Democratic Republic of Algeria

Mohamed Boudiaf – M'sila University

Faculty of Mathematics and Computer Science

Computer Science department

Thesis

By

Ouadah Kheir eddinne

Berra Ilyes Zine Abidinne

About

Cryptanalysis and Improvement on an Image Encryption Algorithm Design Using a Pseudo Random generator

Supervisor:

Mohamed Benouis

Jury members:

Baya Chalabi …… University of M'sila

Fares Mezrag …… University of M'sila

ACKNOWLEDGEMENT

Berra Ilyes Zine Abidinne

First of all, thanks and praise be to God Almighty for my success during my research work to complete the research successfully. I would like to express my deep and sincere gratitude to my research supervisor, Dr. Mohamed Benouis, Dr. Abdessettar Ghemougui and Dr. Rafik Hamza for giving me the opportunity to conduct research and provide invaluable information throughout this research. I was inspired by their dynamism, vision, dedication, motivation and their own diligence. They taught me how to conduct research and present research work as clearly as possible. I am very grateful for what they provided to me. I would also like to thank them for his friendship, sympathy, and good sense of humor. We are very grateful to my parents for their love, prayers, care and sacrifices to educate me and prepare me for my future. I also express my thanks to my brother for their support and valuable prayers. My special thanks to my friends Abdou Delami , kadour djkam , benchelli abdou and Djihad dijiab My Aunt Samira for their keen interest in completing this letter successfully

Ouadah Keir eddinne

First, I would like to thank each and every one who stood with me in my journey, starting with my parents who supported me from the beginning, they did so much for me I cannot list it or return even a little bit of their favor, so, I would like to dedicate their this thesis to them. Next, I would like to thank my supervisor Dr.Mohamed Benouis who was with us since the beginning and worked with us like it is his thesis, whenever we asked him, he was always ready to support us by his encouraging words, and I would like to thank Dr.Abdessettar Ghemougui who helped us in the beginning of our work along with our supervisor Dr. Mohamed Benouis, they were so kind and been with us whenever we called them, also, I would like to thank Dr. Rafik Hamza, the one who developed the algorithm and we shared his ideas to improve our work, he was so kind to give us some of his valuable time to give us some advices that helped us a lot in our work, so, I would like to dedicate this thesis to them and to all my friends and fellow students who supported me in any way , and wish to everyone the best in their lives.

Contents

List of figures:

List of tables:

List of equations

ABSTRACT

In the worldwide, the people exchange a massive data through IP network (wireless or wirefire). Image is mostly found massively in the traffic network as logo, human picture, medical image, satellite image and so on. Image security is a major issue in traffic networks since image travel over the naturally exposed unknown channel where malicious attackers may get access to critical information. To deal with this issue, the devices are mostly making some modification on the data to hide or protect against the man-in-the-middle attack. However, existing encrypted algorithms, such as AES and DES are not sufficient for image encryption due to the massive data capacity and high correlation between pixels in image files. This work proposes a data security approach with less computational and response times based on a modified version of pseudo random function. The pseudo random function has been applied on the bit plan instead of pixel values in which to make it more secure. The proposed approach has been assessed in terms entropy, correlation. Furthermore, it has also been analyzed in terms of encryption/decryption time, computation time, and key generation time for different sizes of data. The comparative analysis with the state-of-the-art encryption algorithms shows that the proposed approach performs better in most of the case

General introduction

1.1 State of art:

The images today present one of the most important information exchanged over internet network where people may make several tasks such as QR scanner, signature, logo or picture in social medias, marketing, and so on. However, aspect security issues have long been an important factor that restricted the exchanging of those data. Especially in the context of the confidentiality of the data content, how to protect either in the computer network or in the network is the hot topic and direction of the research in the field of network security and information security. Among them, digital image has become the important way to interact the human to human, human to machine or machine to machine. To secure these interactions, the security protection of digital images is becoming a big concern of many governments, people and companies. To deal with this issue, there are a lot of encryption functions proposed in recent years to guarantee the security of digital images. The encrypted functions and its properties of cryptography can be related to the fundamentals of random function like sensitivity to initial conditions, probability density function and static orders. Unlike the current encryption algorithms, the classical encryption methods provided a limit of key and straightforward encryption functions, thus mostly argued their efficiency security and key space length. In past couple of decades, a new generation of encryptions method have been appeared on the digital image, audio and video, which adopted different types of pseudo random functions to generate efficient and cryptographic random sequences of bits to be used as encryption keystreams in stream or block cryptosystems to resist any external threat from attacker.

So, any proposed encryption algorithm has to meet the following aspects: authentication, confidentiality, integrity, and non-repudiation. Although the performance of the exiting encryption schemes, many of them are have been analyzed and argued in term weakness and be not resist against hackers. In this way, we can say any efficient encryption algorithm, must be almost resister against all cryptanalysis attacks.

To behavior the performance of encryption algorithms, there are many points must be taken as specific features for any kind of domain and the content data, such as structures of the cryptographic key, the structures of encryption scheme, and their combination. To

get a strong key, we have to build an efficient encryption function that provides strong secret keys with high sensitivity to bit small change. Many researchers claim that the pseudo-random number generator may generate a key is high sensitivity toward initial condition and high sensitivity in the overall structure.

Motivated by the performance of pseudo random functions on the image digital encryption, we are intending to improve some existing PRNG used for digital image encryption. To do this work, we have divided into three chapters. The first chapter, we present a highlight about digital images, by introducing how is manipulated and then we discuss the most fundamental proprieties of image. The second chapter is about the security concepts of the digital images. After we explain how to perform the security concept on the digital images. In the last chapter, we present our contribution, which is made by the hybridization of two kind algorithms, one based on PRNG and the second is based on the bit-plain slicing. The proposed algorithm has been assessed on well-known digital images and also compared with some exiting start of the art encryptions algorithms.

Finally, we present a general conclusion and some perspectives and ideas that open new avenue to improve the current systems in the future works.

Part 1
State of the art

2

Digital images

A picture is worth a thousand word…

2.1 Introduction

Digital images are everywhere, especially in our days, from the digital camera to the smartphone camera, sending and receiving images is an everyday task. In this chapter we are going to see what is a digital image and the types of digital images to get a basic understanding about it, then we are going to see some image processing operations in order to get a basic understanding on how to process and perform some operations on these digital images.

2.2 Digital Images

The images can convey the essence of a topic more effectively than words and description as the English idiom sentence says" an image is worth a thousand words", the essential component of an image is the pixel, an image is a matrix of pixels. Mathematically an image is two-dimensional function of integer coordinates N × N, these coordinates refers to an image element values (pixels) P, such that

$$I(u, v) \in P \text{ and } u, v \in N$$

A digital image can be represented as an array of values contains the value of each pixel, with a length equals to the dimensions of the image (array length = width*hight).

2.3 Images types

2.3.1 Based on coloring

Based on coloring images are divided into three types.[1]

2.3.1.1 Binary image:

A binary image is an image that each pixel value is either 1 (white) or 0 (black) [1].

Figure (1.1) binary image shows a binary image on the left, and on the right a representation of a zoom of a small part of the image to the pixel level to get a good idea how the image is composed.

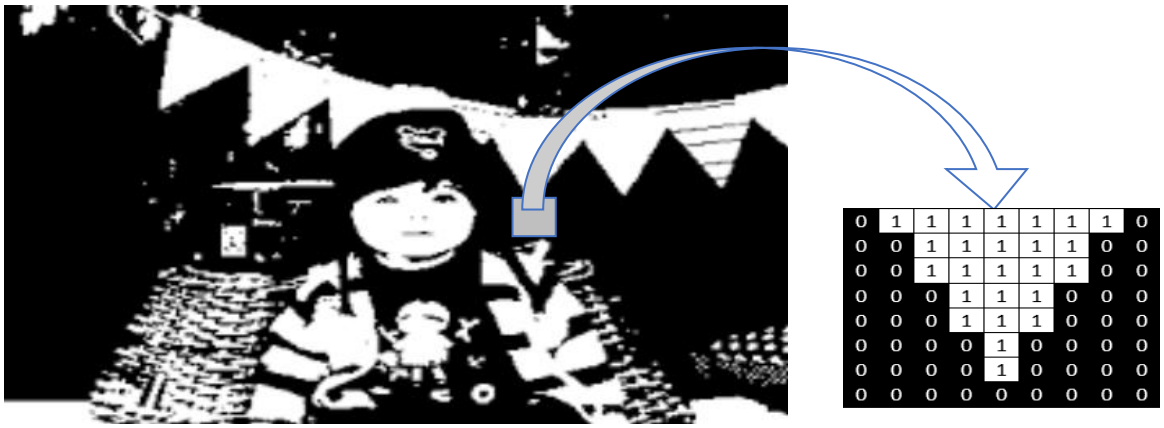


Figure (1.1) binary image

2.3.1.2 Grayscale image

In Our work, we have studied the of 8-bit plan, so the value of each pixel is between 0 and 255 (total white to total black), meaning 256 shades or degree of gray color, that is $\mathbf{2^8} = 256$ [1].

Figure (1.2 represent an image of Lena in grayscale while Figure (1.3 shows all possible 256 bins of gray from image.

Figure (1.2) Lina Grayscale

Figure (1.3) Grayscale in 8-bit

2.3.1.3 True color image

Unlike the grayscale image space color, the RGB image codes each pixel by three values, red, value and the blue value [1].

Figure (1.4 shows Lina in RGB mode, while Figure (1.5 shows a table of some popular colors used in digital images and their pixel range.

Figure (1.4) Lina RGB

Color name	RGB triplet	Color
Red	(255, 0, 0)	
Lime	(0, 255, 0)	
Blue	(0, 0, 255)	
White	(255, 255, 255)	
Black	(0, 0, 0)	
Gray	(128, 128, 128)	
Fuchsia	(255, 0, 255)	
Yellow	(255, 255, 0)	
Aqua	(0, 255, 255)	
Silver	(192, 192, 192)	
Maroon	(128, 0, 0)	
Olive	(128, 128, 0)	
Green	(0, 128, 0)	
Teal	(0, 128, 128)	
Navy	(0, 0, 128)	
Purple	(128, 0, 128)	

Figure (1.5) table of some pixel colors values

2.3.2 Based on the nature

Images are divided into two types:

2.3.2.1 Vector graphics (images)

This type of images can be represented mathematically (straight lines, circles, points, ...) because it is composed of geometric shapes, as shown in Figure (1.6, which is a vector image with extension ’eps’[s1] you can scale it without losing any information, thus, it very useful when you are working on applications that frequently require a deep manipulation and adjusting the object content.

A simple example of vector graphics that is used in most people’s daily life is the graphic shapes in Word, no matter how much you scale its size, it will keep its details.

Figure (1.6) A zoom for a part of a vector image after being scaled

This type of graphics has its own negative points, for instance some manipulations such as color changes are not easy and difficult on an area of an object, on a single object, or on a group of objects as shown in the zoomed part on the right of Figure (1.6) A zoom for a part of a vector image, but in the positive way a vector file is much more compact than a bitmap file. Its size relies on the content of the image, not to its resolution, however, it took a high resource on the digital image (memory) to displayed any object.

