UNIVERSITY MOHAMMED SEDDIK BENYAHIA
JIJEL
FACULTY OF EXACT SCIENCES AND COMPUTER SCIENCE

MASTER'S THESIS

Presented to obtain the diploma of :

MASTER

In **COMPUTER SCIENCE**

Option: ARTIFICIAL INTELLIGENCE

By :

Khalil Bentaiba and **Haitem Bourahla**

Theme

Application of Neural Networks to Image Encryption

M. Boukherrou Zoubida Supervisor
M. Hemioud Mourad Co-Supervisor

2021/2022

Dedecation

I dedicate this modest work:

To my dear Parents,

To my dear siblings,

To all my dears and friends especially ***Amine****,* ***Hamza****,****Aymen****,*

To the Club Esperanza and all the people in it,

To my partner in this project ***Khalil****,*

To all my classmates and the people that have been there for me.

Haitem.

Dedecation

I dedicate this modest work:

To my dear Parents,

To my dear siblings,

To all my dears and friends especially ***Amine***, ***Hamza***,

To the Club Esperanza and all the people in it,

To my partner in this project ***Haitem***,

To all my classmates and the people that have been there for me.

Khalil.

Acknowledgements

We would like to express our special thanks of gratitude to our teachers ***Hemioud Mourad*** *and* ***Boukherrou Zoubida*** *who gave us the golden opportunity to do this wonderful project on the topic "Application of neural networks to image cryptography", which also helped us in doing a lot of Research and we came to know about so many new things, and also get better on so many other things.We would also like to extend our deepest gratitude to our friends who helped us and walked with us through this path, We are also extremely grateful to both of our parent for being out there for us. We are really thankful to all of them.*

Abstract

One of the major aspects of computer systems is cryptography. In this work, we are going to take general ideas of the basics of cryptography systems. And since emerging technologies are not complete without neural networks, and current research has demonstrated that these systems can be used for a wide range of applications, we will talk about the basics of neural networks in cryptography and their domain of application.

In this work, we introduce a system of Six-Dimensional Cellular Neural Network (6D-CNN) to generate a pseudo-random number. With additional proposed improvement method to generate the initial key conditions of the 6D-CNN using Lorenz 3D and Chen's 3D systems. The presented 6D-CNN has hyper-chaos characteristics, very good sensitivity to initial conditions, and excellent randomness.

We proposed a novel image encryption algorithm based on chaos and CNN, where it takes the architecture of chaos based on a substitution-diffusion image encryption cryptosystem. In the substitution part, we shuffle the image pixels coordinates, whereas in diffusion we use the generated pseudo-random sequence by 6D-CNN to encrypt the shuffled image.

Finally, we run a security evaluation for our cryptosystem to assure its ability to offer the necessary security. Additionally, we compared our scheme to other related works to express its efficiency.

Keywords : *Image, Cryptography, Neural Network, Cellular Neural Network, Pseudo-Random Number, Sequence, Chaos, Encryption, Decryption*

Résumé

L'un des principaux aspects des systèmes informatiques est la cryptographie. Dans ce travail, nous allons prendre des idées générales sur les bases des systèmes de cryptographie. Comme les technologies émergentes ne sont pas complètes sans les réseaux neuronaux, et que les recherches actuelles ont démontré que ces systèmes peuvent être utilisés pour un large éventail d'applications, nous allons parler des bases des réseaux neuronaux en cryptographie et de leur domaine d'application.

Dans ce travail, nous introduisons un système de Réseau Neuronal Cellulaire à Six Dimensions (RNC-6D) pour générer un nombre pseudo-aléatoire. Nous avons également proposé une méthode d'amélioration pour générer les conditions initiales de la clé du RNC-6D en utilisant les systèmes 3D de Lorenz et 3D de Chen. Le RNC-6D présenté présente des caractéristiques d'hyper-chaos, une très bonne sensibilité aux conditions initiales et un excellent caractère aléatoire.

Nous avons proposé un nouvel algorithme de cryptage d'image basé sur le chaos et le RNC, où il prend l'architecture du chaos basée sur un cryptosystème de cryptage d'image de substitution-diffusion. Dans la partie substitution, nous mélangeons les coordonnées des pixels de l'image, tandis que dans la diffusion, nous utilisons la séquence pseudo-aléatoire générée par le RNC-6D pour crypter l'image mélangée.

Enfin, nous effectuons une évaluation de la sécurité de notre système de cryptage afin de nous assurer de sa capacité à offrir la sécurité nécessaire. En outre, nous avons comparé notre schéma à d'autres travaux connexes pour exprimer son efficacité.

Keywords : *Image, Cryptographie, réseau neuronal, réseau neuronal cellulaire, nombre pseudo-aléatoire, séquence, chaos, cryptage, décryptage.*

ملخص

يعد التشفير أحد الجوانب الرئيسية في أنظمة الكمبيوتر. في هذا العمل ، سوف نأخذ أفكارًا عامة عن أساسيات أنظمة التشفير. وبما أن التقنيات الناشئة لا تكتمل بدون الشبكات العصبية ، وقد أثبتت الأبحاث الحالية أن هذه الأنظمة يمكن استخدامها في مجموعة واسعة من التطبيقات ، فسوف نتحدث عن أساسيات الشبكة العصبية في التشفير ومجال التطبيق.

في هذا العمل ، قدمنا نظام شبكة عصبية خلوية سداسية الأبعاد (6D-CNN) لتوليد رقم عشوائي شبه عشوائي. مع طريقة التحسين الإضافية المقترحة لإنشاء الشروط الأساسية الأولية لـ 6D-CNN باستخدام أنظمة Lorenz 3D و Chen's 3D . تتميز 6D-CNN المقدمة بخصائص الفوضى المفرطة ، وحساسية جيدة جدًا في الظروف الأولية ، وعشوائية ممتازة.

لقد اقترحنا خوارزمية جديدة لتشفير الصور تعتمد على الفوضى و CNN ، حيث تأخذ بنية الفوضى القائمة على نظام تشفير تشفير الصور بالاستبدال والانتشار. في جزء الاستبدال ، نقوم بتبديل إحداثيات بكسل الصورة ، حيث نستخدم في الانتشار التسلسل العشوائي الزائف الذي تم إنشاؤه بواسطة 6D-CNN لتشفير الصورة التي تم خلطها.

أخيرًا ، نجري تقييمًا أمنيًا لنظام التشفير الخاص بنا للتأكد من قدرته على تقديم الأمان اللازم. بالإضافة إلى ذلك ، قمنا بمقارنة مخططنا بالأعمال الأخرى ذات الصلة للتعبير عن كفاءته.

الكلمات الرئيسية: الصورة ، التشفير ، الشبكة العصبية ، الشبكة العصبية الخلوية ، الرقم العشوائي الزائف ، التسلسل ، الفوضى ، التشفير ، فك التشفير

CONTENTS

LIST OF FIGURES

LIST OF TABLES

GENERAL INTRODUCTION

Preamble

As is well known, the enormous transition of corporate, personal, and governmental services into electronic, web-enabled forms has been made possible by the enormous rise in networks, connectivity, and communications. The threat surface was significantly increased by the advent of e-business and e-commerce. Worldwide, criminals, unscrupulous business rivals, and country states. The world has essentially operated as a network monoculture for decades. Most of the internet, the worldwide web, and e-business are powered by a single set of protocols and standards. These standards are used by almost every laptop, smart device, and other endpoints to connect to servers, applications, organizations, and governments. As a result, these models or protocol stacks serve as our threat surface map. To secure the information for every individual, the share of the data itself had to be secured. And this is where cryptography has taken a huge part in communication security.

Cryptography security experts must have a thorough knowledge of how current networks and the internet operate, including its ideas, approaches, technologies, and security problems. Because they must be familiar with the area, their task is similar to police patrols. Specialists in cryptography must be knowledgeable of the finest techniques for protecting the environment from attacks and maintaining security. If the business has limited or no remote sight into its operational systems, the requirement for such a policing mentality is very urgent.

There are numbers of systems cryptography that reflect good security and privacy. one of the famous systems is cryptography based on chaos. This chaos branch has nonlinear dynamics features, which have been extensively researched. This approach's use of nonlinear dynamics is being researched for several applications in actual systems. Chaotic behavior is a subtle, seemingly random behavior of a nonlinear system. However, the source of this randomness is not stochastic. It is just the outcome of the deterministic processes that define it. The high sensitivity of chaos to the system's initial conditions is one of its key properties.

In recent years, there was a huge blow in the domain of neural networks. Which improves a lot of research. Emerging technologies are not complete without neural networks, and current research has demonstrated that these systems can be used for a wide range of applications. The development of neural networks has given us several options to improve cryptosystems. where neural network techniques take a huge part in the domain of chaos-based cryptography.

In chaos cryptography, neural networks play the role of random number generator (RNG).

The results show, that neural networks have good randomness and a great sensitivity to initial conditions. These RNG system-generated numbers are applied to secure telecommunications and transaction systems. The idea is to cipher a message with a chaotic generated pseudo-number to disrupt attacks.

Context and Motivation

Worldwide, communication frequently involves the use of images. Numerous applications employ images, such as medical images, military images, remote sensing, educational images, electronic commerce, and so on. Due to the availability of the Internet. We can exchange information and images anywhere, and anytime.

Nevertheless, certain images could make reference to sensitive business or private data. In order to defend against unauthorized access, modification, and other threats, image security should receive a lot of attention. Recently, Numerous image encryption algorithms, such as chaos-based techniques, compressive sensing-based systems, visual cryptography, and others, have been presented and published. We require a greater extent of security protection for images in the age of information and the introduction of big data. Since images are shared and accessed via open networks, applying encryption to the image is an easy technique to secure data and hide image details. However, the traditional image encryption techniques will face a challenge from quantum computers operating in a 5G network environment. Further research is required on new image encryption methods.

Context and Motivation

Worldwide, communication frequently involves the use of images. Numerous applications employ images, such as medical images, military images, remote sensing, educational images, electronic commerce, and so on. Due to the availability of the Internet. We can exchange information and images anywhere, and anytime.

Nevertheless, certain images could make reference to sensitive business or private data. In order to defend against unauthorized access, modification, and other threats, image security should receive a lot of attention. Recently, Numerous image encryption algorithms, such as chaos-based techniques, compressive sensing-based systems, visual cryptography, and others, have been presented and published. We require a greater extent of security protection for images in the age of information and the introduction of big data. Since images are shared and accessed via open networks, applying encryption to the image is an easy technique to secure data and hide image details. However, the traditional image encryption techniques will face a challenge from quantum computers operating in a 5G network environment. Further research is required on new image encryption methods.

Contributions

In this work, we introduced a novel image encryption-based chaos and a six-dimensional cellular neural network (6D-CNN) pseudo-random number sequences generator. And we proposed an improved method to generate the initial key conditions for the 6D-CNN using a three-dimensional Lorenz system and three-dimensional Chen's system, which make the initial conditions for the 6D-CNN have chaotic characteristics.

The image encryption is based on the chaos theory "Substitution-Diffusion". Firstly, in the Substitution part, we take the image and shuffle all the pixel positions using a chaos substitution method known as Arnold's Cat map, the result of this part is a scrambled image. Secondly, in the diffusion part, we generate a pseudo-random number sequence using the 6D-CNN generator, and by using the XOR operator, we encrypt the scrambled image in the substitution part. Lastly, we did an analysis of the generated pseudo-random 6D-CNN sequence chaos and efficiency. We did a general evaluation and analysis of the proposed cryptosystem to test its ability in offering necessary security. Moreover, we did a competition with other image encryption references to show efficiency and performance.

The main contributions are summarized as follows:

- new proposition for image encryption based on Chaos "Substitution-Diffusion" and 6D-CNN.
- image encryption based on substitution method Arnold's Cat mat.
- image encryption based on Diffusion method 6D-CNN sequence generator.
- new proposition for the initial key conditions of the 6D-CNN.
- testing and analyzing the chaos of the 6D-CNN generated pseudo-random sequences.
- evaluating and analysis to the proposed image encryption
- comparison between the proposed image encryption with other related schemes.

Many figures, shapes, and examples are added through this work to simplify the comprehension of the approach and algorithm of this scheme.

Report Organization

This memoir is structured in four chapters, we detailed the contents of the various chapters.

In the first Chapter 1, we walked through the Foundations and basic concepts of Cryptography, listing some of the known algorithms in it, then we disgusted and explained both Block Ciphers and Stream Ciphers and how they work. Following that we went ahead to Chaos-Based Cryptography which is known to have great results in the field, we explained the chaos theory and listed some of the maps that are used in it, and even explained the Architecture that's used for that, we talked about some of the chaos applications and finished the chapter by explaining various of security evaluation measures techniques.

In Chapter 2, we presented the basics of neural networks and showed some types of neural networks with more detail in cellular neural networks. We talked about the application categories of neural networks in cryptography. We presented some references related to neural network cryptography applications with a few details about some of them. Lastly, we presented some references related to a type of neural network cryptography, which is cellular neural network cryptography applications with a few details about some of them.

In Chapter 3, we introduced a Six-Dimensional cellular neural network (6D-CNN) to generate pseudo-random number sequences and we apply it to image encryption. We explained in detail the algorithm of how to generate works. We made an additional proposition to improve and

generate the initial key condition for 6D-CNN. the proposition used Lorenz 3D and Chen's 3D chaotic systems. We did some tests to evaluate the randomness and sensitivity of the generated sequence the prove that it has chaos characteristics. Then we take the generated sequences and employ them to image encryption by explaining each step of the encryption architecture. Lastly, we tested and evaluated the security of this cryptosystem and did a comparison of the test results with other related works to show its performance.

In the last Chapter 4, we introduced new image encryption based on chaos and 6D-CNN pseudo-random sequences generator that was explained in chapter 3. The proposed architecture is based on chaos "Substitution-Diffusion", and we explained in detail each part of the architecture. Lastly, we tested the security of the proposed cryptosystem by doing a comparison with other related works.

The General Conclusion contains an overview of the work done for this master memoir. We presented the strengths of our proposed contributions, the perspectives results, and future objectives work.

