	100%	99.99%	99.99%
	UACI	28.797%	30.12%	32.95%	33.95%
Barbara.jpg	Key sensitivity (CDR)	99.57%	99.615%	99.721%	99.98%
	Entropy	7. 7.983	7.999	7.998	7.999
	NPCR	99.57%	99.619%	99.721%	99.99%
	UACI	28.797%	30.043%	34.262%	32.95%
Lena.jpg	Key sensitivity (CDR)	99.998%	100%	100%	99.999%
	Entropy	7.993	7.995	7.998	7.998
	NPCR	99.998%	100%	100.0%	99.99%
	UACI	33.217%	32.101%	33.95%	32.93%

We can clearly see from the previous table that the results of performance and security analysis of the last two other ciphers (that are based on the combination between MGF and the Logistic map) are much better compared with the first two ciphers (which are based on only on the logistic or the MGF).

The first two ciphers have a not enough key space (less than 2^{100}) and as a result they often will be subject to brute force attack, so we can say that these ciphers are not secure enough, the last two one in contrast are more secure, and both have excellent performance results: both have a very sensitive secret key with a large key space (2^{146}) and both produces random images having an almost ideal entropy (close to 8) and a UACI close to the ideal value (33.33%) and an encryption rate sometimes equals to 100%.

3.8 Conclusion

In this chapter, we have proposed a new approach an image encryption scheme which is the combination between generating functions and chaotic functions, also we have proposed an image encryption scheme which is based on a modified generating function of Lucas balancing numbers and the logistic map and we have proved the high performance and high security of our proposed scheme using simulation

CONCLUSIONS

Conclusions

Nowadays, Visual content is the most used type of data, digital images and videos are stored and transmitted over computer networks and their security have become a problem because of the non-suitability of the traditional encryption schemes like AES to encrypt high correlated data, and one of the promising solutions of this problem is using the phenomena of chaos in the encryption because of special characteristics like sensitivity to initial conditions and ergodicity.

In this master thesis, we shaded the light on chaos-based image encryption schemes, first we presented the generalities in the three related components: images, cryptography and chaos, after that we have proposed a stat of the art of this area.

In the light of the previously mentioned study, we first proved the chaotic behavior in the generating function of Lucas balancing numbers, then, we enhanced the characteristics of this generating function like the chaotic range and the uniformity of the output. after that we provided an image encryption scheme based on the new proposed function.

The experimental results showed that the proposed algorithm is a fast algorithm and has a high level of security and performance, the key space is large enough, which makes brute force attacks infeasible. Therefore, the encrypted image histogram is very uniform, as a result, histogram attacks are not possible, in other words, the attacker cannot extract any information from the histogram of the encrypted image. Also the proposed algorithm produces encrypted images having an entropy close to the ideal value 8 and a low correlation coefficient between adjacent pixels near to 0.

Future works:

- Extend this cryptosystem to support other color models: because this actual is designed for only RGB color images.
- Make comparisons of the results of algorithm with other recent works and using other test images.
- Look for other chaotic ranges where the proposed chaotic function produces random and uniform numbers, this is because the proposed chaotic function has an infinite chaotic range, and as a results, other chaotic ranges are likely exists.
- improve the proposed modified generating function by combining it with other chaotic systems, this may result better chaotic functions.
- Investigate the chaotic behavior in other generating functions.
- Extending the proposed algorithm to encrypt the other type of visual content: videos.

BIBLIOGRAPHY

Bibliography

[1] "Digital_image," [Online]. Available: https://en.wikipedia.org/wiki/Digital_image.

[2] k. A. Hunt, The art of image processing with java, CRC Press, 2010.

[3] B. Wilhelm and B. M. J, Digital image processing an algorithmic introduciton using java, Springer, 2016.

[4] B. Furht, E. Akar and W. A. Andrews, "Digital Image Processing: Practical Approach," Springer, 2018.

[5] "JPEG," [Online]. Available: https://en.wikipedia.org/wiki/JPEG.

[6] S. Vaudenay, A classical introduction to cryptography :Applications for Communications Security, Springer, 2005.

[7] W. Stallings, Cryptography and Network Security: Principles and Practice, Sixth Edition, 2014.

[8] J.-P. Aumasson, serious Cryptography A Practical Introduction to Modern Encryption, No Starch Press, 2018.

[9] "Data_Encryption_Standard#Brute-force_attack," [Online]. Available: https://en.wikipedia.org/wiki/Data_Encryption_Standard#Brute-force_attack.