2.3.2.2 Matrix graphics (images)

The matrix representation is the most popular type used in our days, and the most image encrypted algorithms are working on this type, this latter represents by a set of points(pixels), each one has its intensity (values), and also has specific spatial frequency information. In the literature, we found many formats of these images, so we listed at least the 5 well used in digital image:

- BMP: Windows BitMap
- TIFF: Tagged Image File Format
- JPEG: Joint Photographic Expert Group

- GIF: Graphics Interchange Format
- PNG: Portable Network Graphic

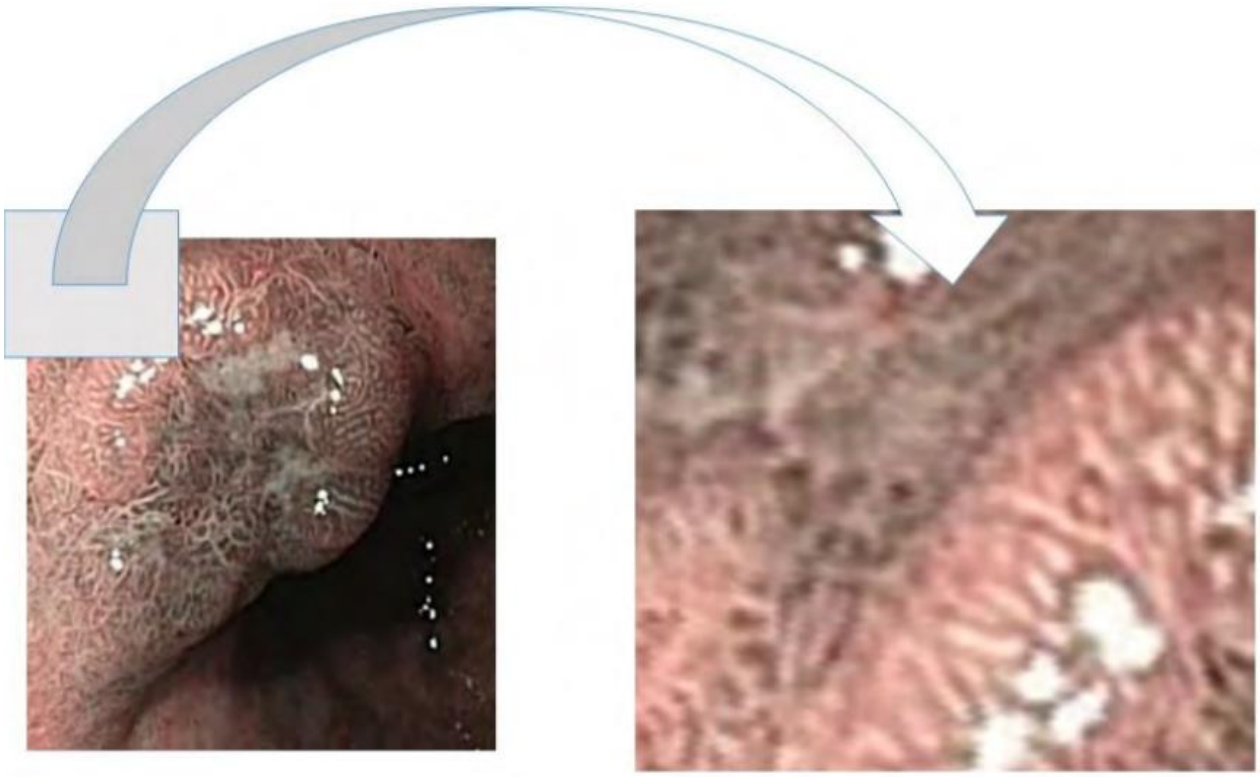


Figure (1.7) A zoom for a part of a matrix image

2.4 Digital image characteristics

2.4.1 Bit depth:

Bit depth is determined by the number of bits to represent each pixel in the image which translate to color or a grayscale value, so in binary images we have either black or white, because all binary images are 1-bit depth, here we are going to work with are 8-bit depth, in other words, we have 256 possible value (in RGB images, each pixel is coded with 24 bit depth , so a total of 2^{24}-bits possible value.

2.4.2 Image size and resolution

The size of an image is computed directly from the number of pixels along the width M (number of columns) and height N (number of rows) of the image matrix I [2].

In general, the resolution means the amount of number pixel used to represent the details, edge, foreground and background of image [3].

The pixel density, it can be measured using pixels per inch (PPI), dots per inch (DPI), or pixels per centimeter (PPCM). Those are common terms used to express measurements of the resolution for digital images.

Figure (1.8) Different Pixel Resolutions for logo of university of Mohamed boudiaf – M'sila. (a) a block of [512,512] pixels, (b) a block of [256, 256] pixels, (c) a block of [128, 128] pixels, (c) a block of [64, 64] pixels, (e) a block of [32, 32] pixels.

2.4.3 RGB channels and space color

The basic space color of image refers to the RGB space, which each pixel represents by the combination of three channels: R, G and B. The Figure (1.9) (a) Original image, (b), (c), and (d) components Red, Green and Blue, respectively. depicts an image with RGB space and Figure (1.10) RGB image colour ordering. (a) RGB colour image in component ordering. (b) RGB-colour image using packed ordering. depicts the three channel separately.

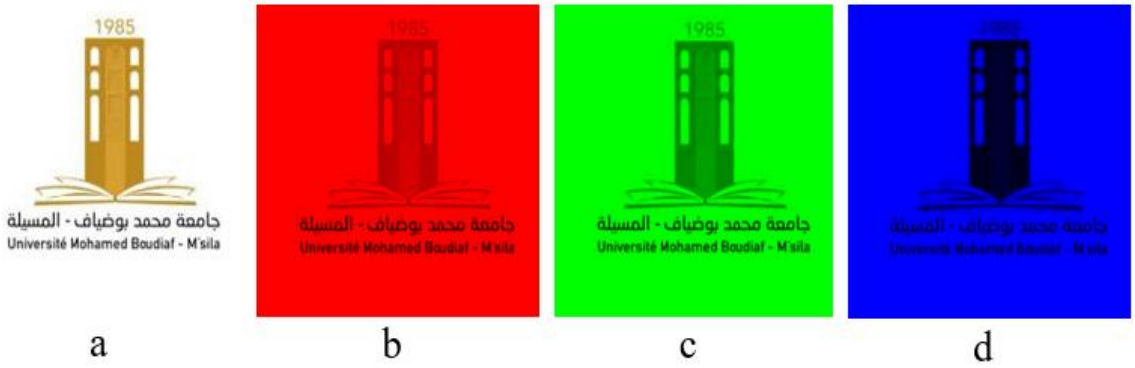


Figure (1.9) (a) Original image, (b), (c), and (d) components Red, Green and Blue, respectively.

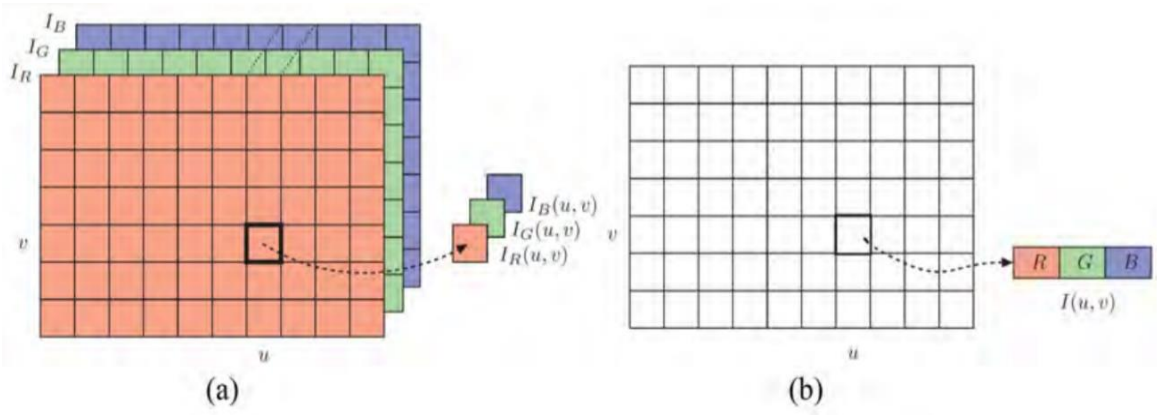


Figure (1.10) RGB image colour ordering. (a) RGB colour image in component ordering. (b) RGB-colour image using packed ordering.

Figure (1.10) RGB image colour ordering. (a) RGB colour image in component ordering. (b) RGB-colour image using packed ordering. (b) shows an example matrix representation of an image, the pixel I at the position (u,v) is composed of three values, one for R (red), and one for G (green), and the last for B (blue), so for example:

if $I_{(u,v)}$ = [67,67,67] then: we can get the three channels as:

$I_{R(u,v)}$ = [67,0,0] , $I_{G(u,v)}$ = [0,67,0] , $I_{B(u,v)}$ = [0,0,67]

2.5 Image processing operations

In real time application, the image may manipulate by different rules of matrix operations like addition, subtraction, multiplication and division. On other hand, we can find different format of matrix as binary image (in a case of pixels are 0 or 1), a gray image (in a case of one matrix), color image, RGB image (in a case of Multi-matrices) and image palette or multi-color (tensor matrix) [3].

In addition, the image can be manipulated by logic operator and filters in order to extract or enhance its quality regarding and also very crucial for image security applications. For example, image encryption is a manipulation from a plain image to a random image or unclear image that cannot lead us visually to the original ones.

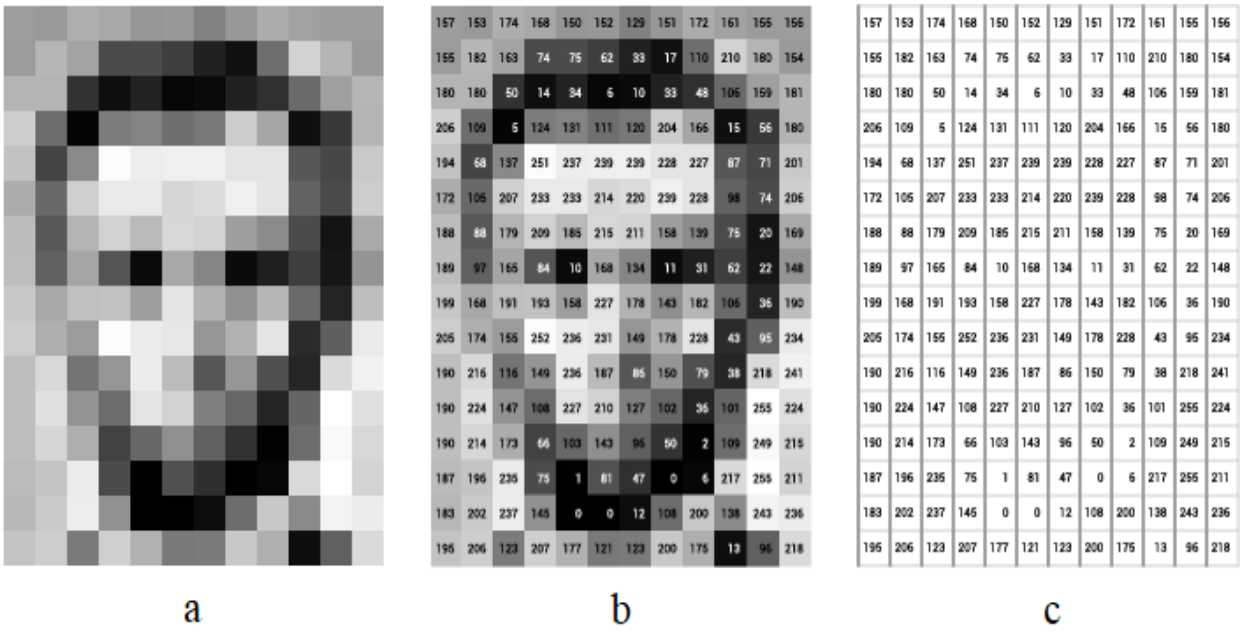
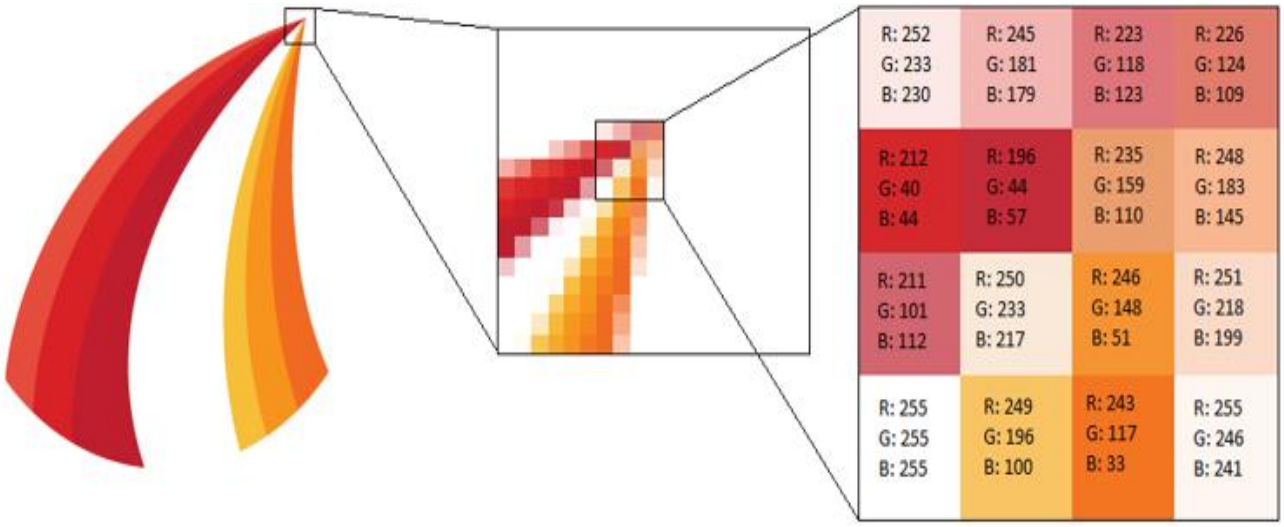