CHAPTER 1

BASICS OF CRYPTOGRAPHY

1.1 Introduction to Cryptography

In our time, billions of people use the internet and technologies to communicate with each other, which made it become an important aspect of our daily life to a greater extent of being utilized for business. A lot of us use these technologies on daily basis, for example: to play games, watch movies, read, communicate, and so on. But, at other times we unintentionally download viruses, that make us get hacked or disturb our system which leads to losing data[7]. Nowadays, security issues become a tendency. In regards to the rise in cybercrime, such as identity theft, data theft, service disruptions, hacktivism, and even the threat of terrorism, many businesses are scrambling or seeking various security solutions[8] to avoid these scenarios. And this is where the topic of *Cryptography* comes from.

"*Cryptography* is the science of keeping secrets secret" [9]. Since ancient times, *Cryptography* has been a science that hides information by writing them in a secret code. Around 1900 B.C., an Egyptian scribe used non-standard hieroglyphs in an inscription, which was the earliest known use of cryptography in writing. According to some scholars, cryptography evolved spontaneously after the discovery of writing, with applications ranging from diplomatic messages to battlefield battle plans. It's no wonder that new kinds of cryptography emerged shortly following the widespread evolution of computer communications. *Cryptography* is required in data and telecommunications when interacting over any untrusted medium, which is applied to any kind of network, especially the Internet[10].

Cryptography can be used for user authentication as well as protecting data from theft or change. Secret key cryptography, public key cryptography, and hash functions are the three types of the most commonly used cryptography systems to achieve those goals[10].

For this chapter, we walked through some of the Foundation and Basic Concepts of Cryptography in Section 1.2. After that talked about the Cryptography of images and listed some examples of it in Section 1.3. Following that by explaining the Chaos Theory, Chaos Maps, and its Architecture in image Cryptosystems and chaos Applications in Cryptography in Section 1.4. In Section 1.5, we listed some of the Security Evaluation Measures and explain them. Last but not least, we will draw a general conclusion in Section 1.6

1.2 Cryptography: Foundation and Basic Concepts

1.2.1 Symmetric Encryption Algorithms

In Symmetric encryption, there have been two well-known algorithms or cryptosystems, which are AES which stands for Advanced Encryption Standard, and DES which means Digital Encryption Standard. The DES was contentious because of how it was implemented, and it has mainly been replaced by the AES, which is nearly without controversy. AES is present in widespread usage, in part because of the fact that it is a NIST (National Institute of Standards and Technology) standard, and in part due to its architecture, which makes it quick and useful on a number of systems with varying computing capabilities, and it has been proven to be resistant to all the attempts of finding practical attacks. But, even with all that, both DES and AES are symmetric encrypting algorithms, which means that the key used for encryption a data is the same key used for decrypting that data, which creates a duty on the users to maintain proper key management and security[11].

1.2.2 Block Ciphers

As the name implies, a block cipher is an algorithm that operates on a block of data or block of bits such as 64 or 128 bits, which are converted into identical blocks of the same size with the use of a secret key[12]. Two instances of the same input block will produce identical output blocks when using the basic block cipher with the same key, generating a block of ciphertext from a plaintext block AES, for example, uses 128-bit plaintext blocks to generate 128-bit ciphertext blocks. Generally, the encryption of the block cipher takes a formula like this $c = ENC_k(m)$, with c standing for the cipher text and m for the message of plaintext that we want to encrypt under the use of a secret key ENC_k. As for the decryption, the process of the encryption would be reversed to give the formula $m = DEC_k(c)$.

The block cipher has two important parameters, that are the block size and the key size. A b-bit block cipher maps the set M of 2^b b-bit inputs onto the same set of 2^b outputs for a given key:

$$M = \{ \overbrace{0...10}^{b}, \overbrace{0...00}^{b}, \overbrace{0...01}^{b}, \overbrace{0...11}^{b}, ... , \overbrace{1...10}^{b} \}$$

This is done in such a way that each and every possible output only appears once. The mapping is a permutation of the input set, and we get other permutations by changing the secret key[12]. As a result, a block cipher is a method of producing a family of permutations, which is indexed by a secret key k. The space of all potential permutations that a block cipher might create is determined by the block size b. The amount of permutations that are really created is determined by the key size.

1.2.3 Stream Ciphers

If we dive into a little bit more details in cryptography algorithms, we can see that symmetric cryptography can be divided into block ciphers and stream ciphers. While the block cipher works on the entire block of data at a time as we talked about before, the stream cipher actually encrypts bits individually and this is done by adding a bit from a secret key stream s_i modulo 2 to a plaintext bit x_i, while the ciphertext and the key stream consist of individual bits $(x_i, y_i, s_i \in \{0, 1\})$[13].

- Encryption: $y_i = e_{s_i}(x_i) \equiv x_i + s_i mod 2$
- Decryption: $x_i = d_{s_i}(y_i) \equiv y_i + s_i mod 2$

We can simply portray the basic operation of a stream cipher as shown in Figure 1.1 because the encryption and decryption of it are both just addition modulo 2 which is represented as a circle with a plus inside of it.

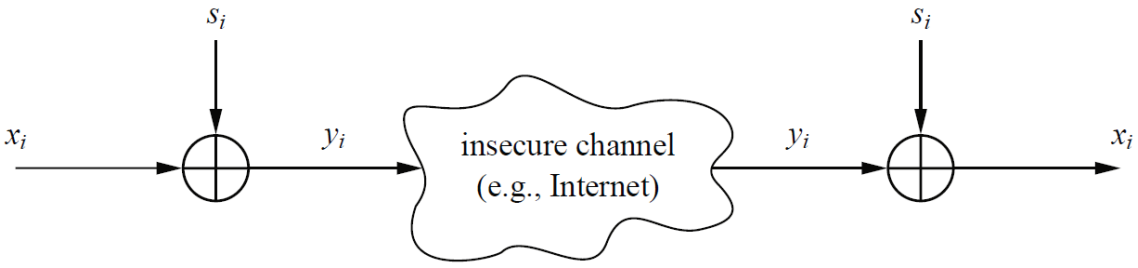

Figure 1.1: Encryption and decryption with stream ciphers

The key stream s_i is actually the main issue for the security of the stream cipher, and the entire security of the stream cipher actually depends on it. Since the key stream bits s_i are not the actual key bits, Stream ciphers are primarily about generating key bits[13]. We can already assume that one of the main requirements for the key stream bits should be that they seem to an attacker as a random sequence. Otherwise, the attacker may guess the bits and decrypt the message on his own.

1.3 Image Cryptography

Images were one of the huge data that needed to be secret. Since cryptography started to get attraction, images were one of the first data that started to be tested with the various methods of cryptography.

1.3.1 Visual Cryptography Scheme

A visual cryptography scheme (VCS), enables the encryption of a hidden image and its secret sharing among many users as shown in Figure 1.2. Visual cryptography was originally invented and pioneered by Moni Naor and Adi Shamir in Eurocrypt 1994[14]. The decryption is carried out utilizing our human vision system without complex computation, the process is quick, and there are no information exchanges or communications between VCS shares[15].

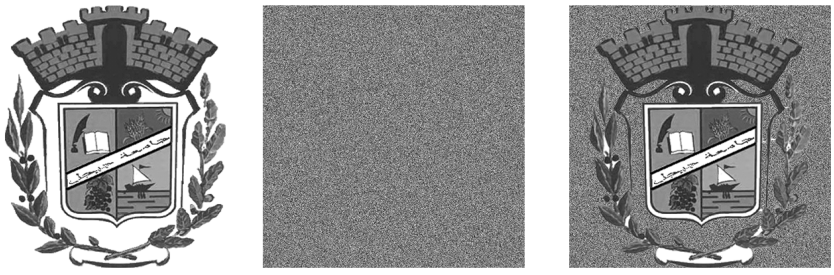

Figure 1.2: Visual Cryptography

1.3.2 Image pixel shuffling technique

In terms of security analysis, a method of shuffling the image's pixel values has proven to be very effective as shown in Figure 1.3. After component shifting, additional pixel swapping in the image file strengthened the security of the image against all currently feasible attacks[16]. A permutation map is used in pixel shuffling to reduce the correlation between adjacent pixels.

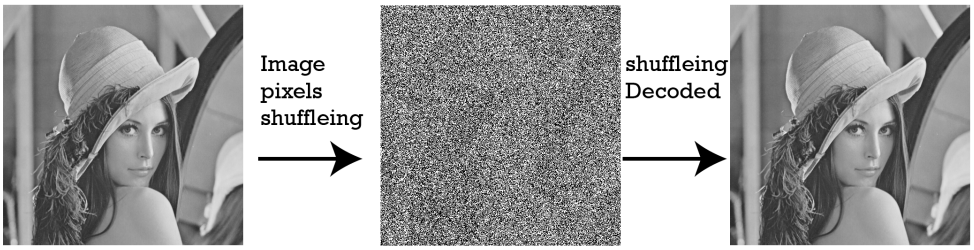

Figure 1.3: Shuffling Encryption Algorithm

1.4 Chaos-Based Cryptography

The interest in cryptography systems has increased a lot over the past few years, and its application became a popular research area and proposed massive encryption algorithm. The majority of simulated chaos passwords employ chaos synchronization technology, which involves chirping the channel in order to achieve secret signaling.

1.4.1 Chaos Theory

The techniques of cryptography that we've talked about in the previous section, are based on algebraic or number theoretic principles. On the other hand, we have chaos that has a lot of potential in the field, which is inspired by the science of nonlinear dynamics. The chaotic behavior of a nonlinear system is a delicate behavior that appears random but has no stochastic base. It's just a science of discovering hidden order in apparently random data. The crucial characteristic of chaos is actually the initial conditions of the system. The fundamental characteristics that contribute to the development of secure communication methods based on chaos are actually the sensitivity of the initial conditions and system parameters[17].

A wide range of chaos-based cryptosystems for end-to-end communications have been proposed, as a consequence of the study shown in Figure 1.4, which shows the Chaotic features and respective cryptography relationships[17].

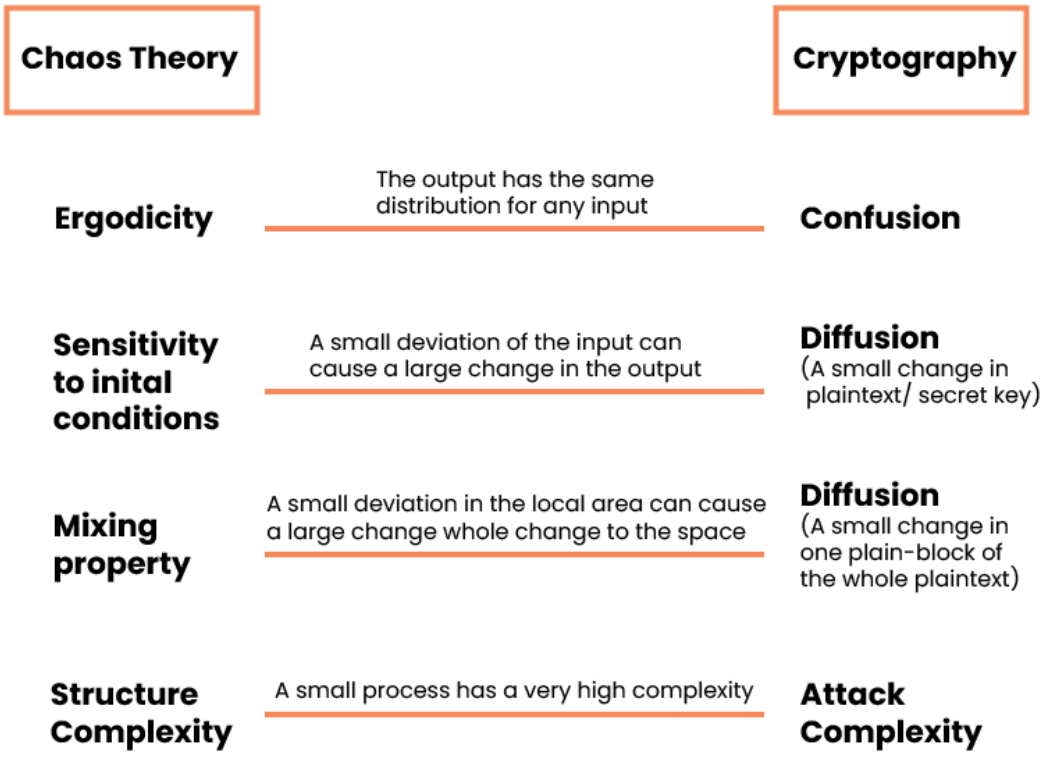

Figure 1.4: Chaos Theory vs. Cryptography

1.4.2 Architecture of Substitution-Diffusion Type Chaos-Based Image Cryptosystems

There are mainly two steps for this kind of type Chaos-Based Image Cryptosystems (Substitution and Diffusion) which are going to be applied multiple times with the use of a factor for each one of them. As shown in Figure1.5.

- Substitution: Without altering the image pixels values, all of them are permuted during the substitution stage according to various transformations[18]. We will talk about some of the Chaotic Maps in more detail in the next section, which is used in this stage.
- Diffusion: After the Substitution stage, the image pixels values are changed sequentially in the Diffusion stage, causing a small change in one pixel to affect the entire image[18]. As a result, the histogram is uniform and appears the same for any plain images.