[10] "AES_instruction_set," [Online]. Available: https://en.wikipedia.org/wiki/AES_instruction_set.

[11] D. R. Stinson and M. B. Paterson, Cryptography theory and practice frouth edition, CRC press, 2019.

[12] L. Kocarev and S. Lian, Chaos-Based Cryptography, Springer, 2011.

[13] M. Asgari-Chenaghlu, M.-A. Balafar and M.-R. Feizi-Derakhshi, "A novel image encryption algorithm based on polynomial combination of chaotic maps and dynamic function generation," *Signal Processing,* 2018.

[14] E. Yavuz, "A novel chaotic image encryption algorithm based on content sensitive dynamic function switching scheme," *Optics and Laser Technology,* 2019.

[15] Z. Tang, Y. Yang, S. Xu and X. Z. Chunqiang Yu, "Image Encryption with Double Spiral Scans and Chaotic Maps," *Security and Communication Networks,* 2019.

[16] H. Prasetyo, "A New Image Encryption Technique Using Simple Chaotic Maps," *IEEE,* 2018.

[17] C. Pak and L. Huang, "A new color image encryption using combination of the 1D chaotic map," *Signal Processing,* 2017.

[18] H. Prasetyo, "A new image encryption scheme based on confusion and diffusion using an enhanced skew tent map," *IEEE,* 2018.

[19] Z. Hua, B. Zhou and Y. Zhou, "Image content-based encryption algorithm using high-dimensional chaotic system," *International Symposium on Nonlinear Theory and its Applications,* 2015.

[20] D. Sravanthi, K. K. Patro, B. Acharya and S. Majumder, "A Secure Chaotic Image Encryption Based On Bit Plane Operation," *Advances in Intelligent Systems and Computing,* 2018.

[21] J. Zhang, D. Fang and H. Ren, "Image Encryption Algorithm Based on DNA Encoding and Chaotic Maps," *Mathematical Problems in Engineering,* 2014.

[22] E. Rasul, A. A. Hanan, I. I. Fauzi, A. Ayman and L. Malrey, "Image encryption using a synchronous permutation-diffusion technique," *Optics and Lasers in Engineering,* 2017.

[23] X.-Y. Wang, Y.-Q. Zhang and X.-M. Bao, "Anovel chaotic image encryption scheme using DNA sequence operations," *Optics and Lasers in Engineering,* 2015.

[24] J.Wu, X. Liao and B. Yang, "Image encryption using 2DHenon- Sine map and DNA approach .," *Signal Processing,* 2018.

[25] P. Nayak, S. K. Nayak and S. Das, "A Secure and Efficient Color Image Encryption Scheme based on Two Chaotic Systems and Advanced Encryption Standard," 2018.

[26] J. Wu, X. Liao and B. Yang, " Color image encryption based on chaotic systems and elliptic curve ElGamal scheme," *Signal Processing,* 2017.

[27] X. Wu, H. Hu and B. Zhang, "Parameter estimation only from the symbolic sequences generated by chaos system," *Chaos Solitons Fractals,* p. 359, 2004.

[28] A. Behera and G. K. Panda, "On the Square Roots of Triangular Numbers.," *The Fibonacci Quarterly,* pp. 37, 98-105, 1999.

[29] F. S and P. A., " On k- Fibonacci sequences and Polynomials and their derivatives," *Chaos, Sulutions & Fractals* , vol. 39, pp. 1005-1019, 2009.

[30] H. VE, "Fibonacci and Lucas numbers," *Palo Alto, CA: Houghton-Mifflin,* 1969.

[31] "jFoenix," [Online]. Available: http://jfoenix.com/documentation.html.

[32] "controlsFX," [Online]. Available: https://github.com/controlsfx/controlsfx.

[33] Z. Huay, B. Zhouz and Y. Zhouy, "Image content-based encryption algorithm using high-dimensional chaotic system," *International Symposium on Nonlinear Theory and its Applications,* 2014.

[34] G. Layek, "An Introduction to Dynamical Systems and Chaos," Springer, 2015.

[35] L. Xu, X. Gou, Z. Li and J. Li, "A novel chaotic image encryption algorithm using block scrambling and dynamic index based diffusion," *Optics and Lasers in Engineering,* 2017.

[36] H. Jinwen, L. Rushi, W. Shouhua and L. Xiaonan, "An Integrated Chaotic System with Application to Image Encryption," 2017.

[37] Z. Yicong, LongBao and C. PhilipChen, "A new 1D chaotic system for image encryption," *Signal Processing,* 2014.