Figure (1.11) (a) zoomed image of a person's face, (b) same image with the value of each pixel, (c) the matrix of pixels values of the image (a)

Figure (1.11) (a) zoomed image of a person's face, (b) same image with the value of each pixel, (c) the matrix of pixels values of the image (a) (a) shows the face of a person, (b) shows the same image with the value of each pixel on top of it, (c) shows each pixel value of the image (a) organized in a matrix, naturally with the same dimensions of the original image.

In case of an RGB image, they are going to split the image into three matrices, red, green and blue like as is explained in pervious section.

Figure(1.12) the representation of pixel values of a zoomed part of an RGB imageshows an example of pixel values representation of small zone of RGB image.



Figure(1.12) the representation of pixel values of a zoomed part of an RGB image

Figure (1.13) Encryption of the symbol of university of Mohamed Boudiaf -M'sila shows an example of three images. The first image is the original, the second seems fully noisy, which we have done a number of manipulation of pixels values through a specific function, the third image is the recovered image after having applying the inverse function [29].

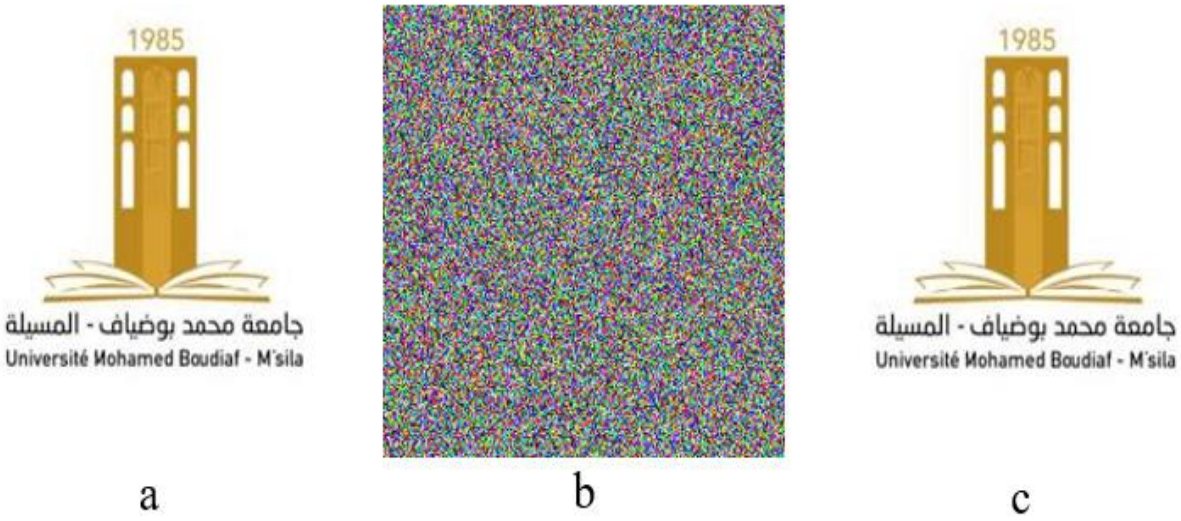


Figure (1.13) Encryption of the symbol of university of Mohamed Boudiaf -M'sila

2.6 conclusion

In this chapter, we explain some basic proprieties about digital image and how to process it, it’s very important to well cover the image propriety in order to well protect it and to develop an efficient encrypted image without destroying its content.

3

Security Mechanisms

Security is not a product but a process

Bruce Schneier

3.1 Introduction

Now, the digital images are being used in different area of our life and it's important to secure it against any malicious, or attack. The cryptography is an important field that handle many algorithms that can used to secure the digital images. In this chapter, we are going to give a big picture about image processing and explaining how to apply the security mechanisms on the digital image.

3.2 Secure Digital Images

As we saw in the previous chapter, an image is represented by a set of matrices, the symmetric cryptography offers a suitable algorithm can be directly performed on the image as matrix (i.e, AES, DES and so on) [4].

As we explained in the previous chapter, the encryption and decryption of the logo of our university was made by secret encryption key with a simple XOR operation between the image and the key, the main objective of cryptography is to make sure that the attacker cannot read the encrypted message or can know the encrypted function by finding the secret key. In the following section, we introduce the cryptography and cryptanalysis in the following sections and chapters.

3.2.1 Cryptology

Cryptology is the foundation of all information security; it basically addresses the security studies of information with cryptography and cryptanalysis. Mainly, cryptography and cryptanalysis are disciplines from cryptology [5]. So, we can say cryptography studies the structures of cryptosystems, while cryptanalysis studies how to break down these cryptosystems.

3.2.2 Security concepts

Key concepts of information security based on cryptology are mainly divided into four aspects.

- **Authentication**

Authentication is to verify the user identity, the receiver must know who is the true sender of the message, an intruder must not be able to take the true sender identity and pretend to be him. A good example would be copyright, in order to insure identity authentication. Protecting property rights of a digital image is most important in multimedia applications [6].

- **Confidentiality**

Confidentiality means only an authorized person is allowed to access and use data, meaning only those who have the right permission are allowed to access data and use it, a good way to ensure that encryption methods and mechanisms, in this example only authorized parties with the right permission can decrypt the data.

- **Integrity**

When we say that a digital object has "integrity", we mean that it has not been corrupted over time or in transition; in other words, that the receiving end have in hand the same set of sequences of bits that were sent to him. An intruder must not be able to send a false data to the authorized recipient.

- **Non-repudiation**

Non-repudiation is the assurance that someone cannot deny the validity of something. Ensuring authentication and integrity is the easiest way to accomplish non-repudiation.

3.2.3 Cryptography

Employing math equations to create means to ensure the security of information in communication. Cipher, hash, digital signature, key generation are all means purposes along with others such as authentication [7]. Cryptography is the science concerned with the study of secret communication. Encryption basically is some process or algorithm (known as a cipher) to make information hidden or secret. And to make that process useful, you need some code (or key) to make information accessible.

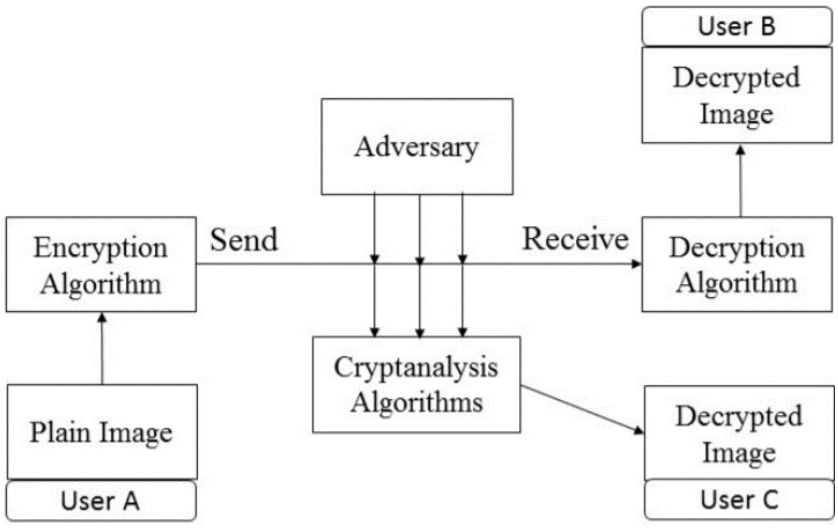
3.2.4 Cryptanalysis

Cryptanalysis focuses on analyzing algorithms, in image encryption it means analyzing the encrypted image and trying breakdown and examine the encryption structure. It should be impossible for anyone who is not authorized to decrypt the encrypted image.

Recently, some security issues have been presented with the encryption schemes [8,9,10,11], from three main reasons: key space, the structured algorithmic and their combination.

Figure(3, 1) Scenarios of cryptosystem/cryptanalysisshows a normal basic process of cryptosystem with two authorized users A and B, and an unauthorized user C or an attacker trying to crack the encrypted image without the secret key/keys.

According to Kerckhoff's principal "A cryptographic system should be secure even if everything about the system, except the key, is public knowledge." [12], the security of a cryptosystem should not depend on keeping the cryptographic algorithm secret [13]. Meaning that you should consider that the cryptosystem structure is known to the attacker, so, here the attacker will go after either the key or the cipher structure.



Figure(3, 1) Scenarios of cryptosystem/cryptanalysis

3.2.5 Probabilistic approach

A probabilistic approach means that the result and/or the way the output is obtained depends on chance, randomized encryption algorithms are defined as probabilistic algorithms.

3.2.6 Randomized algorithms

Randomized algorithms are algorithms that make random choices during their execution [14]. The first probabilistic encryption scheme was introduced in 1982 by Goldwasser-Micali [15].

In randomized encryption, the encrypted image should be different even if we use the same secret key and plane image. Randomization is used a lot by cryptographers to enhance their encryption process [16].

3.3 Encryption terminologies

The following points show the meaning of famous terms in image encryption.

- **Encryption algorithm**

Encryption Algorithm contains the steps of encrypting the plain images (clear form) into encrypted image (unclear form). The encryption of an image cannot proceed without a secret key.

- **Decryption algorithm**

Decryption Algorithm is the steps of inverting the operations of encryption algorithm, which decrypt the encrypted image (unclear form) back into the plain images (clear form). The success of decryption of an encrypted image cannot be achieved without the secret key.

- **Plain image**

Plain image is the original image before proceeding to encryption processes. Also, the original images can be defined as a plain Images or clear Images, and unencrypted images.

- **Ciphered image**

Ciphered image is the encrypted image which is a result of transforming the plain image to ambiguous and unintelligible image.

- **Decrypted image**

Decrypted image is a result of transforming the encrypted image to plain image. The decryption can lead to a decrypted image appear as the plain image, but not with same accuracy the original ones.