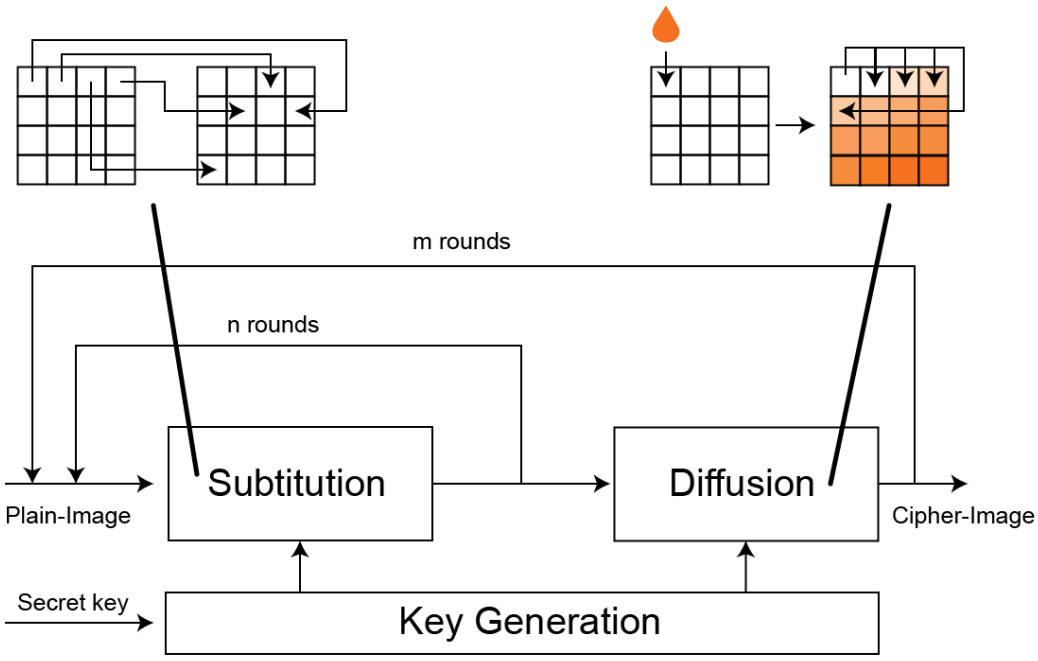

Figure 1.5: Architecture of substitution-diffusion type chaos-based image cryptosystems

1.4.3 Chaotic Maps

Chaotic maps are basically systems that are unstable dynamically with a strong sensitivity to initial conditions, which means that very small deviations in the initial conditions cause significant deviations in the corresponding orbits, thus long-term forecasting for chaotic systems becomes impossible. The chaotic systems are actually distinguished as Entropy producing deterministic systems. These two are ones of commonly-used chaotic maps in image cryptography:

- Baker Map: It compresses the rectangles vertically by a factor of 0.5 while stretching them horizontally by a factor of 2. The unit square is then formed by stacking the right rectangle on top of the left one[18]. It's called Baker map because it looks like making a pastry like shown in the Figure 1.6.

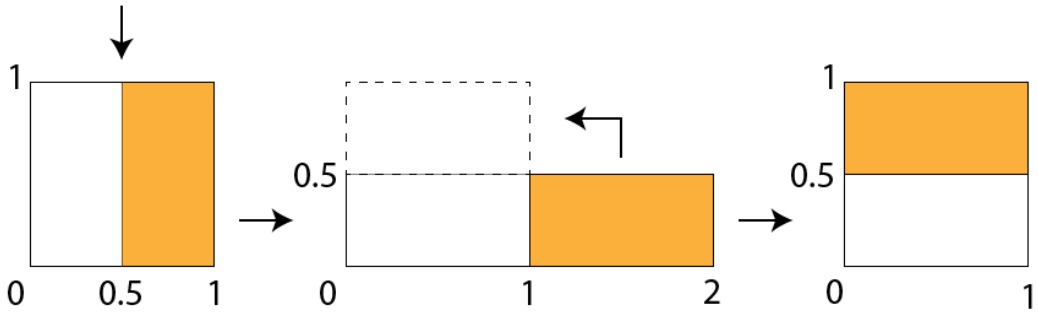

Figure 1.6: Baker Map

- Cat Map: A linear transform stretches the unit square. The mod operation splits the distorted unit square into four pieces and reassembles them to produce the original unit square[18]. The effects of the cat map can be visualized in Figure1.7.

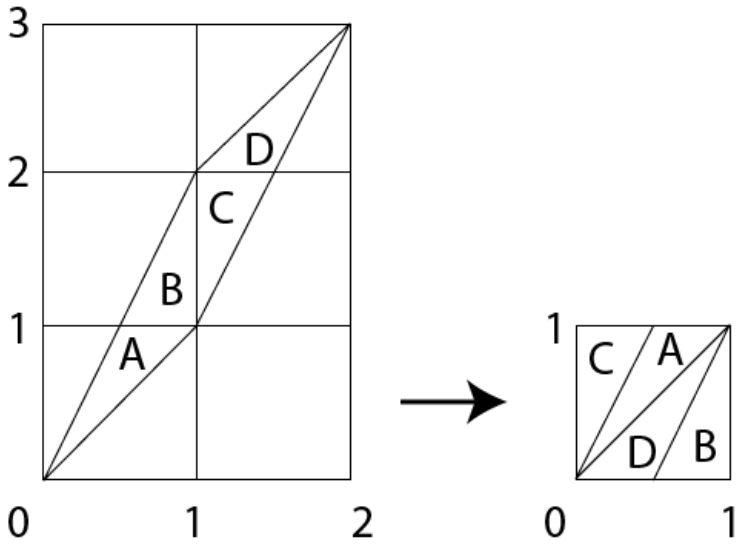

Figure 1.7: Cat Map

- Logistic map: A common image encryption approach is the classical one-dimensional (1D) logistic map, which has a straightforward structure, good chaos results, and desirable auto-correlation and cross-correlation features[19]. The following is its mathematical expression:

$$x_{i+1} = \mu x_i(1 - x_i) \tag{1.1}$$

where $x\epsilon(0, 1)$ and $\mu\epsilon[0, 4]$ are referred to as logistic parameters. The logistic map is in a chaotic state when $x\epsilon(0, 1)$.

- Lorenz System: Lorenz put up a straightforward model in 1963 to explain the weather's erratic behavior. He modeled the motion of a fluid cell that was warmed from below and cooled from above using the fluid convection theory. If x is thought of as the convective fluid motion, then y and z are the horizontal and vertical temperature variations, respectively[20]. then it is possible to formulate Lorenz's equations as:

$$\begin{cases} x' = -\sigma x + \sigma y \\ y' = rx - y - xz \\ z' = xy - bz \end{cases} \tag{1.2}$$

- Chen's Chaotic System: While researching chaotic feedback control, the authors of[21]came across a system that exhibits more intricate dynamic behaviors than the Lorenz system. Compared to the Lorenz system, the chaotic attractors of Chen's system have more varied and complicated dynamic properties. The chaotic system described by Chen's mathematical model is stated as:

$$\begin{cases} x' = a(y - x) \\ y' = (c - a)x - xz + cy \\ z' = xz - bz \end{cases} \tag{1.3}$$

where a, b, and c are system parameters, when a = 35, b = 3, and c = 28, the system is in a chaotic state[22].

1.4.4 Chaos Applications in Cryptography

Recent research in the field of nonlinear dynamics, particularly in the domain of systems with chaotic behavior, has prompted a number of investigations into practical applications of such systems.

1.4.4.1 Analog Chaos Encryption

The majority of conventional analog chaotic-based secure communication systems fall into one of three basic categories: chaotic modulation, chaotic switching (also known as chaotic shift keying, or CSK), and chaotic masking[23]. Although other new designs have been put forth in recent years, the majority of them are really modified or generalized versions of these three fundamental ideas.

1.4.4.2 Digital Chaos Encryption

It takes a numerical solution for integration to apply a chaotic function digitally. Numerical methods like Euler's method or Runge-Kutta are typically used in commercial software like Matlab or Mathematical to solve these problems[24]. The security of digital images is extremely important, and the development of picture encryption algorithms is becoming more and more popular because of their unparalleled intuitiveness and volume of information.

1.4.4.3 Numerical chaos Encryption

In view of physical disturbances, numerical simulations of chaotic dynamical systems can be trusted if they provide accurate solutions to issues that are "sufficiently close" to the mathematical model of the actual problem being studied[25]. The Gauss map from the theory of continuous fractions serves as a good didactic example since it makes backward error analysis for discrete dynamical systems easy to understand and illustrates the consequences of floating-point arithmetic. It's defined as: $G : [0, 1) \to [0, 1)$

$$G(x) = \begin{cases} 0, & if \;\; x = 0, \\ x^{-1} mod 1, & otherwise, \end{cases} \tag{1.4}$$

1.5 Security Evaluation Measures

To confirm the efficiency of an encryption approach, evaluation measures are essential. These parameters are used to investigate the various qualities of an encryption technique.

1.5.1 Key Space Analysis

Security keys are an essential component of every encryption technique since the algorithm's strength is dependent on them. The secret keys should be able to withstand any form of attack. The key space for an effective and secure image cryptography system should be large enough to prevent a brute-force search attack. Even if the encryption and decryption keys change just slightly, a high key sensitivity assures that no part of the plain image can be retrieved.

1.5.2 Histogram Analysis

A histogram is a representative graphic of numerical data distribution. In the image, the histogram shows the distribution of pixel values. The original image's histogram should be

completely different from the encrypted image's histogram. The cipher-histogram image is plotted to check if it is suitably uniform. For any plain images, an efficient image encryption technique should always create a cipher-image with a uniform histogram[18]. In the encrypted image histogram, all pixels should be distributed equally in the graph.

1.5.3 Correlation Analysis

The correlation coefficient is used to compare matching pixels in an encrypted and the original image. In the three directions, horizontal, diagonal, and vertical axes, the values of neighboring pixels in an original image are tightly connected. A decent image encryption algorithm lowers this connection in the ciphered image[26]. The steps below are used to examine the correlation between two nearby pixels.

To begin, randomly choose pairs of two horizontally, vertically, or diagonally adjacent pixels from the image and use the following formulae to compute the correlation coefficient r_{uv} of each pair.

$$r_{x,y} = \frac{C(x,y)}{\sqrt{D(x)}.\sqrt{D(y}} \tag{1.5}$$

where,

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

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

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

x and y are the values of two adjacent pixels (horizontally, vertically, or diagonally) in an image. $cov(x,y)$ represent the covariance, where $D(x)$ refers to the variance of variable x, and $E(x)$ is the expectation of variable x. the range of to correlation is $r_{x,y} \in [-1,1]$.

Plain-image correlation coefficients are typically large (near to 1) since adjacent pixels in natural images have significant similarities. However, while better-ciphered image cryptosystems should decorrelate the connection between adjacent pixels, those of cipher-image should be very low (near to 0).

1.5.4 Information Entropy

It determines the average amount of information per bit in an image. It provides all of the information that is available in the image. Every pixel has a unique value. As a result, the entropy of an encrypted image signifies that each pixel has a uniform probability distribution[27]. The information entropy (IE) goes by the equation 1.6.

$$H(m) = -\sum_{i=0}^{L-1}P(m_i) \times log_2 P(m_i) \tag{1.6}$$

$H(m)$ is the entropy and m is the message source. m_i indicates to the pixel values, L is the total number of pixel values, and $p(m_i)$ denotes the possibility of a pixel with value m_i

occurring. The histogram of a cipher image is regarded sufficiently uniform if its entropy is near to log L bits. The result of IE should be in the range of $IE \in [0, 8]$.

The pixel values are evenly distributed when the entropy values are close to 8, and the password is not vulnerable to statistical attacks. As a result, a good encryption system should add enough randomness to the image that the entropy approaches the theoretical value of 8.

1.5.5 Plaintext Sensitivity Analysis

When one or more pixels of plaintext are changed, the original plaintext and the slightly changed plaintext are encrypted separately. The attacker can use the differential attack to decipher the key if the change in position or in the degree of the acquired two ciphertext pixels is not clear or follows a regular pattern. As a result, a good encryption method should be very sensitive to changes in the original images in order to successfully withstand differential attacks. The number of pixel change rates ($NPCR$) and the unified average changing intensity ($UACI$) are commonly used to assess a system's capacity to withstand different attacks.

$NPCR$ represents the difference between pixel numbers of the two encrypted images. It's the proportion of changed pixel numbers between two encrypted images with only a one-pixel difference in plain images[28]. which is defined by the following equation 1.7 :

$$NPCR = \frac{\sum_{i,j} D(i,j)}{W \times H} \times 100 \tag{1.7}$$

where,

$$D(i,j) = \begin{cases} 0, & \text{if } E(i,j) = E'(i,j) \\ 1, & \text{if } E(i,j) \neq E'(i,j) \end{cases} \tag{1.8}$$

W represent the width and H the height of the image. $D(i,j)$ reflect the difference between corresponding pixels of the encrypted image of the original image $E'(i,j)$ and encrypted image of the modified image $E(i,j)$ shown in equation 1.8. The result of $NPCR$ will be in range of $NPCR \in [0, 100]$.

$UACI$ calculates the average intensity of difference between two encrypted images, which corresponds to a one-pixel difference in plain images[29]. $UACI$ generalized by the equation 1.9:

$$UACI = \frac{\sum_{i,j} E(i,j) - E'(i,j)}{255 \times W \times H} \times 100 \tag{1.9}$$

The original and encrypted images are both represented in equation 1.9 by $E(i,j)$ and $E'(i,j)$.

The encryption method passes the security criterion when $NPCR$ reaches around 99.6% and $UACI$ reaches approximately 33.4%[30].

1.5.6 Robustness Analysis

The attacker may add noise to the encrypted image to corrupt the relevant information. As a result, the intended user will be unable to correctly recover the original image after the decryption. In the encrypted image, the attacker uses additive, Poisson, Gaussian, and other

types of noise. As a result, a good image encryption method should be resistant to noise attacks[31].

1.6 Conclusion

The evolution of cryptography has demonstrated that it must closely follow the rate of technological advancement since stronger cryptanalysis attacks are now possible due to computing power and hardware becoming more affordable.

Undoubtedly, the advancement of cryptography shows how adaptable and progressive humans are. Cryptography will evolve as technology, the economy, and political change. As a result, cryptography will only continue to advance in order to provide unbreakable security, time and money efficiency, and support for the broadest range of applications and environments.

A noticeable advancement in cryptography is the applications of neural networks in them, which have shown some great results in the field. We will talk about these applications in more detail in Chapter 2.