- **Secret keys**

A Secret Key is an array of data that are employed to encrypt or decrypt the digital images using cryptographic functions. The Secret keys can be symmetric or asymmetric (see the next sections for more details). Generally, the secret keys are employed to produce appropriate keys for the digital images (Encryption Keys). Note that in most cases of chaos-based image encryption, the initial values (called seeds) of the chaotic maps considered as secret keys in many works [17,18,19,20].

- **Encryption keys**

An Encryption key is a cryptographic algorithm that uses the secret key to generate appropriate keys for encrypt and decrypt the digital images. PRNG is well known method to produce these encryption keys for the digital images. Figure (3,2) Framework of generating the cryptographic keys shows the steps of using the secret keys to manufacture PRNG, the generated keys will be employed either for encryption or decryption.

-Pseudo random numbers generator

A pseudo random numbers generator (PRNG) refers to an algorithm that uses mathematical formulas to produce sequences of random numbers [21,22]. In most cases, the generated sequences are for image encryption algorithms (see chapter 4). A good example here is linear feedback shift register (LFSR) [23,24] which can be used as a PRNG too. The initial secret keys are required to generate the random sequences which are keys employed in many cryptographic purposes.

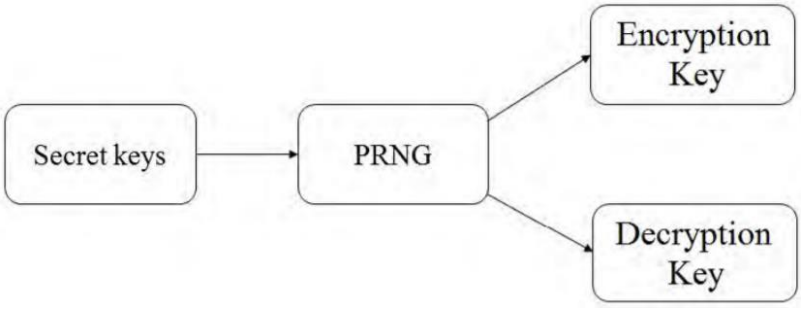


Figure (3,2) Framework of generating the cryptographic keys

3.4 Encryption types

Here, we present the types of encryption algorithms, based on the secret key structure, which can be divided into two categories.

3.4.1 Asymmetric encryption

Unlike the key generation in symmetric encryption, the asymmetric encryption uses two keys, public and private key. Mathematically, we can define the asymmetric encryption as follow:

$$\begin{cases} C = E(P,K_E) \\ P' = D(C,K_D) \end{cases} \qquad eq(1)$$

Here K_E is the encryption key and K_D is the decryption key.

Figure (3,3) Architectures of asymmetric encryption. shows the architectures of asymmetric image encryption based on the equation above, while respectively, P, E(), D(), P' are the original image, encryption algorithm, decryption algorithm, and decrypted image.

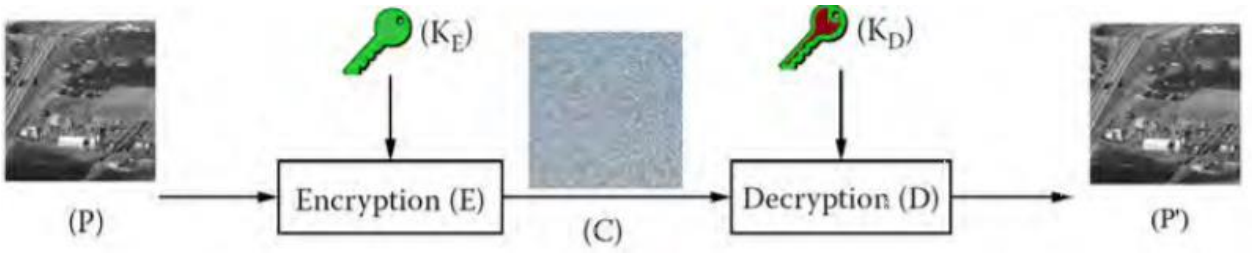


Figure (3,3) Architectures of asymmetric encryption.

3.4.2 Symmetric encryption

Here, one key is used to encrypt and decrypt the image, meaning that the decryption operation is symmetric to the encryption operation, mathematically presented as:

$$\begin{cases} C = E(P,K) \\ P' = D(C,K) \end{cases} \qquad \text{eq}(2)$$

Here, P, C, P', K, E(), and D() are the original image, encrypted image, Decrypted image, the secret keys, Encryption scheme, and Decryption scheme ,respectively.

The architecture of symmetric image encryption is shown in Figure (3,4) Architectures of symmetric encryption. based on the eq 2.

In this architecture, only the sender and receiver can know the encryption key no one else is allowed to know it.

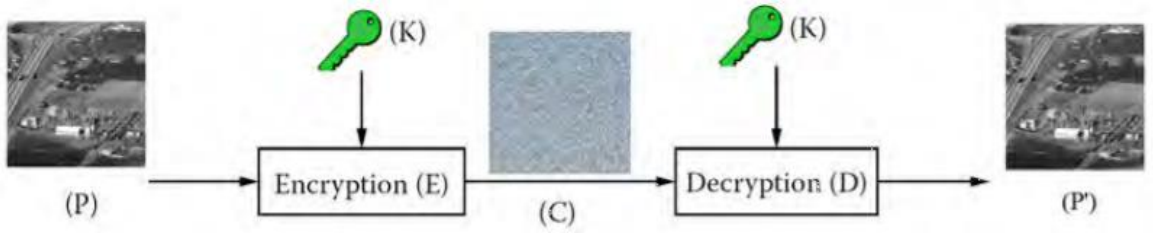


Figure (3,4) Architectures of symmetric encryption.

3.5 Substitution-Permutation Network

One of the most common symmetric cryptosystems is the substitution-permutation network. The network is comprised of a number of rounds of permutation, substitution [25].

In 1945, Shannon [26], proposed the most famous properties in encryption data: the confusion and diffusion properties. On other hand, the original image will be passed go through a number of rounds, in each round performs the confusion and diffusion operation on the pixel image [26] in order to ensure the ciphered image security [27].

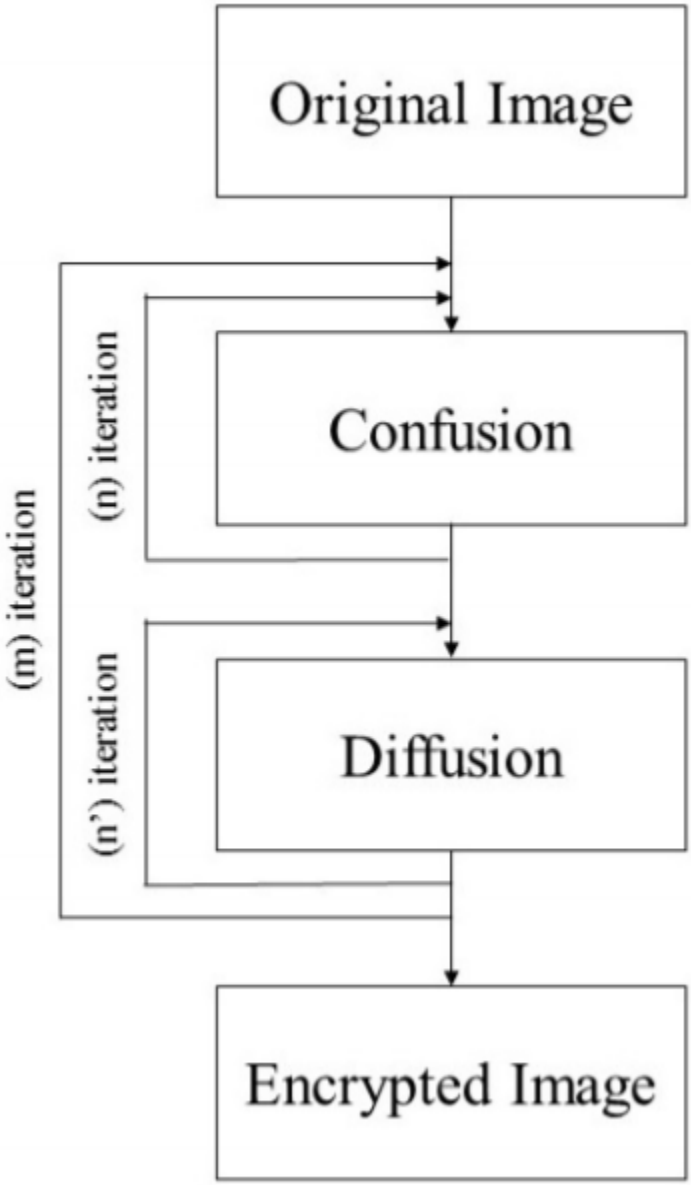


Figure (3,5) Confusion and diffusion algorithm

Figure (3,5) Confusion and diffusion shows the common steps of encrypting an image by applying the confusion and diffusion operators, where n and n' iteration number of each confusion and diffusion step, and m iteration for both steps.

- Confusion in image encryption means to take the plan image and change the pixels positions to get a scrambled image as shown in Figure (3,6) An instance of confusion algorithm of a pixels block..

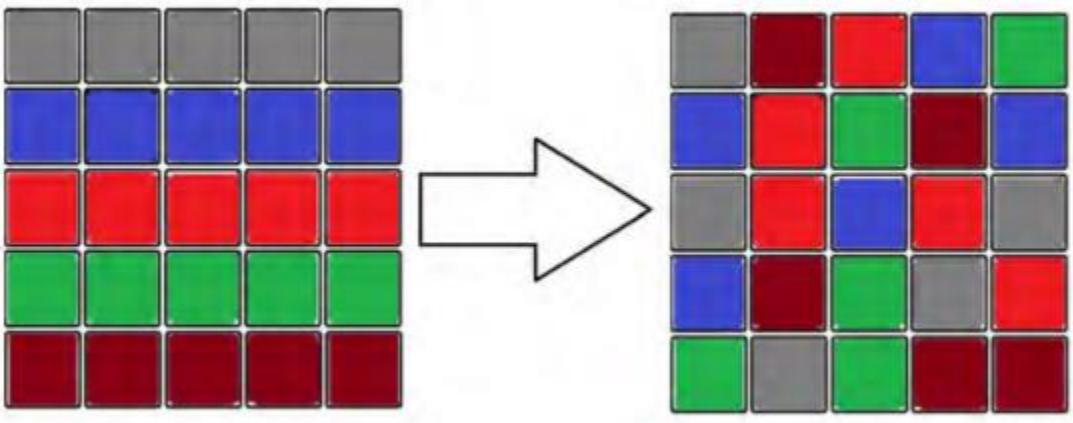


Figure (3,6) An instance of confusion algorithm of a pixels block.

- Diffusion in image encryption means that changing a single pixel of the plain image shall change many pixels of the ciphered image. For example, in Figure (3.7), the confusion step changed the values of many pixel which lead to well secure the image.

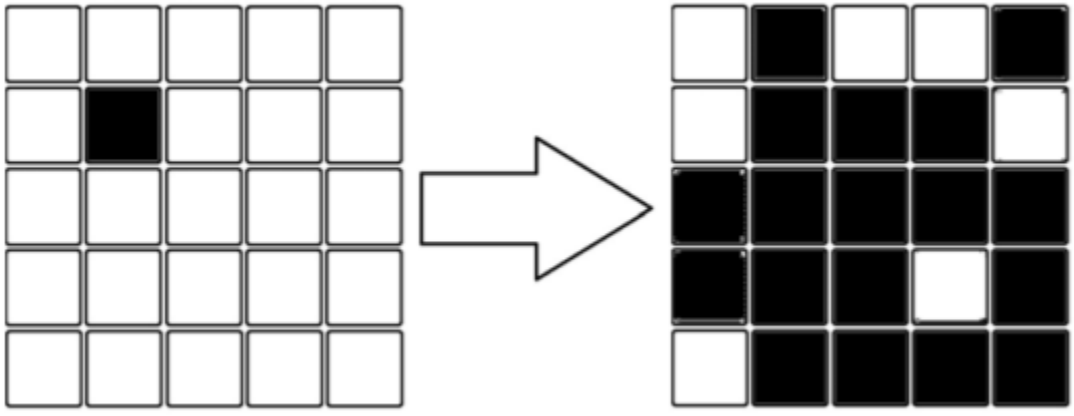


Figure (3,7) An instance of diffusion algorithm of a pixels block.

This technique is used in many encryption algorithms which may operate on the pixel or bit plan values [28].

3.6 Conclusion

In this chapter, we shown several mechanisms and techniques related to our studies which already used in several security applications. In this work, we have been focusing on the pseudo function generator which will be discussed in the next chapter.

4

Proposed Work

A picture is worth a thousand word, but after we are done you will be left speechless.

4.1 Introduction

As we have mentioned in the previous chapters the importance of encrypting digital images, especially in term key function generation to ensure some level of security, we are willing to propose an encryption algorithm for digital image encryption. Motivated by the pervious works [28,29], our proposed work combines the both advantages of chaotic maps and bit plan slicing to further provide higher security in image encryption.

The main contributions include in this work are as follows:

- We propose a secure image encryption algorithm based on Bitplane slicing and a secure pseudorandom generator. The Bitplane slicing computation converts the image from pixel level to bit level (i.e., each pixel coded by 8 bit). As a result, this operation provides an additional level of security to the image encryption algorithm.
- An enhanced efficient pseudorandom based on previous PRNG [29] is proposed in this paper to encrypt the digital images. The chaotic pseudorandom number generator algorithm is employed to produce the secret keys.
- A permutation between neighbor pixels is performed to reinforce the statistical performance of the proposed encryption algorithm. The permutation phase ensures higher confusion among image pixels and bit plans.

4.2 The pseudo random number generator

First, we are going to explain the behavior of the pseudo random sequence that will be used to generate the key space. So, we are using the Chen chaotic system introduced by Chen[30], This map is also called Lorenz-like systems, which means it is similar as the chaotic system Lorenz [31,32], The complex dynamical behaviors of Chen chaotic system can be elaborated in Figure (4,1) . Chaotic behavior of Chen’s system..

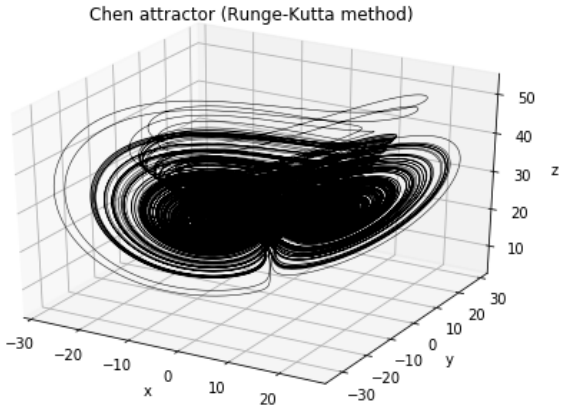


Figure (4,1) . Chaotic behavior of Chen's system.

The Chen chaotic system can be represented mathematically like this:

$$\begin{cases} x' = ay - ax \\ y' = cx - ax + cz - xz \\ z' = xy - bz \end{cases} \qquad eq(3)$$

Where x, y and z are the samples of this system while a, b, and c are the control parameters.

In our first experiment we took almost all parameters as mentioned in [29], meaning the control parameters: a = 35, b = 3, and c = 28, and the initial values $(x_0, y_0, z_0) = (-1.5, 0.6, 17)$.

To generate the pseudo random sequence, we use Runge-kutta step size 0.01, with iterating the chaotic system for n/3 or (1+n)/3 or (2+n)/3 times, where 'n' is the length of the pseudo random sequence we want to generate which is equal to the total number of pixels in the image we want to encrypt, for example Figure (4,3) Lena Grayscale) is a 512×512 grayscale image of Lena, meaning the total number of pixels is 262144. And we iterate the chaotic system for 87382 times as the following: (2+n)/3 = 2+ 262144/3 = 87382.

The result here is three lists x_i , y_i , z_i , each with the length of 87382, we call this number 'k'.

Based on a large number of experiments done in [29], we follow the following proposed coding Algorithm. The generated sequences are uniformly distributed.

$$\begin{cases} P(3 \cdot i) = \alpha \cdot m \cdot x \\ P(3 \cdot i + 1) = \beta \cdot m \cdot y \\ P(3 \cdot i + 2) = \gamma \cdot m \cdot z \\ i = 0, 1, 2, 3, \ldots, k. \end{cases} \quad eq(4)$$

Where:

$$\begin{cases} \alpha = \dfrac{\Sigma\ |x(i)|}{n} \\ \beta = \dfrac{\Sigma\ |y(i)|}{n} \\ \gamma = \dfrac{\Sigma\ |z(i)|}{n} \\ m = 251 \cdot \alpha \cdot \beta \cdot \gamma \end{cases} \quad eq(5)$$

x(i), y(i), z(i) are the samples from Chen chaotic system, α, β, γ are the averages of absolute sample values, and k is length of one of the orbits x, y, z.

The generated sequence P is a sequence of real numbers (IR), we remove the last two values in the sequence P that we added in the beginning, then we can generate the sequence S using this equation:

$$S = \text{round}|P| \bmod l \quad eq(6)$$

Where l is either 2 for a binary sequence or 256 for values between (0,255), as shown in Figure (4,2) flowchart of the .

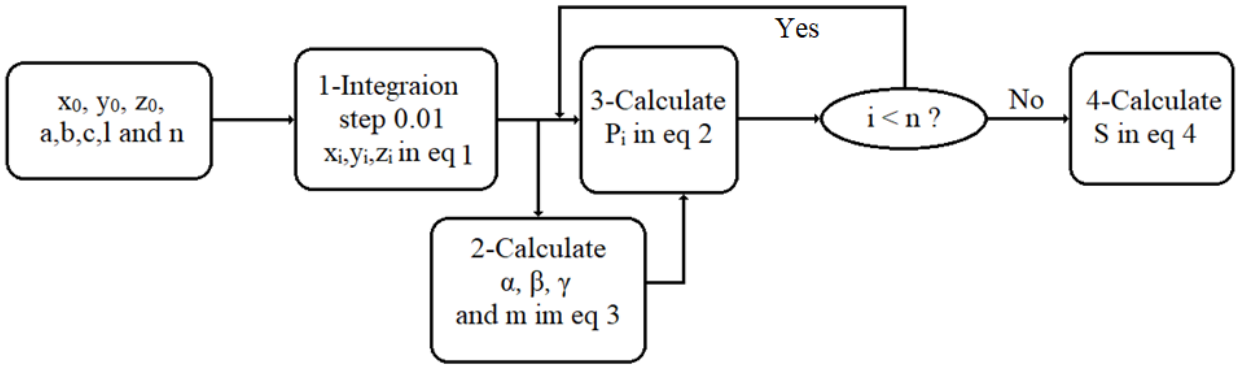


Figure (4,2) flowchart of the PRNG

Figure (4,3) Lena Grayscale

In our experiment, we gave the initial values (x_0 , y_0 , z_0) a random values each time so we can guarantee we never have the same sequence twice even for the same image. we suppose there is some noise associate our voice recording, we adjust the frequency to 27000 and we recorded noise for 3 seconds only. Figure (14,4) Example of recorded voiceshows an example of recorded voice. As results, we got a list of 81000 values, which is 27000×3, the problem is most of the values are too small and we cannot work with them, so we took the generated list 'a' and we applied the following equation on it: a = abs(a×frequency×256)

The result was a list of positive values and enough to use in. Later, we removed the zero values. Now, when we perform the PRNG, we take the original initial values (x_0 , y_0 , z_0) = (−1.5, 0.6, 17) and we adjust the new initial values as the following:

$x_0 = a_0 \% -1.5$
$y_0 = a_1 \% 0.6$
$z_0 = a_2 \% 17$

then a_0, a_1 and a_2 are removed from the list 'a' to ensure the randomness of the next sequence.

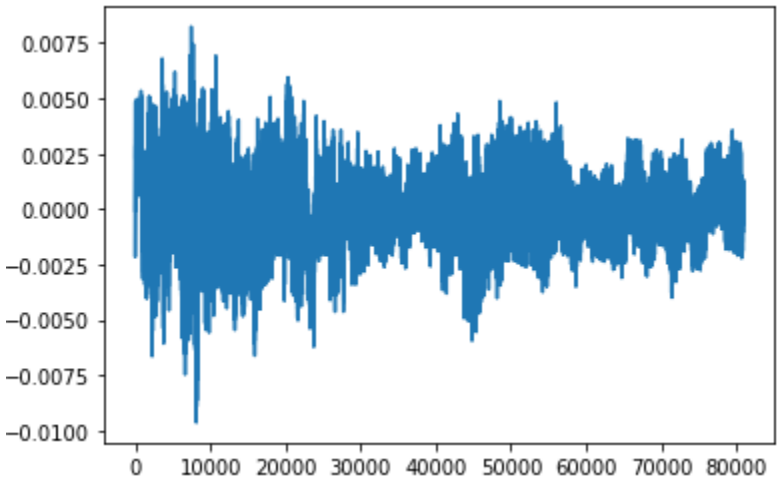


Figure (14,4) Example of recorded voice

4.3 The PRNG in RGB digital images

In the case of RGB digital image, we have two types:

The first when the shape of the digital image is [x,y,3], where x,y are the dimensions and 3 is the number of matrices.

The second when the shape is [x,y,4], the 4^{th} matrix is the transparency matrix.

In the case of 3, we iterate the chaotic system for 3×(n/3) or 3×((1+n)/3) or 3×((2+n)/3).

In the case of 4, we iterate the chaotic system for 4×(n/3) or 4×((1+n)/3) or 4×((2+n)/3), and the rest steps stay the same.

4.4 The proposed encryption algorithm:

Recently, a number of chaos-based image encryption algorithms have been proposed at the pixel level, but little research at the bit level has been conducted.

The encryption process goes through 7 steps:

1- Iterate the PRNG 8 times or 24 times if the image is RGB with three matrices and 32 times if it is with four matrices (transparency matrix), each time with a different initial values (x_0,y_0,z_0), these values are taking from a random source to ensure no value is equal to another, and finally we take l = 2 for a binary sequences, so the result of this step is the generating of s1, s2, s3, s4, s5, s6, s7, s8, no sequence is like the other.

2- The next step is to take the digital image and we slice it into 8 (or 24 or 32) bit-plans, the bit-plan is shown in the Figure (4,5) 3-bit image matrix bit-plane slicing with a 3-bit image example:

110	111	110	110	111
000	000	000	001	010
001	001	001	010	011
100	101	101	100	010
110	110	110	111	111

1	1	1	1	1
0	0	0	0	0
0	0	0	0	0
1	1	1	1	0
1	1	1	1	1

1	1	1	1	1
0	0	0	0	1
0	0	0	1	1
0	0	0	0	1
1	1	1	1	1

0	1	0	0	1
0	0	0	1	0
1	1	1	0	1
0	1	1	0	0
0	0	0	1	1

Figure (4,5) 3-bit image matrix bit-plane slicing

The result of this step is 8 (or 24 or 32) bit-plane images: b1, b2, b3, b4, b5, b6, b7, b8…, each one with the same dimensions as the original image.