CHAPTER 2

APPLICATION OF NEURAL NETWORK IN CRYPTOGRAPHY

2.1 Introduction

Simulating intelligent tasks carried out by the human brain led to the development of Artificial Neural Networks (ANNs). They are primarily used by soft computing methods since they can simulate complicated input/output connections in any system. The advantages of ANNs include their quick operational reaction time, high degree of structural parallelism, accuracy, and efficiency. They can also generalize outcomes from known situations to difficult situations. ANNs can train and perform well if a set of input-output data pairs related to an issue is available. Due to the potential for their adaptive behaviors to easily mimic severely nonlinear properties, applications of ANNs have become an attractive field of research.

The Neural Networks method provides an impressive solution to cryptography problems. In that, it offers an effective approach within which data can be easily encrypted. Numerous research has looked at various machine learning techniques, particularly neural networks, and their use in cryptosystems.

The structure of this chapter is as follows: In Section 2.2 we talk about the basics of artificial neural networks and show some examples of other types of neural networks. In Section 2.3 we explain and describe the application and categories of neural networks in cryptography. Section 2.4 shows some previous works on cryptography based on neural networks. Finally, we draw a general conclusion in Section 2.5.

2.2 Basics of Neural Networks

2.2.1 Artificial Neural Network

An Artificial Neural Network (ANN) is a simplified representation of the biological nervous system. An ANN is a massively parallel distributed processing network with a large number of processing components called neurons with a design inspired by the brain that has a natural greater propensity for storing experiential information and making it available for later use. Each neuron is linked to other neurons by directed communication connections, each has a weight associated with it. Each neuron has an internal state known as activation or activity

level, which is determined by the inputs it receives. A neuron's activity is often sent as a signal to multiple other neurons. The neurons can be connected using a variety of designs. It can be utilized as a mapping function by selecting the right model and properly training the network. Feedforward networks utilizing the backpropagation learning method are applied in a variety of architectures.

The capabilities of perceptron and other one-layer networks in backpropagation neural networks are severely limited. To solve these constraints, multilayer Feedforward networks with Backpropagation learning and non-linear node functions are utilized. Multiple layers make form a multilayer feedforward network. This class of architectures, in addition to having an input and output layer. The neurons in one layer were connected to the neurons in the next layer, and so on until the output layer was reached as shown in Figure 2.1. Before sending the input to the output layer, the hidden layer assists in performing useful intermediary calculations. Backpropagation neural nets are feedforward networks that are trained using the backpropagation learning method. The feedforward of the input training pattern, the computation and backpropagation of the associated error, and the modification of the weights are the three steps of backpropagation network training. The final weights are saved in a file once the process has converged. After training, the network's application is limited to the feedforward phase calculations.

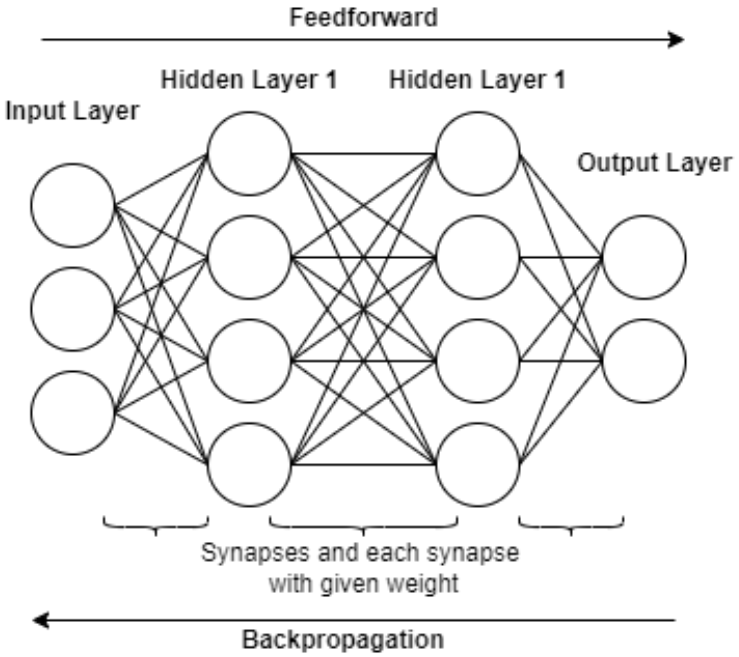

Figure 2.1: Artificial Neural Network

2.2.2 Types of Artificial Neural Network

Artificial neural networks come in many different types. These networks are constructed using a set of parameters and mathematical procedures that determine the output. Next, we will talk about some of them:

2.2.2.1 Recurrent Neural Networks

Recurrent Neural Networks (RNNs) are networks in which neurons send feedback signals to each other, such as Hopfield networks, Elman and Jordan networks, long short-term memory

RNNs, and bidirectional networks. When compared to forward networks, this allows for the modeling of dynamic behaviors with the disadvantage of consuming more memory.

2.2.2.2 Chaotic Neural Networks

Chaotic neural networks have a considerable memory capacity. Each memory is encoded by the unstable periodic orbit (UPO) of the chaotic attractor. A chaotic attractor is a collection of states in the state space of a system that has the unique attribute of being an attracting set. As a result, the system begins to move in the suitable basin and finishes up in the set. The most important is that once the system is on the attractor, neighboring states diverge exponentially quickly from each other, amplifying tiny quantities of noise. The biases and weights of neurons are determined using a binary sequence obtained from a chaotic environment.

2.2.2.3 Convolution Neural Networks

A Convolution neural network is a type of deep structure feedforward neural network that includes convolution processing and is one of the deep learning's representative algorithms. Convolutional Neural Networks perform well in image recognition, video analysis, natural language processing, and many other applications. Multiple continuous convolution layers and pooling layers are typically used in Convolution neural networks. Through convolution and pooling processes, Convolution neural network can automatically learn various layers of image features. The convolution neural network convolutional and pooling layers constantly extract and compress image features to ultimately obtain high-level image features. We can utilize the characteristics we've gathered to perform tasks like classification and regression. It can also be used to manage scrambling processes in encryption algorithms to protect against plaintext attacks that are known or selected.

2.2.2.4 Neural Cryptography

Is a field of cryptography that studies the use of stochastic algorithms in encryption and cryptanalysis, particularly neural network algorithms. The capacity of neural networks to selectively explore the solution space of a problem is widely recognized. In the domain of cryptanalysis, this property has a natural application niche. At the same, Neural Networks provide a novel approach to attacking ciphering algorithms, based on the principle that any function can be recreated by a neural network, which is a powerful and well-proven computational tool that can be used to find the inverse function of any cryptographic algorithm. Mutual learning, self-learning, and stochastic behavior of neural networks and similar algorithms can be applied to various aspects of cryptography, such as public-key cryptography, solving the key distribution problem with neural network mutual synchronization, hashing, and the generation of pseudo-random numbers.

2.2.2.5 General Regression Neural Networks

It is presented a memory-based network that offers continuous variable estimates and converges to the underlying (linear or nonlinear) regression surface. The generic regression neural network (GRNN) is a highly described one-pass learning technique. The approach is proven to offer smooth transitions from one observed value to another, even with sparse data in multi-dimensional measurement space. Any regression issue in which the assumption of linearity is not justifiable can be solved using the algorithmic approach.

2.2.2.6 Cellular Neural Networks

Chua[32;33] introduced in 1988 a circuit architecture class of information processing system known as Cellular Neural Networks (CNN). Cells are the fundamental key component of cellular neural networks. It has both linear and nonlinear circuit components, such as independent sources, linear capacitors, linear resistors, and linear and nonlinear controlled sources. CNNs have a structure that is similar to cellular automata in that each cell in a cellular neural network is only connected to other cells that are nearby to it. Direct interaction between the neighboring cells is possible. Because of the propagation effects of the continuous-time dynamics of CNNs, cells that are not directly connected to one another may yet have an indirect impact on one another. Figure 2.2 shows an illustration of a two-dimensional cellular neural network.

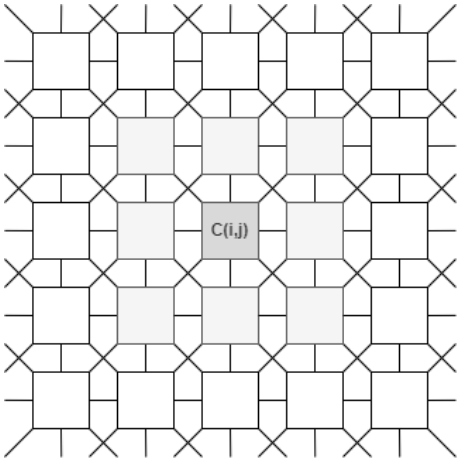

Figure 2.2: Structure of a Typical CNN Model in Two-Dimensional

$C(i,j)$ represent the cell in row i and column j. Consider an $(M \times N)$ cellular neural network with $(M \times N)$ cells organized in M rows and N columns, the neighborhood of a cell C(i,j) in a CNN is defined by:

$$N_{ij}(r) = C_{ab}|max(|a-i|,|b-j|) \leq r, 1 \leq a \leq M, 1 \leq b \leq N \tag{2.1}$$

Where r a positive integer number represent the number of neighborhood of a cell $C(i,j)$. Figure 2.3 shows some examples of of neighborhood of cell C_{ij} with different r value.

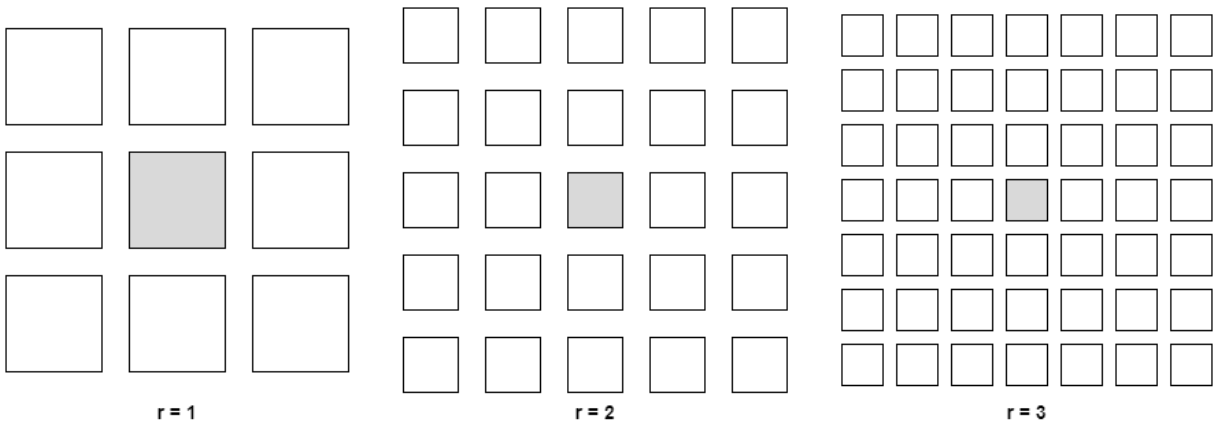

Figure 2.3: Neighborhoods of cell C_{ij} where $r = 1$, $r = 2$, and $r = 3$

According to Figure 2.4, every CNN cell has an equivalent circuit. The input, state, and output parameters of the cell are denoted by the letters u, x, and y, respectively. The initial

amplitude value of the node voltage, x_{ij}, which represents the cell's state, is not more than 1. The input of the cell is represented by the node voltage u_{ij}, which must have a constant amplitude and be less than 1.

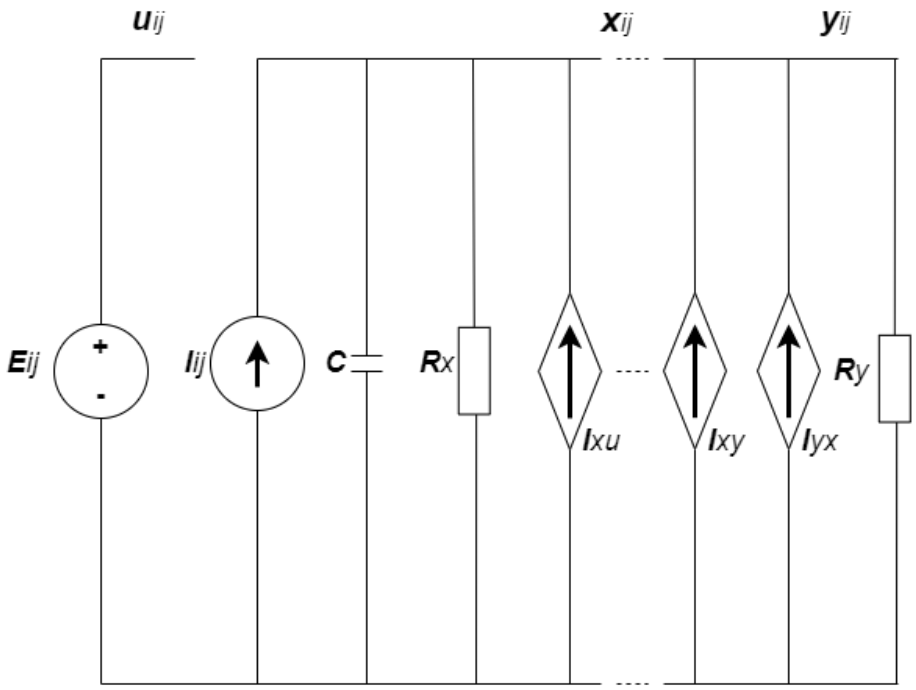

Figure 2.4: An example of a cell circuit of cell C_{ij}

A cell is defined as a circuit that can be described by the first order nonlinear differential equation 2.2.

$$C\frac{dx_{ij}(t)}{dt} = -\frac{x_{ij}(t)}{R_x} + \sum_{k,l\in N_{ij}(r)} A_{kl}y_{kl}(t) + \sum_{k,l\in N_{ij}(r)} B_{kl}u_{kl} + I_{ij} \tag{2.2}$$

In equation 2.2 x_{ij} is a state variable, y_{kl} is the outputs of cells, u_{kl} is the input of cells, C and R_x are system constants, I_{ij} is the threshold, B is the control parameter matrix, while A is the matrix of feedback parameters. The matrix elements are indicated in the equation by the subscripts that follow the matrices. The behavior of CNN is determined by these parameter matrices. The CNN output equation is provided by:

$$y_{ij}(t) = \frac{1}{2}(|x_{ij}(t)+1| - |x_{ij}(t)-1|) \tag{2.3}$$

2.3 Application of Neural Network in Cryptography

In recent years, there has been an increased interest in the use of several types of neural networks to cryptography issues and challenges. The use of neural networks in cryptosystems has been in lots of recent research. Key management, generation, and exchange protocols are common examples, as are steganalysis, pseudo-random generators, visual cryptography, and digital watermarking[34;35].