3- We XOR each ‘s’ sequence with a ‘b’ sequence, the result of this operation is x1, x2, x3, x4, x5, x6, x7, x8.

4- Then we re-assemble the image in a backward fashion, for example: b1(1) =0, b2(1) =1, b3(1) =1, b4(1) =0

b5(1) =1, x6(1) =0, b7(1) =0, b8(1) =0

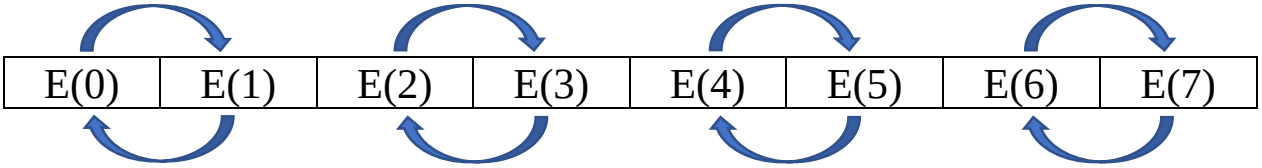
so the normal re-assemble result would be: $01101000_{(2)} = 104_{(10)}$

but the backward re-assemble would be: $00010110_{(2)} = 22_{(10)}$

The result of this step is an encrypted image ‘E’

5- Now we iterate the PRNG one last time with a random initial values for (x_0,y_0,z_0), with l = 256, the result of this step is a pseudo random sequence ‘F’.

6- We switch each 2 pixels with each other, like this:



So, the result is:

E(1)	E(0)	E(3)	E(2)	E(5)	E(4)	E(7)	E(6)

We call the result ‘K’

7- We XOR ‘K’ and ‘F’ to get the final encrypted image.

The decryption process is the reverse of the encryption process.

Figure (4.6) and (4.7) (a, b, c. d) depict the original image (lean and Baboon) (left) and the encryption image (right) and its corresponding histogram, respectively.

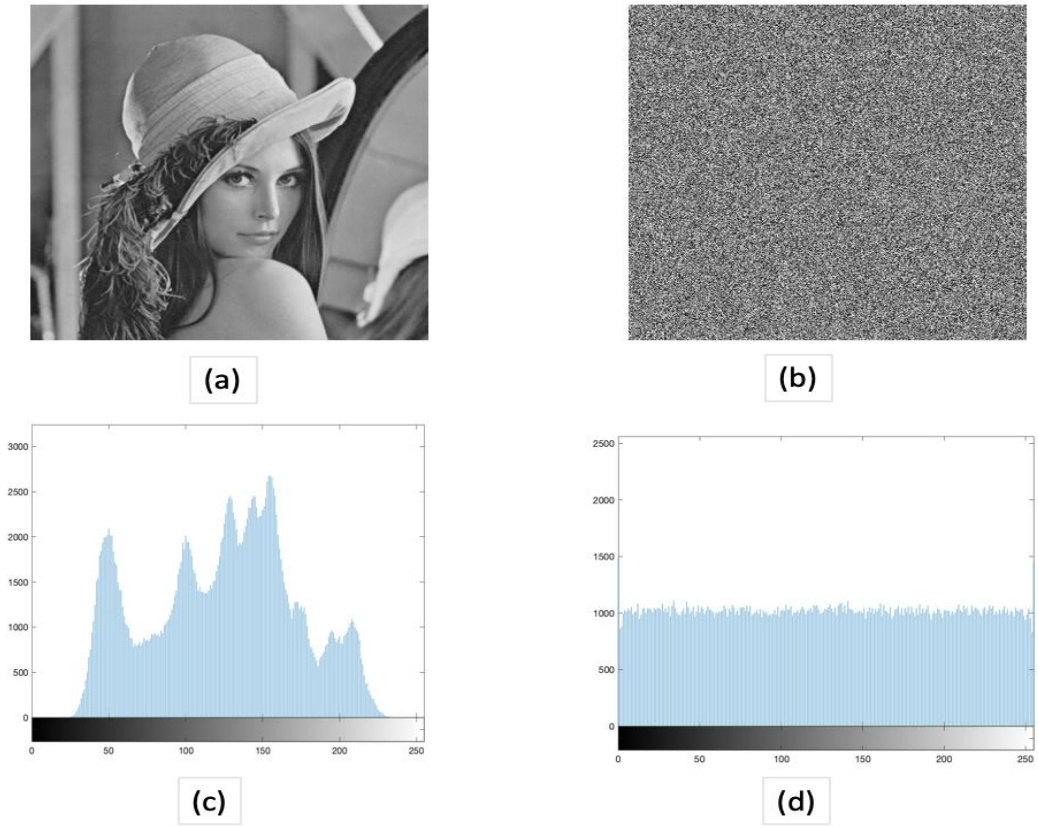


Figure 15) (a) Lena grayscale digital image, (b) the encryption of Lena image using the proposed algorithm, (c) the histogram of Lena, (d) the histogram of the encrypted image

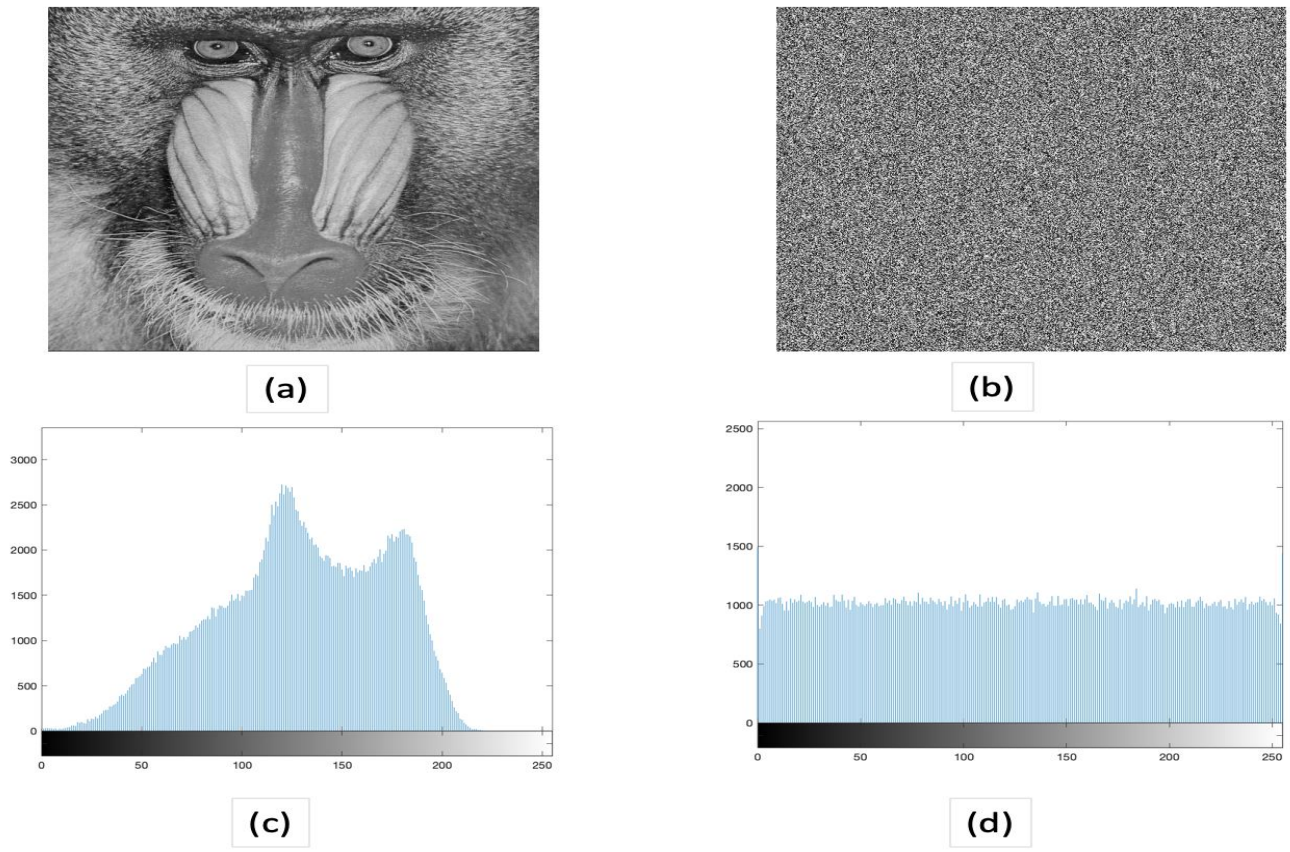


Figure (4,7) (a) Baboon grayscale digital image, (b) the encryption of Baboon image using the proposed algorithm, (c) the histogram of Baboon, (d) the histogram of the encrypted image

5. Experimental results and performance analysis

In this section, we discuss and analyze the performance of the proposed encryption scheme, including histograms, correlation coefficients, key sensitivity, and the steps order sensitivity. In the experiments, the images for testing are the two digital images (Lena and Baboon) of size 256 × 256 is shown in Fig. 4 and 5(a). Moreover, we used python and Matlab R2019b to assess the robustness of the encryption algorithm in a personal computer with a 2.6 GHz AMD CPU and 8 GB memory.

5.1 Histogram analysis

An image histogram represents the pixels values intensity distribution in an image, so to resist any statistical attack and to ensure a secure encryption system, the histogram of the encrypted image must be uniform. Figure (4.6) shows the histograms of the Lena image and the corresponding encrypted image. Figure (4.7) also shows the histograms of the Baboon image and its encrypted image. In Figure (4.6) (d) and Figure (4.7) (d) all of the grayscale values of the

encrypted images are distributed uniformly over the interval [0, 255], which is significantly different from the distribution of the original Lena and Baboon images in Figure (4.6) (c) and Figure (4.7) (c).

5.2 Correlation analysis

The original image's adjacent pixels show a high correlation in horizontal, vertical, and diagonal directions. An ideal encryption algorithm should ensure that the correlation coefficients of the pixels in the encrypted image have a low enough correlation to withstand statistical attacks. To this end, we employed 4000 pairs of neighboring pixels in are randomly picked from the plain image and its encrypted picture to evaluate and compare the similarities of the adjacent pixels in the plain image and encrypted image. The correlation distribution in three directions of the two neighboring pixels is shown in Figure (4.8). As observed, the dissemination of adjacent pixels in the original image are highly concentrated, meaning there is a strong correlation in the original image. However, the distributions of adjacent pixels in the encrypted image of the original image are random, meaning the encrypted image has a low correlation. The results show that the correlation coefficients of the original image are close to 1, whereas the coefficients of the encrypted image are around 0 in all three directions. Table 1**Error! Reference source not found.** shows that the adjacent pixels of the encrypted image have extremely low correlation and a good confusion and diffusion properties of the proposed image encryption scheme.