2.3.1 Steganalysis

Steganography is a method of secret communication, whereas steganalysis is the science of analyzing secret messages contained in digital information. Law enforcement and the media

have both focused on steganography and steganalysis[36]. Nowadays, digital steganography focuses on keeping hidden information among redundant image bits that will be sent. The principal objective is to increase the usage of steganography so that the attacker cannot secretively enter the message or discover that a secret message is encrypted. In the past few years, a lot of studies have applied neural networks in steganalysis[37–39]. At first, various approaches are used to evaluate communication such as discrete cosine transformation analysis and wavelet texture decomposition and analysis. Afterward, neural networks are used to classify images based on whether or not they contain secret information. They generate different weight sequences depending on whether or not they have been directly affected by an image hiding technique. The process of hiding data inside an image is non-linear. Neural networks beat most simple linear classifiers because of their excellent capacity to learn from training data to estimate non-linear issues.

2.3.2 Digital Watermarking

Since the birth of public media communication, secure media transmission has been a field of study. Watermarking is a technique for identifying who owns the copyright of digital information including audio, images, and videos. To enable authentication and content protection, the approach requires integrating a digital signature within the media. Neural networks have qualities that help to increase the performance of watermarking techniques. The watermarked image is created by combining the original image with a randomly generated watermark. To extract relevant coefficients, a wavelet decomposition approach is performed initially. In a neural network-based technique, the extra information is used to train the network. After that, the neural network successfully extracted enough data to estimate the watermark. This data was utilized to create the image attached. The effectiveness of such neural network-based watermarking techniques has varied depending on the approach utilized. Successful techniques have achieved acceptable outcomes, such as greater accuracy offered by neural networks' adaptive decision-making capacities, as well as better algorithm robustness against various types of attacks.

2.3.3 Visual Cryptography

Noar and Shamir were the first to introduce the visual cryptography approach in 1994[40]. Visual cryptography is a type of cryptography that allows visual information, such as text and images, to be encrypted in a way that decryption may be accomplished by the human visual framework without the need for computers[41]. Visual cryptography is a method for securely transferring visual secrets over a public network. The neural network applied in visual cryptography is known as the quantum neural network (Q'tron) model. The neural network takes a hidden image in a gray type of format as input. At the time that the neural network starts to relax, the first output is a series of binary shadow images. The method of obtaining such binary images from grey images is known as image half-toning. The purpose of image halftoning is to generate a binary image that looks very similar to its grey tone version and is seen as such by just blurring one's eyes. The secret gray tone image is divided in Q'tron neural network cryptography in such a way that no one halftone image shows any part of the target image, but stacking each subset of shares defined in the access method generates a halftone image that mimics the gray tone target-image. While none of the stacked images are gray tone images, the conclusion is that the decryption is done by the human eye, without the assistance of a computer. Another valuable feature of the Q'tron neural network is its ability to transpose past stages automatically. The neural network may then be used to recreate the original gray tone target image by reversing the Q'trons operations, removing the need for human interaction,

and totally automating the process.

2.3.4 Secret Key Protocols

Describes a neural key exchange protocol that relies on tree parity machines synchronizing. The tree parity machine is a sort of multi-layer feed-forward neural network that consists of I input neurons, N hidden neurons, and W weight range[42]. In each stage, the two neural networks get the same random inputs and learn their identical outputs. The notion of synchronization via continuous learning is established as a result of this[43]. A time-dependent weight vector is used to synchronize two machines. This approach was used to create a safe secret key for synchronization. The single secret key obtained is used to encrypt and decrypt private information. Any algorithm, such as Advanced Encryption Standard (AES), can be used for encryption and decryption.

2.3.5 Pseudo-Random Number Generator

The pseudo-random number has a great benefit and does great help in network security, data encryption, image transmission, satellite navigation, and several other things. The study of an algorithm that can generate a random number with a high level of randomness has always been a hot subject in the field of information security. The pseudo-random number has been applied in several types of neural networks, in artificial neural networks, recurrent neural networks, cellular neural networks, and so on. Neural networks have excellent generalization capabilities after being trained on numerous well-known input vectors, allowing them to give reasonable output to complex numbers, provided the input pattern is recognized. When the network is over-fitted, it will be unable to predict the input pattern when it receives unknown input patterns, resulting in unpredictable outputs[44]. Multi-layer perception (MLP) neural networks could be used as a powerful independent random generator or as a technique of improving existing generators by feeding pseudo-random numbers generated by linear computational generators into neural networks.

2.4 Literature Review

2.4.1 Previous Works on Cryptography Based on Neural Networks

Recently, there is a lot of literature and studies that indicate the usage of neural networks in cryptography. This shows that there is an increase of interest in this application of different classes of neural networks to problems related to cryptography in the past few years.

Man et al.[1] presented in 2021, a double image encryption technique based on dynamic adaptive diffusion and Convolutional Neural Network. Compared to the current double image encryption technique, this approach is different. They developed a dual-channel (digital channel / optical channel) encryption method in accordance with the properties of digital images, which not only guarantees the security of double images but also increases encryption strength and lowers the risk of being attacked. To increase the security of the key, the initial values of a five-dimensional conservative chaotic system are first controlled using a chaotic map. Additionally, we use a chaotic sequence as the convolution kernel of a convolution neural network to create chaotic pointers that are connected to plaintext in order to successfully resist known-plaintext attacks and chosen-plaintext attacks. This allows us to decide how two images are scrambled. A new image fusion technique is developed on the basis of this, which separates and fuses two images into two separate sections based on the amount of information they contain.

Additionally, a dual-channel image encryption method with optical and digital channels is created for the two aspects following its fusion. The presented scheme's precise implementation methods are presented in Figure 2.5.

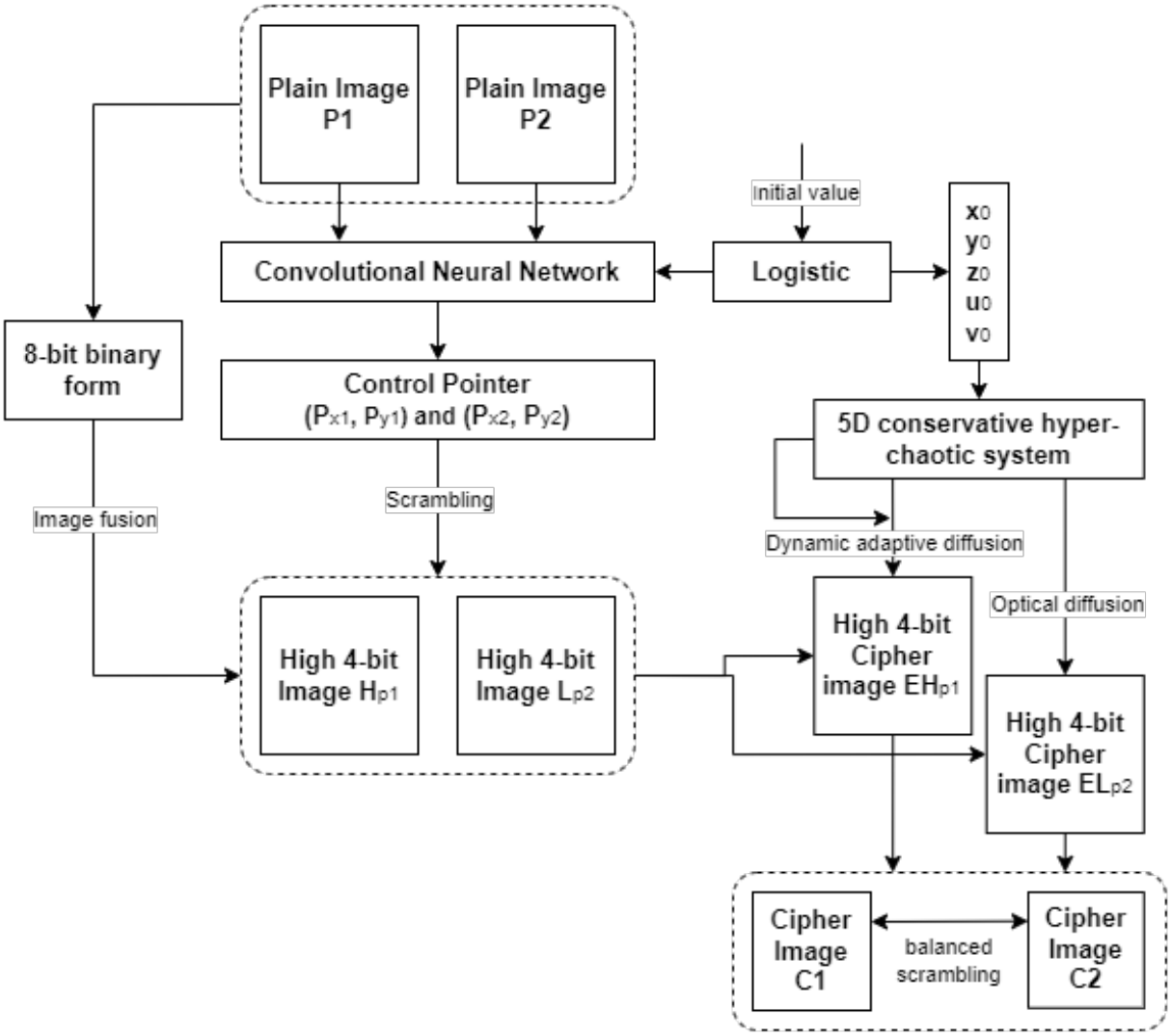

Figure 2.5: Man et al.[1] Proposed Encryption Flow Chart.

Dridi et al.[2] introduces in 2016, a new chaotic-neural network for image encryption/decryption used in the medical field. The main purpose of the suggested method is to protect medical images while using an algorithm that is less complicated than those used by current techniques. All pixels connected to the host image are XORed with a generation key to increase robustness. Then after, a chaotic system (logistic map) is used to generate the binary sequence. In order to determine the weights w_{ij} and bias b_i of the neural network and encrypt the pixels produced in the previous phase. The flow chart of this literature is generalized in Figure 2.6.

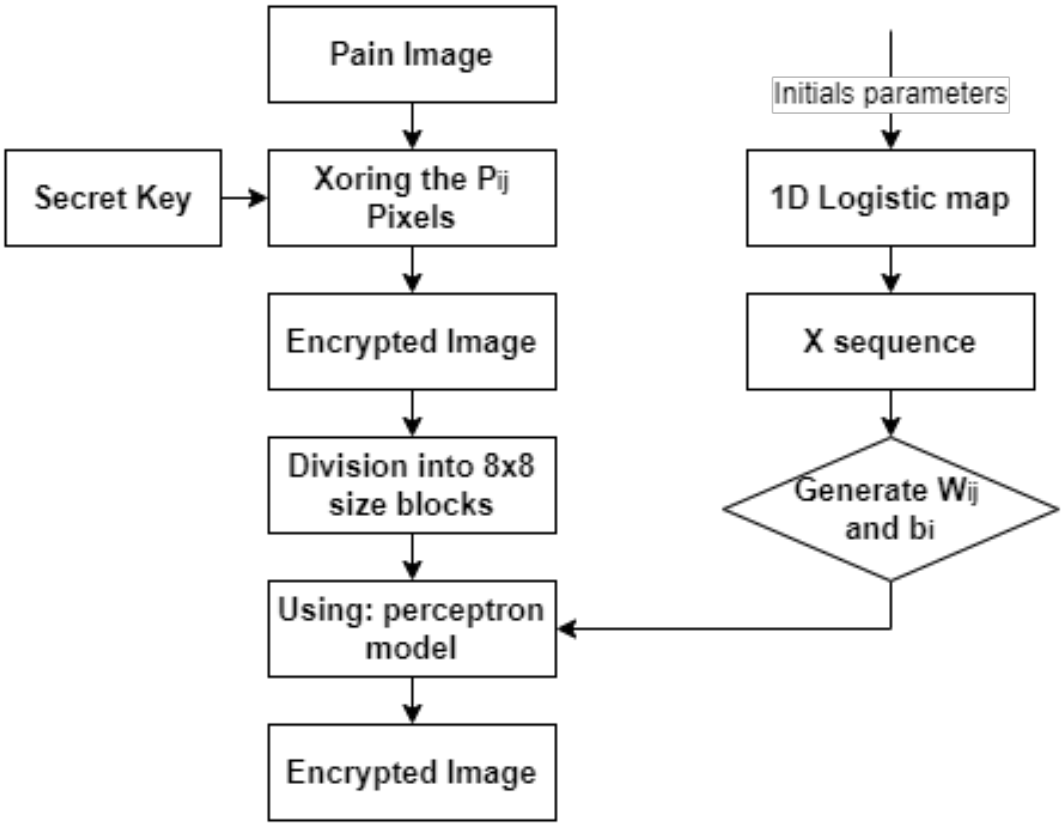

Figure 2.6: Dridi et al.[2] Proposed Encryption Flow Chart.

Patel et al.[3] proposed in 2021, a random number generator for cryptography created by combining the highly chaotic properties of hybrid chaos maps with neural networks. The control parameters and iteration value of the two-hybrid chaotic map are constructed in this study as a layer transfer function to achieve high unpredictability. The obtained sequences and deoxyribonucleic acid encoding technology are used to produce colored image encryption. The explained scheme is generalized in Figure 2.7.

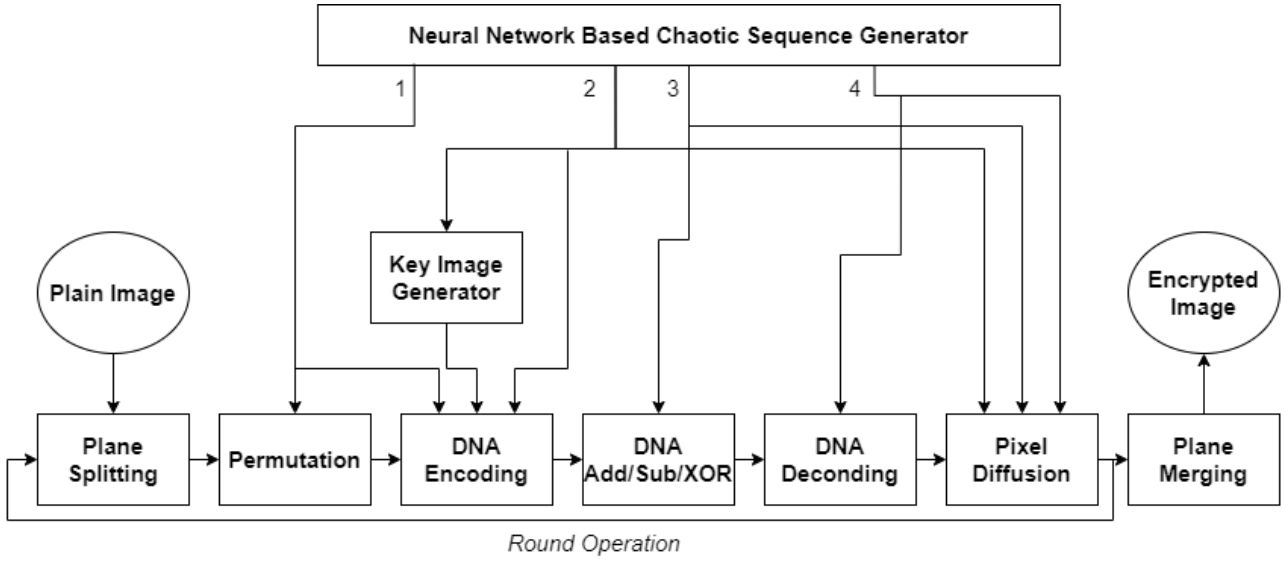

Figure 2.7: Patel et al.[3] Proposed Encryption Flow Chart.

We represent the description of some other previous works on cryptography based on the Neural Networks method in the following Table 2.1.

Table 2.1: Previous Works on Cryptography Based on Neural Networks

Type	Ref	Author(s)	Year	Description of the method
Artificial Neural Nets	[45]	Joshi et al.	2012	proposed a new image encryption and decyption using artificial neural networks.
	[46]	Saraswat et al.	2019	utilized the soft computing idea of auto associative neural networks in combination with encryption techniques to transfer data safely over a communication network.
	[47]	Volna et al.	2012	This article is about utilizing neural networks in cryptography and creating neural networks that can be utilized in the field of cryptography. An experimental demonstration is also included in this article.
	[48]	Valencia et al.	2022	Deep learning techniques are being used to create a cryptographic system. By employing the synaptic weights of an autoencoder neural network as encryption and decryption keys, the solution removed the need for large prime numbers.
Convolutional Neural Nets	[1]	Man et al.	2021	introduced a convolutional neural network and dynamic adaptive diffusion-based double image encryption technique.
Neural Cryptography	[49]	Dong et al.	2019	proposed a complex-valued tree parity machine network-based neural cryptography (CVTPM).
	[50]	Jeong et al.	2021	presented the Vector-Valued Tree Parity Machine (VVTPM), an extended architecture of TPM models that is more efficient and safe for real-world systems.
	[51]	Pattanayak et al.	2017	proposed a novel model for encrypting/decrypting a secret code using Neural Networks, as opposed to previous private key cryptography models that were based on theoretic number functions.
	[52]	Zhu et al.	2018	developed a neural cryptography scheme based on a new topology changing neural network architecture known as the Spectrum-diverse unified neuro evolution architecture.
Chaotic Neural Nets	[53]	Bigdeli et al.	2012	introduces a new image encryption/decryption system based on a chaotic neural network.
	[2]	Dridi et al.	2016	introduced a novel chaotic neural network for image encryption and decryption in the medical field.
	[54]	Yu et al.	2006	used a chaotic neural network to generate binary sequences for masking plaintext.
	[55]	Maddodi et al.	2018	a novel mixed neural network and chaos-based pseudo-random sequence generator, as well as a chaotic encryption technique based on DNA rules for safe image transmission and storage.

2.4.2 Previous Works on Cryptography Based on Cellular Neural Networks

A single chaotic system encryption scheme can no longer meet the security and real-time requirements of modern image communication rhythms. The security and real-time needs of the current daily rhythm for picture transmission can no longer be met by a single chaotic system's encryption technique. When Cellular Neural Network (CNN) approach is implemented to the chaotic encryption method, researchers noted that hyperchaotic systems exhibit more complicated dynamic characteristics, including strong randomness, unpredictability, and greater security performance[56].

CNN is a nonlinear analog processor with double-valued output signals that are locally connected[32;33]. It is an artificial neural network created by combining the Hopfield neural network with cellular automata[57], which uses the local connection of cellular automata while also resolving the Hopfield neural network's difficulties on hardware. CNN not only has complex chaotic dynamics properties as a flexible and efficient local interconnection network, but it can also be simply integrated on the very large scale integration (VLSI)[58]. CNN has a larger key

space and superior permutation and diffusion features than classical chaotic systems. s a result, CNN is commonly used in encryption systems[59], and it has demonstrated effective encryption results in digital watermarking[60], voice encryption[61], and image encryption[6].

Wang et al.[4] introduced 2020, a new method for encrypting chaotic images. First, An original phased composite chaotic map is implemented. The comparison analysis demonstrates that the cryptographic properties of the map are preferable to those of the logistic map, and the map is utilized as the Fisher-Yates scrambling controller. Second, the fractional-order 5D cellular neural network system is utilized as a diffusion controller in the encryption process due to its higher level of complexity. We can get the final ciphertext by combining the secret key, mapping, and plaintext. Figure 2.8 shows the encryption procedure used in this work.

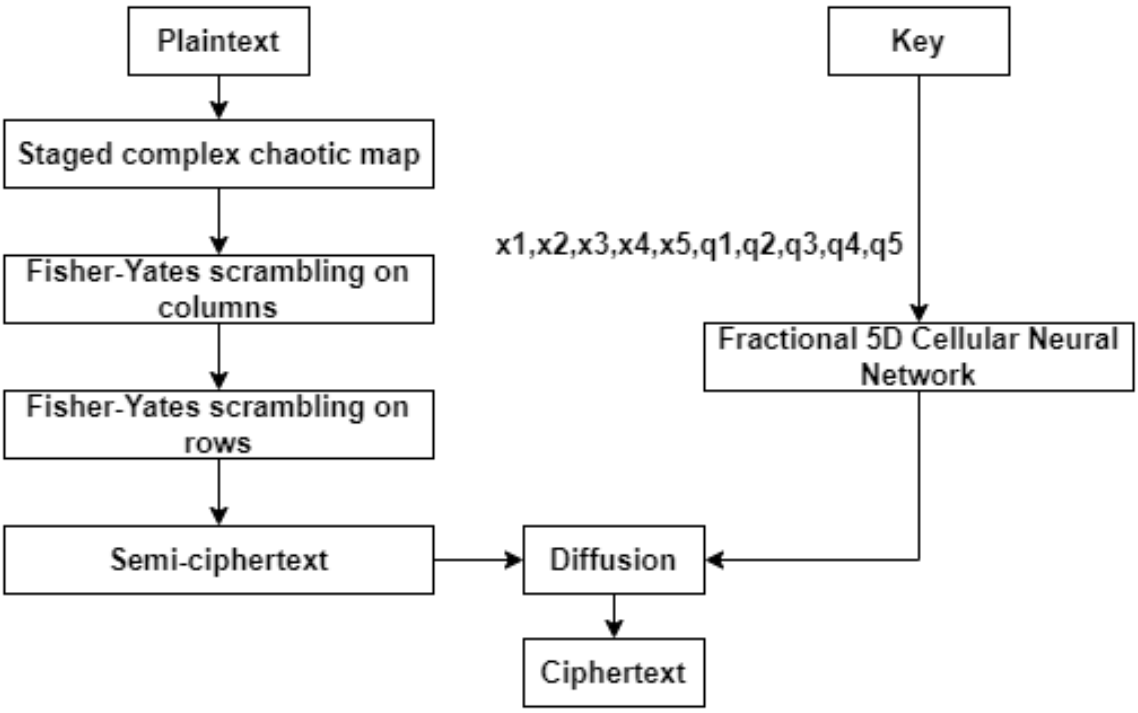

Figure 2.8: Wang et al.[4] Proposed Encryption Flow Chart.

Sheela et al.[5] presented in 2020, a new cryptosystem secure medical images in teleradiology applications. The presented cryptosystem is based on the Fridrich architecture, which performs cryptographic operations using hyperchaotic cellular neural networks (CNN) and DNA technology. The cellular neural network crumb coding transform (CNN-CCT) is suggested in the literature as a method to carry out confusion operations. It is utilized to generate random pixel values. The cipher block chain (CBC) method of XOR operation, which offers higher performance on hardware platforms, is used to perform the diffusion process. The pixel values are changed using the diffusion process to increase security. Figure 2.9 shows a diagrammatic depiction of the proposed method.

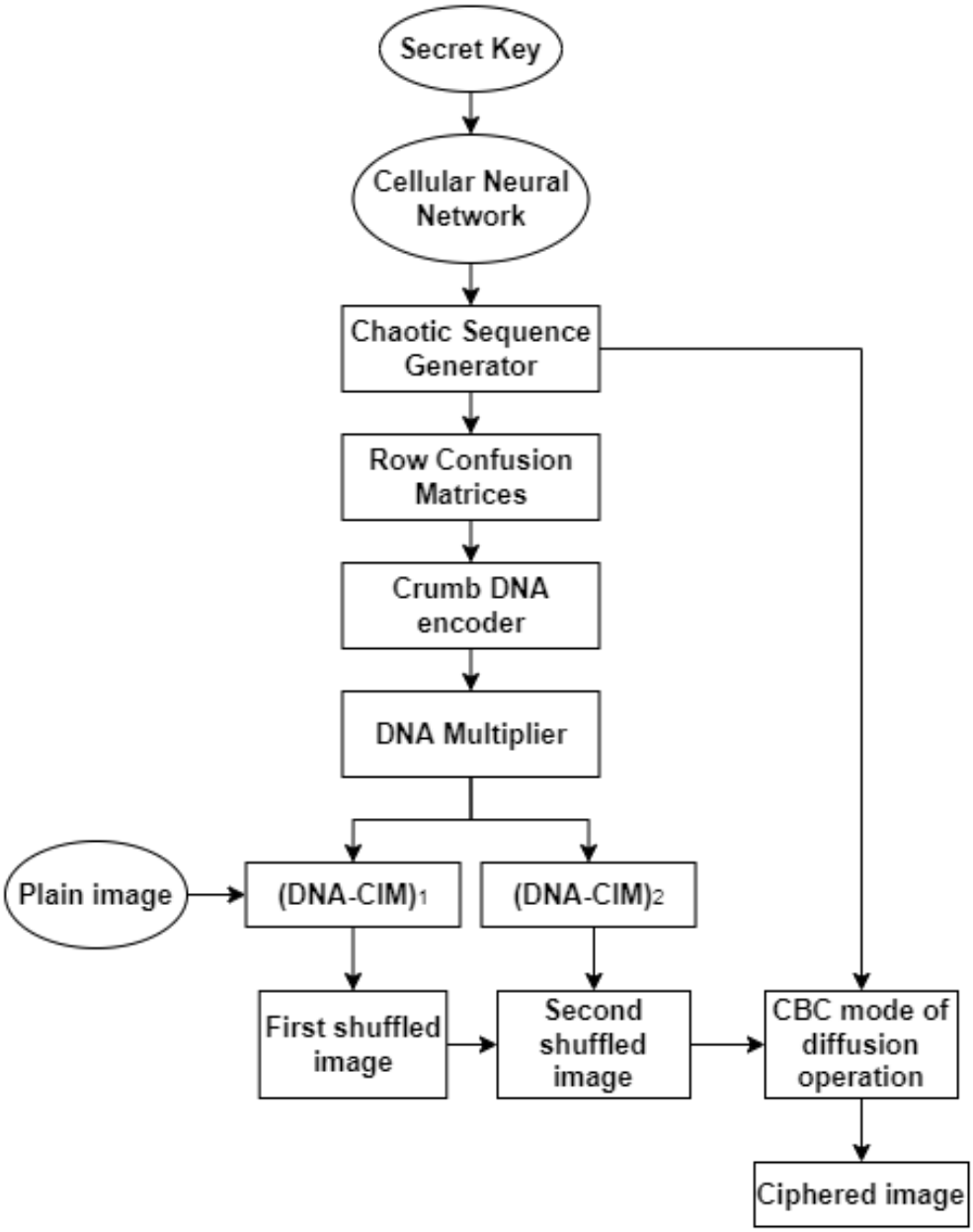

Figure 2.9: Sheela et al.[5] Proposed Encryption Flow Chart.

Wang et al.[6] introduced in 2017, a new encryption technique for colored images based on Deoxyribonucleic acid DNA sequence processes and cellular neural network (CNN). The presented cryptosystem in this work has characteristics of a large key space and complicated structure. First, The basic colored image is separated into three channel matrices (R, G, and B), then each matrix is converted into a DNA matrix according to the DNA encoding parameters. CNN's chaotic sequences are used to shuffle the elements' places in each of the three DNA sequence matrices. Thirdly, the cipher-image is retrieved by the DNA decoding rules via the DNA matrices after the three DNA matrices have been added up in accordance with certain rules and completed by complementary rules. Figure 2.10 illustrates the CNN-based color image encryption algorithm's DNA sequence operation.