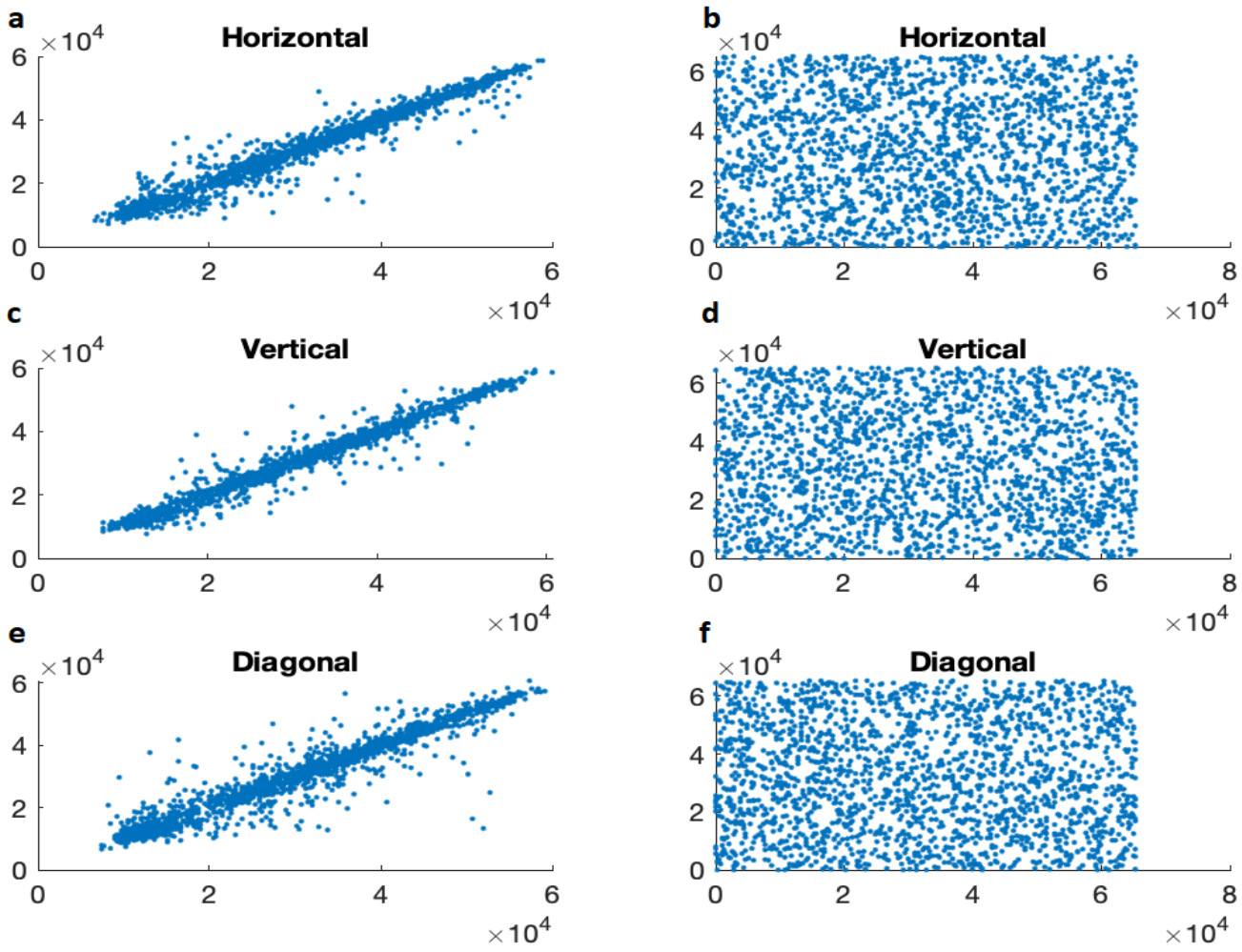


Figure (4,8) The correlation plots of the Lena image and the corresponding ciphered image: (a) horizontal correlation of the Lena image, (b) vertical correlation of the Lena image, (c) diagonal correlation of the Lena image, (d) horizontal correlation of the ciphered image of Lena, (e) vertical correlation of the ciphered image of Lena, (f) diagonal correlation of the ciphered image of Lena.

Direction	Horizontal	Vertical	Diagonal
Plain-image of Lena	0.97	0.9847	0.9596
Encrypted-image of Lena	-0.0249	-0.094	-0.147
Plain-image of baboon	0.86	0.76	0.72
encrypted-image of baboon	0.03	0.08	-0.0094

Table 7 Correlation coefficients of the original and encrypted images.

5.3 Information entropy analysis

In the Information Theory, the most critical measure of randomness is information entropy. Let 'm' be the source of information which is the grayscale image, the formula for calculating the information entropy is shown in the following equation:

$$H(m) = \sum_{i=0}^{2^n-1} p(m_i) log_2 \frac{1}{p(\boldsymbol{m_i})} \qquad \text{eq(7)}$$

Where n is the number of bits that is required to represent the symbol mi, and p(mi) is the probability of symbol m. when all pixels are equally distributed, the entropy would be 8, which is the maximum value of an 8-bit image, meaning that the information is random, so the closest the entropy value of an encrypted image is to 8 the hardest is for the attacker to decrypt the ciphered image. After calculating the entropy of the encrypted images, we present the result in Table 2. The entropies are all very close to the ideal value 8, which means that the probability of accidental information leaking is minimal. Thus, the proposed algorithm has the desired property of information entropy.

Image	Lenna	Baboon
Entropy	7.992	7.9978

Table 8 Information entropy

5.4 The proposed encryption system Analysis

Secure encryption that means the proposed algorithm is high sensitivity to any the noise effects caused by minor modification in pixels in the decrypted image. For checking the capability of our proposed scheme in opposing noise and data loss attacks, we take the Lena image of size. Three standard measures of testing the influence of one-pixel change on the whole image Lena as a test case. We evaluated the entropy and correlation using the standards measures in the horizontal, vertical, and diagonal directions.

	Correlation Horizontal	Correlation vertical	Correlation diagonal	Entropy
Case 1	0.0167	0.0461	0.0121	7.992
Case 2	0.4	0.0142	0.0183	7.9978
Case 3	0.0178	0.0363	-0.0051	7.9982

Table 9 Correlation and entropy values of the proposed encryption scheme tests

Case 1: The sensibility to the reversing between the bit plan values (Step 4)

In this test, we made the decryption step by step and but we missed to reverse the fourth step, to show that the produced decrypted image and its histogram changed utterly compared to the original image Figure (4.9).

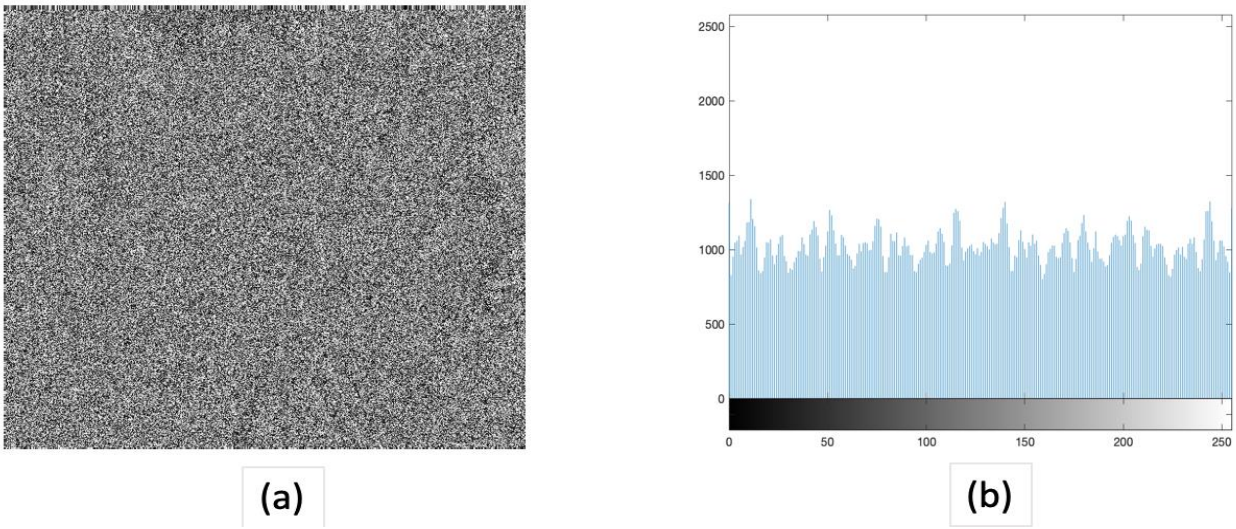


Figure 16) The decrypted image when neglecting the fourth step, (b) the corresponding histogram

Case 2: The sensibility of the permutation between two neighbor pixels positions. (step 5)

We performed the decryption by ignoring the permutation of step 5. Then, we computed two variances of histograms of ciphered images. The final encrypted image and its histogram changed utterly compared to the original image figure (4.10).

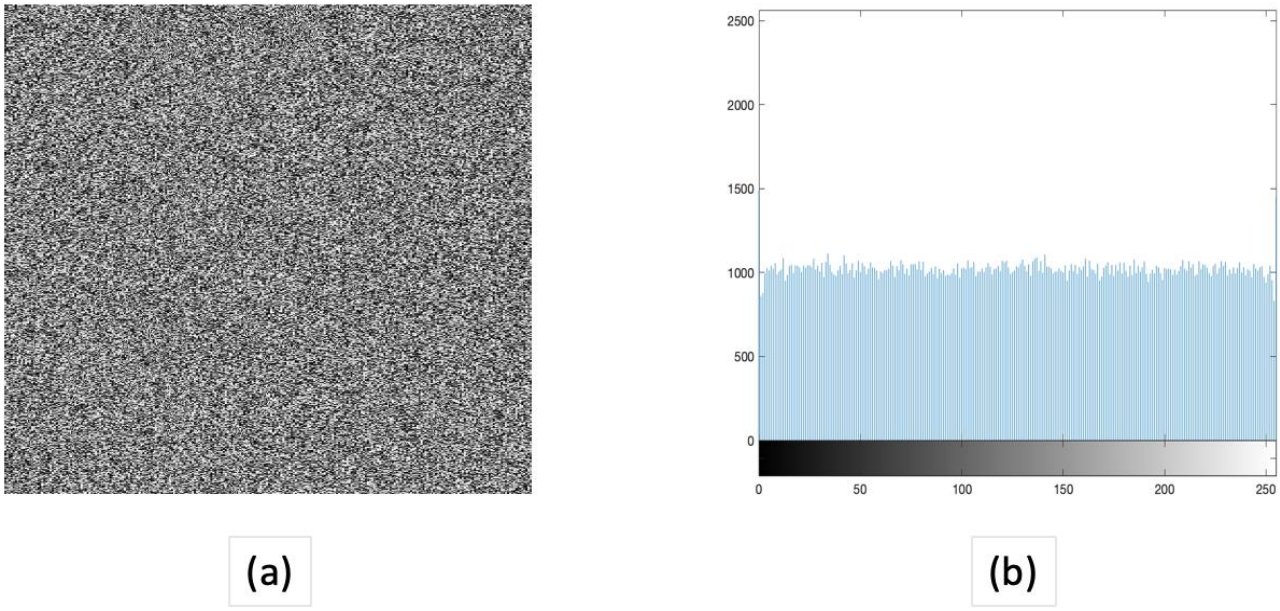


Figure (4,10) (a) the decrypted image when not switching every two pixels, (b) the corresponding histogram

Case 3: Key sensitivity (step 2)

Any image encryption algorithm should be highly aware of its secret key, and a little modification of a single-bit in its secret key should destroy an entirely different encrypted result. In the sensitivity analysis for the secret key, a highly sensitive key is required. We notice here, the output of the decrypted algorithm completely changed even if we make a minor change in any of both components of Keys (binary and integer). For example, we generated the key1 to key9 using a slightly off values, Table 4 (a) shows the original correct values used in the encryption process. In contrast, Table 4 (b) shows the slightly wrong values used in the decryption process, and we only changed small values. We focused on the values of 'y' to show the sensitivity of the pseudorandom number generator we used, Table 4 (a) shows the result of decrypting the ciphered image using the wrong keys input from Table 4 (b) and Figure (4.11) (b) shows the corresponding histogram; the image still impossible to read with a distribution similar to the first encrypted image in Figure 26(d).