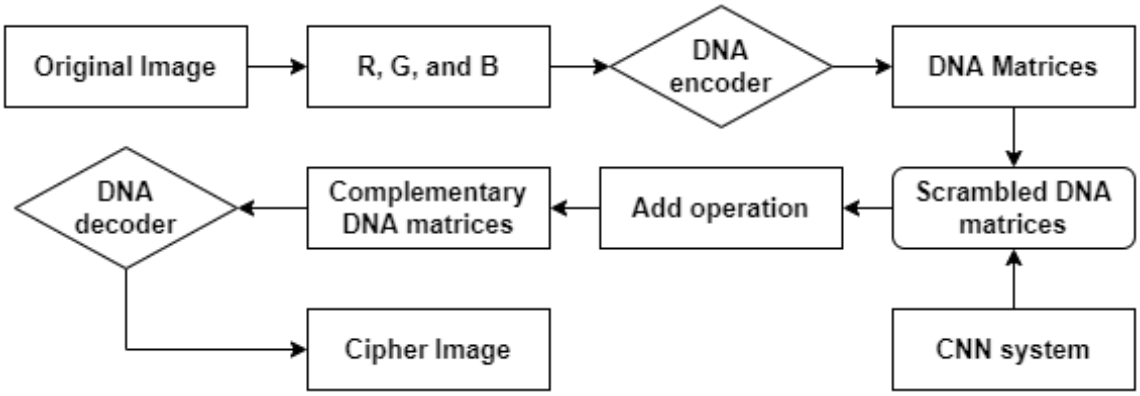

Figure 2.10: Wang et al.[6] Proposed Encryption Flow Chart.

We represent the description of some other previous works on cryptography based on the Cellular Neural Networks method in the following Table 2.2.

Table 2.2: Previous Works on Cryptography Based on CNNs

Ref	Author(s)	Year	Description of the method
[62]	Zhang et al.	2005	proposed a new color image encryption technique based on Deoxyribonucleic acid (DNA) sequence operations and CNN.
[63]	Qing et al.	2006	examined the chaotic behavior of the CNN model.
[64]	Peng et al.	2009	proposed an image encryption system based on the theoretical model in [63]
[65]	Xing et al.	2010	developed a multi-ary number communication system Based on a hyperchaotic system of 6th-ordercellular neural networks.
[66]	Li et al.	2013	introduced a color image cryptography system based on hyperchaotic CNN and chaotic control parameter, which used a compound "scrambling-diffusion" framework to encrypt the color image.
[56]	Kadir et al.	2014	used the CNN hyperchaotic system to design the diffusion sequence to encrypt images based on the "scrambling diffusion" method.
[67]	Wang et al.	2018	proposed the using of cellular automata (CA) to solve difficulties related to parallel computing.
[4]	Wang et al.	2020	introduce a new algorithm based on fractional order 5D cellular neural network and Fisher-Yates scrambling for image encryption. Using a fractional-order chaotic system and a cellular neural network.
[5]	Sheela et al.	2020	introduced a unique cryptosystem for medical image security, which is critical in teleradiology applications. The presented cryptosystem is based on the Fridrich architecture, which performs cryptographic operations using hyperchaotic CNN and DNA technology.
[30]	Zhang et al.	2021	proposed a new color image encryption technique, which combain Chen's chaotic system with a 6-dimensional cellular neural network (CNN). Symmetric cryptography involves this encryption approach.
[68]	Musanna et al.	2022	proposed a digital image encryption technique. The encryption is based on a permutation-substitution architecture basedo n chaos. The most important contribution is key generation, which is based on the Merkel–Damgard technique. Conway's game of life and the NARX network are used to carry out the diffusion process.

2.5 Conclusion

In this chapter, we discuss the basics of neural networks and shows some other type of neural networks. We discuss some current studies on the use and applications of neural networks in the domain of cryptography. We review some previous work on cryptography based on neural networks.

The designed neural network-based cryptosystem is a good idea for creating a very complex cryptosystem, where the crypto analyst or cracker not only needs to know the key and the topology of the neural network to break the system but also needs to be aware of the number of adaptive iterations and the final weights for the encryption and decryption systems. Higher

plain-text/cipher-text ratios are applied to the neural network-based cryptosystem in order to reduce error rates as much as possible.

In the next Chapter 3, we are going to implement a Six-Dimensional Cellular Neural Network (6D-CNN) to generate a pseudo-random number sequences. We will propose an efficient technique to generate the initial key conditions of the 6D-CNN by using two chaotic systems Lorenz 3D and Chen's 3D. The generation of the pseudo-random sequences will be explained in details of each step. Finally, to prove the efficient of the generated sequences, we going apply it on image encryption and run some security test to it.

CHAPTER 3

PROPOSITION OF AN EFFICIENT NEURAL GENERATOR FOR IMAGE ENCRYPTION

3.1 Introduction

Nowadays, random numbers play a critical part in modern-day cryptographic applications. A cryptographic system's security is based on randomly generated numbers, which are unpredictable and have acceptable statistical characteristics.

Random number generators (RNGs) are the significant constituents in many cryptographic systems, such as the generation of keys in both public and private key cryptography.

Chaos is the unpredictability of behaviors in a nonlinear deterministic system that is sensitive to the initial conditions. This description makes it clear that chaotic behavior can meet the complexity (nonlinearity) and unpredictable (sensitivity to initial conditions) requirements of RNG applications. Until now, a lot of studies have been focused on chaos-based RNG for this purpose.

The interconnected structure of the neurons in a neural network introduces complexity to the system. The mixing of neuron outputs from one layer into a single neuron of the next layer makes the system sensitive to the plaintext and produces a seemingly random output. This property of neural networks makes them suitable for generating cryptographic keys and the inputs at various points.

In this work, the features of both chaotic systems and neural networks are exploited to build an efficient cryptographic pseudo-random sequence based on a cellular neural network. The analyses of the proposed generator against the statistical, randomness, and encryption analyses demonstrate its excellent characteristics and improved performance.

In this chapter, we are going to introduce an efficient technique using cellular neural networks (CNN) to generate a pseudo-random number sequence with an additional technique to improve the initial key condition for 6D-CNN. Then after, we took the 6D-CNN and employ it in image encryption with explaining in detail each step. This chapter is organized as follow: in Section

3.2 we explain in depth how to generate pseudo-random sequence using 6D-CNN. In Section 3.3 we proposed a technique to improve the initial key value. Section 3.4 we did an analysis of the generated 6D-CNN sequences. Section 3.5 we used the 6D-CNN generated pseudo-random number sequence to encrypt/decrypt a colored or grayscale image. In Section 3.6 we do experimental results and security analysis. Finally in Section 3.7 we draw a conclusion.

3.2 Generate Pseudo-Random Sequence using 6D-CNN

To generate the pseudo-random number sequence there are two steps to go. the First is to evaluate the system of equations of the 6D-CNN. The second step is to apply the Runge-Kutta 4th-order the evaluated 6D-CNN systems of equations and generate the sequence.

3.2.1 Evaluating The 6D-CNN System Equations

Chua and Yang[32;33] introduced CNN for the first time in 1988. CNN's basic component unit is the cell. The state equation of each cell is represented by a nonlinear 1-order circuit consisting of linear resistance, a linear capacitance, and some voltage-controlled current sources. Which is generalized as the following equation.

$$C\frac{dx_{ij}}{dt} = -\frac{x_{ij}}{R_x} + \sum_{C_{kl}\in N_{ij}(r)} A(i,j;k,l)y_{kl} + \sum_{C_{kl}\in N_{ij}(r)} B(i,j;k,l)u_{kl} + I \tag{3.1}$$

where x_{ij} is the cell (i,j) state variable. The network's external output is represented by the character I. The matching input voltage of the cell (i,j) is referred by u_{kl}. C and R_x are system constants. A and B are the feedback parameter matrix and control parameter matrix respectively. y_{kl} is the equivalent output of the cell (i,j), and its output function. $f(x_{ij})$ is a linear function with the equation presented in 3.2:

$$y_{ij} = \frac{1}{2}(|x_{ij}+1| - |x_{ij}-1|) = f(x_{ij}) \tag{3.2}$$

More details about Cellular Neural Network are explained in Chapter 2 Section 2.2.2.6.

In Reference[69], the authors represent the chaotic phenomena in the 3rd-order CNN system. while in literature[70] they address the hyperchaotic phenomena in the 4th-order CNN system. In[4] proposes a simplified CNN model that separates the fractional order into 5th-order CNN. In[65], the authors expanded the number of cells to 6 to get the hyper-chaotic 6th-order CNN system. The equation for each cell of 6th-order CNN can be described as follow:

$$\frac{dx_i}{dt} = -x_j + a_j p_j + \sum_{k=1,k\neq j}^{6} a_{jk}p_k + \sum_{k=1}^{6} b_{jk}x_k + i_j \tag{3.3}$$

The parameters for each of the six cells are as follows:

$$a_j = 0 \quad (j = 1,2,3,5,6), \quad a_4 = 404$$
$$i_j = 0 \quad (j = 1,2,3,5,6)$$

A=0 and a_{jk} is a value in the matrix A in position (j,k) which mean

$$a_{jk} = 0 \quad (j,k = 1,2,...,6; j \neq k)$$

b_{jk} is a value in the matrix B, and B represents the following:

$$B = \begin{bmatrix} 1 & 0 & -1 & -1.2 & 0 & 0 \\ 0 & 3 & 1 & 0 & 0 & 0 \\ 11 & -12 & 1 & 0 & 0 & 0 \\ 92 & 0 & 0 & -94 & 1 & -1 \\ 0 & 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & -11 \end{bmatrix}$$

by following these parameters and equation 3.3 and 3.2 we can calculate the system equations of 6D-CNN as follow:

$$\begin{bmatrix} \frac{dx_1}{dt} \\ \frac{dx_2}{dt} \\ \frac{dx_3}{dt} \\ \frac{dx_4}{dt} \\ \frac{dx_5}{dt} \\ \frac{dx_6}{dt} \end{bmatrix} = - \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 404 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} f(x_1) \\ f(x_2) \\ f(x_3) \\ f(x_4) \\ f(x_5) \\ f(x_6) \end{bmatrix} + \begin{bmatrix} 1 & 0 & -1 & -1.2 & 0 & 0 \\ 0 & 3 & 1 & 0 & 0 & 0 \\ 11 & -12 & 1 & 0 & 0 & 0 \\ 92 & 0 & 0 & -94 & 1 & -1 \\ 0 & 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & -11 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix}$$

The result is the system equations 3.4

$$\begin{cases} \frac{dx_1}{dt} = -x_3 - 1.2x_4 \\ \frac{dx_2}{dt} = 2x_2 + x_3 \\ \frac{dx_3}{dt} = 11x_1 - 12x_2 \\ \frac{dx_4}{dt} = 92x_1 - 95x_4 + x_5 - x_6 + 202(|x_4 + 1| - |x_4 - 1|) \\ \frac{dx_5}{dt} = 5x_3 - x_5 \\ \frac{dx_6}{dt} = 5x_4 - 12x_6 \end{cases} \tag{3.4}$$

By evaluating the Lyapunov exponents of the system 3.4 we can analyze it dynamical behavior and the Lyapunov exponents of this system are[71]: $\lambda_1 = -0.3824, \lambda_2 = 0.1283, \lambda_3 = 0.1596, \lambda_4 = -0.3995, \lambda_5 = -1.3580, \lambda_6 = -0.5473$. As can be seen, the system 3.4 has two positive Lyapunov exponents, showing that it is a hyperchaotic system.

3.2.2 Apply The Runge-Kutta 4th-order and Generate the Pseudo-Random Sequences

3.2.2.1 Runge-Kutta 4th-Order

The Runge-Kutta method is a powerful approach for solving differential equations initial value issues. The Runge-Kutta technique is used to build high-order accurate numerical methods from functions without the necessity for high-order derivatives.

The Runge-Kutta 4th-order is the most well-known member of the Runge–Kutta family, and it computing formula for a single equation f generalized as follow:

$$\begin{cases} t_{n+1} = t_n + h \\ for \quad n = 0, 1, 2, ..., N \\ k_1 = hf(t_n, x_n) \\ k_2 = hf(t_n + \frac{h}{2}, x_n + \frac{k_1}{2}) \\ k_3 = hf(t_n + \frac{h}{2}, x_n + \frac{k_2}{2}) \\ k_4 = hf(t_n + h, x_n + k_3) \\ x_{n+1} = x_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \end{cases} \tag{3.5}$$

t represents the time and h is the step size parameter that can be chosen as a small number, and x is the initial value.

To apply the Runge-Kutta 4th-order in a 6th-order ordinary differential equation ($ODEs$). For our 6D-CNN equations system 3.4 f_i represent dx_i ($f_i = dx_i$). $x_1, x_2, x_3, x_4, x_5, x_6$, and h are the initial values and used as the CNNs system keys.

3.2.2.2 Generate The Sequences

In the generating of the pseudo-random number, we iterate the Algorithm **??** for n times and n represent the number of pixels in an image $W \times H$ (W is the width and H the height of the image). For each iteration, we obtain six output values. The values of these sequences are taken from these six values of each iteration, and every obtained output value is going to be the input for the next iteration.

3.3 Proposition for The Initial Key Values Improvement

In this section, we made a proposition to improve the initial key values of the 6D-CNN. The proposition is instead of us choosing the initial key values (x_i and h) of the 6D-CNN, we use two methods that each generate three values, every value is for one of the six 6D-CNN x_i. The two methods are going to be $Lorenz\ 3D$ system and $Chen's\ 3D$ system. To generate the 6D-CNN, we apply the Runge-Kutta 4th-order of 3th-order ordinary differential equation (ODEs) and we iterate them n times. The values in the iteration n are going the be the key values for the 6D-CNN. $Lorenz\ 3D$ goes by the following equation 3.6:

$$\begin{cases} \dot{x} = a(y - x) \\ \dot{y} = bx - y - xz \\ \dot{z} = xy - cz \end{cases} \tag{3.6}$$

where a, b, and c are $Loranz\ 3D$ system parameters. $Chen's\ 3D$ defined by the following system of equation:

$$\begin{cases} \dot{x} = a(y - x) \\ \dot{y} = (c - a)x - xz + cy \\ \dot{z} = xy - bz \end{cases} \tag{3.7}$$

a, b, and c are $Chen's\ 3D$ system parameters.