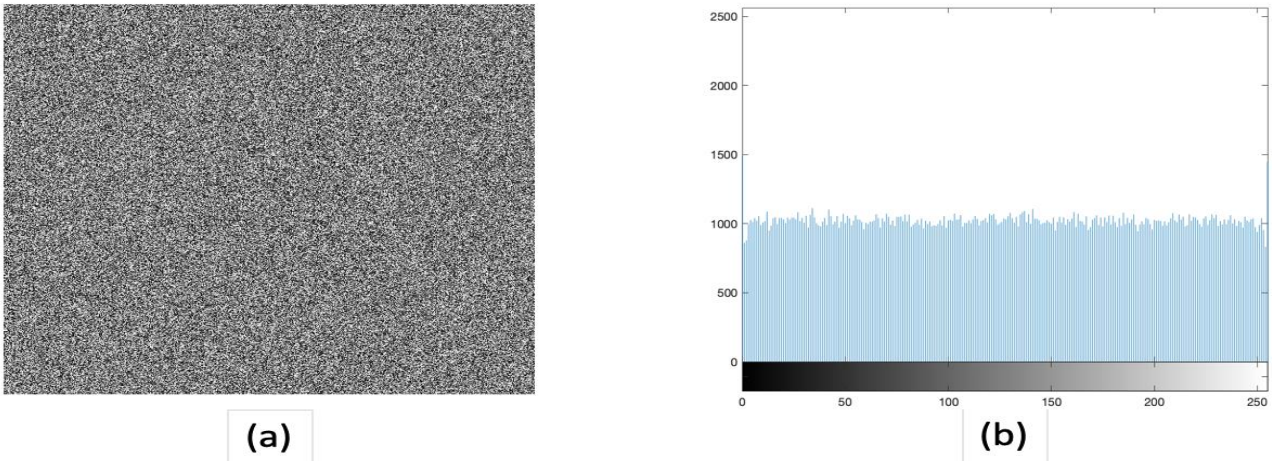


Figure (4,11) the decrypted image using wrong keys inputs, (b) the corresponding histogram

(a)	x_0	y_0	z_0	l
Iteration 1	-1.5	0.6	17	2
Iteration 2	-1	0.8	12	2
Iteration 3	-0.5	0.2	15	2
Iteration 4	-1.2	0.3	13	2
Iteration 5	-1.8	0.7	11	2
Iteration 6	-2.3	0.1	10	2
Iteration 7	-1.4	0.4	14	2
Iteration 8	-2	0.9	18	2
Iteration 9	-2.1	0.9	13	256

(b)	x_0	y_0	z_0	l
Iteration 1	-1.5	**0.5**	17	2
Iteration 2	-1	**0.7**	12	2
Iteration 3	-0.5	**0.25**	15	2
Iteration 4	-1.2	**0.31**	13	2
Iteration 5	-1.8	**0.72**	11	2
Iteration 6	-2.3	0.1	10	2
Iteration 7	-1.4	**0.2**	14	2
Iteration 8	-2	0.9	18	2
Iteration 9	-2.1	**0.8**	13	256

Table 10 (a)The original values used in the encryption process, (b) the wrong values used in the decryption process

Overall comparison of entropy values with proposed attack schemes is given below in Table 6:

Test	Case 1	Case 2	Case 3
Entropy	7.992	7.9978	7.9982

Table 11 Entropy values comparison

6.Conclusion

In this work, we proposed an image encryption algorithm by incorporating the bit plane slicing. The proposed algorithm encrypts the digital image using a bit-plane slicing method and a secure pseudorandom number generator. The results confirm the robustness of our cryptosystem. The experimental results confirm the robustness of our hybrid algorithm against different attacks. The proposed encryption algorithm shows a competitive performance compared to the state-of-the-art image encrypted algorithms in terms of entropy and correlation and can resist to different attacks. In future work, we aim to enhance the proposed research by investigating the quantum encryption based on the PGNR.

Approach	Our work	Ref[33]	Ref[33]	Ref[34]	Ref[35]	*Ref[36]*	*Ref[37]*	*Ref[38]*
Entropy	**7.9981**	7.9974	7.9982	7.9970	7.997	7.993	7,9971	7.9968

Table 12 Information entropy results comparison with some other image encryption algorithms

General Conclusion and future works:

In the last decades, many encryption and security schemes and techniques were proposed to protect and secure digital images against hackers. But, as the security techniques are evolving, the hacking mechanisms are also evolving, so, it is more like a race between hackers and the security community.

The main goal of this thesis is to investigate on the behavior of algorithm based on Chen chaotic map to generate a pseudo random number sequence. In our work, we thought to use chaotic map and bit plain slicing in order to encrypt the digital image on the bit level, where many encryption schemes are focused on the pixel level. To the best of our knowledge, focusing on the bit level makes the cryptanalysis harder for the attacker than the pixel level. Moreover, we add more encryption steps to make our encryption scheme more sensitive when encrypting or decrypting the digital image It should be highly sensitive against any changes in the initial conditions.

However, our algorithm still needs further improvement, by incorporating the confusion step on the pixel position to make our algorithm more secured.

So finally, in the future works, we have planned to tackle two points, the first is the true random number generators or TRNG, which is more suitable for real time application such as image blurring. Having TRNG different properties from the PRNG, so we can study many exiting PRNG to extend it to true random number to perform it in different fields. A hot topic is also appeared in the recent year, called the quantum cryptography, which will be take a big part of the cryptography studies today and well may make a revolution on the industry 4.

References:

[1] Tarun Kumar: A Theory Based on Conversion of RGB image to Gray image, International Journal of Computer Applications (0975 – 8887) Volume 7– No.2, September 2010.

[2] Burger, W., and Burge, M. J. Principles of digital image processing: fundamental techniques, vol. 1. Springer Science & Business Media, 2010.

[3] Parker, M., and Dhanani, S. Digital video processing for engineers: A foundation for embedded systems design. Newnes, 2013.

[4] Fouque, P.-A., Martinet, G., and Poupard, G. Practical symmetric on-line encryption. In FSE (2003), vol. 2887, Springer, pp. 362–375.

[5] Verdult, R. The (in) security of proprietary cryptography. Sl: sn, 2015.

[6] Plataniotis, K., and Venetsanopoulos, A. N. Color image processing and applications. Springer Science & Business Media, 2013.

[7] Lian, S. Multimedia content encryption: techniques and applications. CRC press, 2008.

[8] Lambic, D. ´ Security analysis and improvement of a block cipher with dynamic s-boxes based on tent map. Nonlinear Dynamics 79, 4 (2015), 2531–2539.

[9] Li, C., Liu, Y., Zhang, L. Y., and Chen, M. Z. Breaking a chaotic image encryption algorithm based on modulo addition and xor operation. International Journal of Bifurcation and Chaos 23, 04 (2013), 1350075.

[10] Liu, Y., Zhang, L. Y., Wang, J., Zhang, Y., and Wong, K.-w. Chosen-plaintext attack of an image encryption scheme based on modified permutation diffusion structure. Nonlinear Dynamics 84, 4 (2016), 2241–2250.

[11] Wang, X., Luan, D., and Bao, X. Cryptanalysis of an image encryption algorithm using chebyshev generator. Digital Signal Processing 25 (2014).

[12] Cayre, F., Fontaine, C., and Furon, T. Watermarking security part i: Theory. In Security, Steganography, and Watermarking of Multimedia Contents VII (2005), vol. 5681, SPIE, pp.

[13] Chen, T.-H., Chang, C.-C., Wu, C.-S., and Lou, D.-C. On the security of a copyright protection scheme based on visual cryptography. Computer Standards & Interfaces 31, 1 (2009), 1–5.

[14] Mitzenmacher, M., and Upfal, E. Probability and Computing: Randomization and Probabilistic Techniques in Algorithms and Data Analysis. Cambridge university press, 2017.

[15] Goldwasser, S., and Micali, S. Probabilistic encryption. Journal of computer and system sciences 28, 2 (1984), 270–299.

[16] Rivest, R. L., and Sherman, A. T. Randomized Encryption Techniques. Springer US, Boston, MA, 1983, pp. 145–163.

[17] liang Zhu, Z., Zhang, W., wo Wong, K., and Yu, H. A chaos-based symmetric image encryption scheme using a bit-level permutation. Information Sciences 181, 6 (2011).

[18] Pareek, N., Patidar, V., and Sud, K. Image encryption using chaotic logistic map. Image and Vision Computing 24, 9 (2006).

[19] Wang, X., Teng, L., and Qin, X. A novel color image encryption algorithm based on chaos. Signal Processing 92, 4 (2012).

[20] Wong, K.-W., Kwok, B. S.-H., and Yuen, C.-H. An efficient diffusion approach for chaos-based image encryption. Chaos, Solitons & Fractals 41, 5 (2009).

[21] Hu, H., Liu, L., and Ding, N. Pseudorandom sequence generator based on the chen chaotic system. Computer Physics Communications 184, 3 (2013).

[22] Wang, X.-Y., and Wang, X.-J. Design of chaotic pseudo-random bit generator and its applications in stream-cipher cryptography. International Journal of Modern Physics C 19, 05 (2008).

[23] Lian, S. Multimedia content encryption: techniques and applications. CRC press, 2008.

[24] Luby, M. G., and Luby, M. Pseudorandomness and cryptographic applications. Princeton University Press, 1996.

[25] Brown, J. A., Houghten, S., and Ombuki-Berman, B. Genetic algorithm cryptanalysis of a substitution permutation network. In Computational Intelligence in Cyber Security, 2009. CICS'09. IEEE Symposium on (2009), IEEE, pp. 115–121.

[26] Shannon, C. E. Communication theory of secrecy systems*. Bell system technical journal 28, 4 (1949).

[27] Wu, Y., Zhou, Y., Saveriades, G., Agaian, S., Noonan, J. P., and Natarajan, P. Local shannon entropy measure with statistical tests for image randomness. Information Sciences 222 (2013).

[28] Lu Xu,ZhiLi, JianLi,Wei Hua A novel bit-level image encryption algorithm based on chaotic maps. Optics and Lasers in Engineering 78 (2016).17–25.

[29] Rafik Hamza∗, A novel pseudo random sequence generator for image-cryptographic applications, Journal of Information Security and Applications 35 (2017) 119–127

[30] Chen G, Ueta T. Yet another chaotic attractor. Int J Bifurcation Chaos 1999;9(07):1465–6.

[31] Li Y, Tang WK, Chen G. Generating hyperchaos via state feedback control. Int J Bifurcation Chaos 2005a;15(10):3367–75.

[32] Lorenz EN. Deterministic nonperiodic flow. J Atmos Sci 1963;20(2):130–41.

[s1] Vectorfresh: Sample pack of imaging eps. https://www.vectorfresh.com/freeimages-SamplePack1.html. Accessed: 2017-06-25.

[33] Xu,L.,Li,Z.,Li,J.,Hua,W.:Anovel bit-level image encryption algorithm based on chaotic maps. Opt. Laser Eng. 78, 17–25 (2016)

[34] Wang, X., Liu, L., Zhang, Y.: A novel chaotic block image encryption algorithm based on dynamic random growth technique. Opt Laser Eng. 66, 10–18 (2015)

[35] Zhang Y, Tang Y (2018) A plaintext-related image encryption algorithm based on chaos. Multimed Tools Appl 77(6):6647–6669.

[36] Zhu H, Zhao C, Zhang X (2013) A novel image encryption-compression scheme using hyper-chaos and hinese remainder theorem. Signal Process 28(6):670–680

[37] Chai X, Gan Z, Yanga K, Chen Y, Liu X (2017) An image encryption algorithm based on the memristive hyperchaotic system, cellular automata and DNA sequence operations. Signal Process Image Commun 57:

[38] MOZAFFARI, Saeed. Parallel image encryption with bit-plane decomposition and genetic algorithm. Multimedia Tools and Applications, 2018, vol. 77, no 19, p. 25799-25819.