3.4 Sequences Analysis

The Sequences test analysis are going to be tested with Lorenz 3D parameters values $a = 10$,$b = 28$, $c = 8/3$, and initial values of $x(0) = 0.1$, $y(0) = 0.2$, $z(0) = 0.3$, and for the step size and number of iteration $h = 0.01$ and $t = 490$. For The Chen's system parameters values of $a' = 35$, $b' = 3$, $c' = 28$ and the initial values are: $x'(0) = 0.1$, $y'(0) = 0.2$, $z'(0) = 0.3$, and for the step size and number of iteration $h' = 0.01$ and $t' = 490$. The step size for the 6D-CNN is $h''(0) = 0.01$.

3.4.1 Key Sensitivity Analysis

In order to verify the reliability of the generated chaotic system by CNN. We need to analyze the testing sensitivity of the initial value.

To check the sensitivity, we add a small number to one of the eleven initial value (x, y, z, t, x', y', z', t', h, h', and h'') and verify the different between the first and the modified one sequences.

Figure 3.1 demonstrate the sensitivity between sequences generated with the preview initial values and sequences with the same initial values only with a small change in y' of $y'(0) = 0.2 + 10^{-14}$.

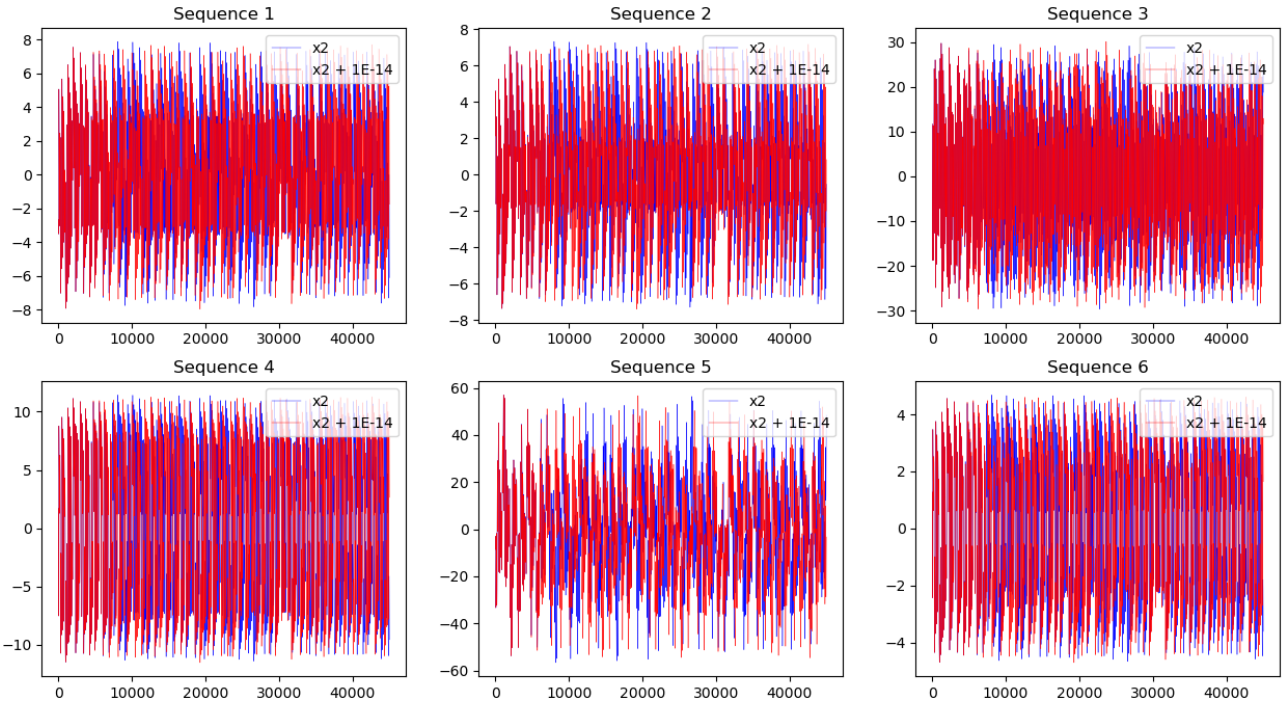

Figure 3.1: Sensitivity test to the initial values of 6D-CNN systems.

Figure 3.1 shows that despite knowing that the initial condition of y' is the only difference of 10^{-14}, the original state of this chaotic signal is overlapped, and a completely other evolution process occurs. The chaotic signal y' is being used as an example to examine the chaotic properties of the initial values. However, the chaotic signals x, y, z, x', and z' all follow a similar development process.

3.4.2 Chaotic Attractors Analysis

Figure 3.2 depicts the phase diagram of the partial 6-order CNN hyperchaotic attractor generated using the Runge–Kutta technique.

With the right parameters, the CNN system can generate a chaotic system, as seen in Figure 3.2.

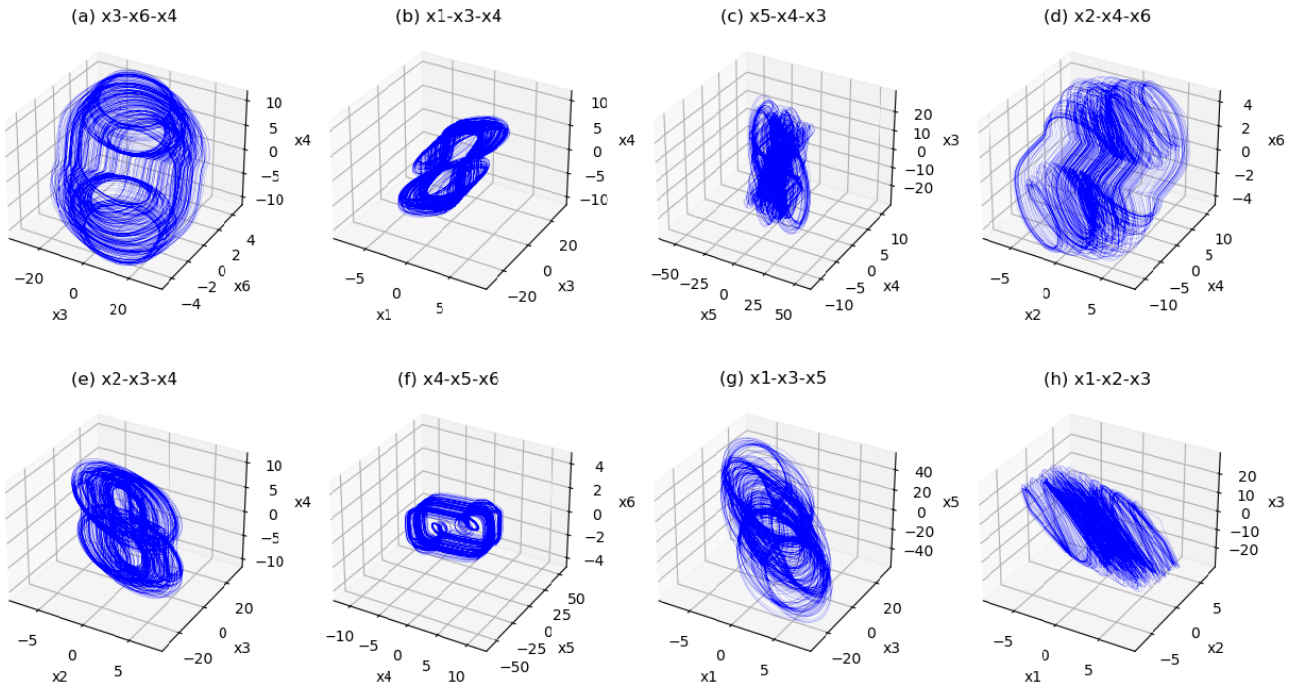

Figure 3.2: The projections of the hyperchaotic attractors of the 6D-CNN system of equations 3.4 in the three dimensional space.

3.5 Architecture of Image Encryption using 6D-CNN Sequences Generator

A general architecture and steps of image encryption with pseudo-random sequences generated using a 6D-CNN are shown in Figure 3.3.

There are three mutually independent stages in this type of image encryption. In the first step, we evaluate the mathematical equations of a 6th-order cellular neural network. The next step is to generate pseudo-random sequences by applying the 4th-order Runge-Kutta method to the equations system obtained from 6D-CNN in step one. Lastly, we cipher the plain image with the generated pseudo-random sequences using the XOR operator.

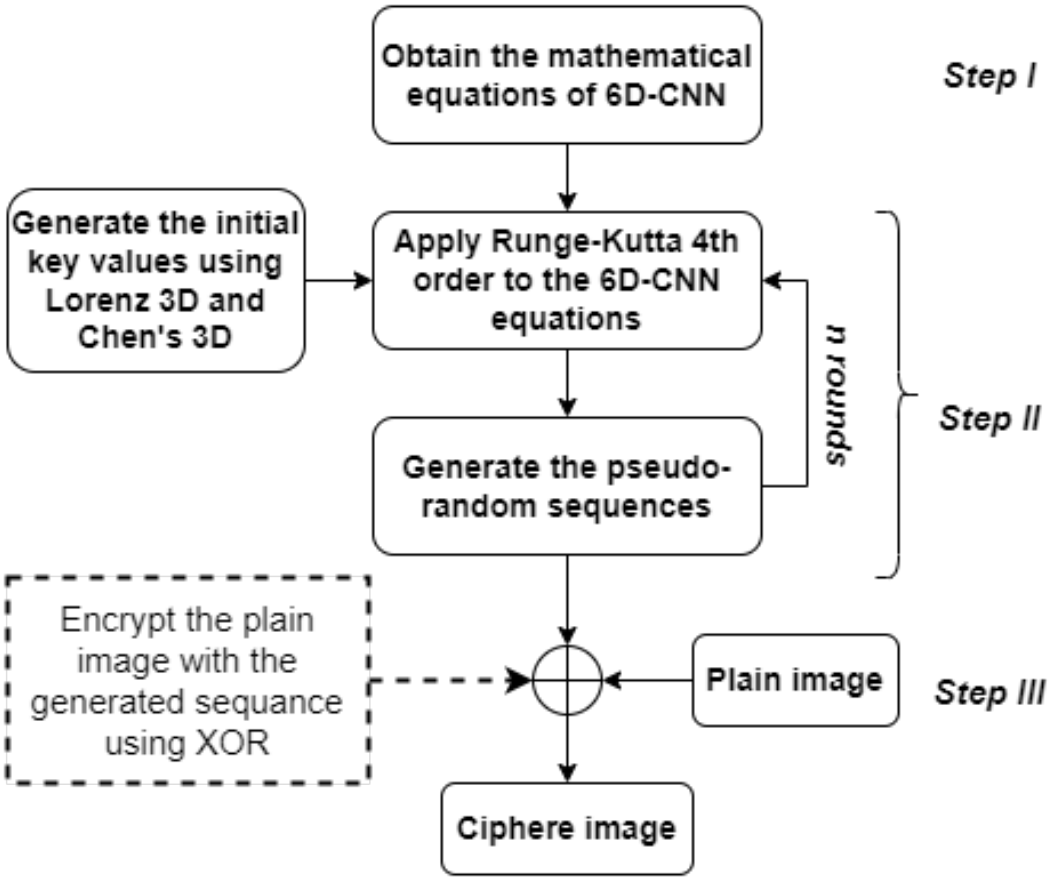

Figure 3.3: Architecture of Image Encryption using pseudo-random sequence generated with a 6D-CNN

3.5.1 Image Encryption/Decryption

Having outlined the 6D-CNN system and obtain the generated six pseudo-random sequences, in this part, we outline the design process of encryption and decryption of grayscale or colored image with a size of $W \times H$.

After iterating n times ($n = W \times H$), suppose the values of the generated sequences S_i are $(x_0, x_1, x_2, ..., x_n)$. To be able to implement these sequences for encryption and decryption, the values need to be between 0 and 255 which means $x_j \in [0, 255]$. We can manage it as the following equation:

$$\begin{cases} S_i = x_{i,j} \times 10^{14} \quad mod \quad 256 \\ for \quad i \in \{1, 2, ..., 6\} \\ and \quad j \in \{0, 1, ..., n\} \end{cases} \tag{3.8}$$

Next step, if the plain image we are going to encrypt is grayscale, we take only one of the six generated sequences and use it for both encryption and decryption. Otherwise, if the plain image is colored, we choose three sequences from the six generated sequences, for each of the three channels (R, G, and B channels) in the colored image, we give a sequence.

Suppose we are going to encrypt a colored image, we need to reshape the sequences to the image shape, which is as follows:

$$\begin{aligned} S_i &= Reshape(S_i, W, H) \\ where \quad & i \in \{1, 2, 3\} \end{aligned} \tag{3.9}$$

For the encryption part, we use the XOR ($\oplus$) operator. Each pixel will be encrypted to its corresponding coordinate in a sequence with a XOR operation. Suppose I is the plain image with it three channels I_R, I_G, and I_B. C is the ciphered image with three channels C_R, C_G, and C_B. The encryption is generalized as follow:

$$C = \begin{cases} C_R[i,j] = I_R[i,j] \oplus S_1[i,j] \\ C_G[i,j] = I_G[i,j] \oplus S_2[i,j] \\ C_B[i,j] = I_B[i,j] \oplus S_3[i,j] \end{cases} \tag{3.10}$$

The decryption part is going to be the same as the encryption, since the inverse of XOR is XOR itself. Which is as follows:

$$I = \begin{cases} I_R[i,j] = C_R[i,j] \oplus S_1[i,j] \\ I_G[i,j] = C_G[i,j] \oplus S_2[i,j] \\ I_B[i,j] = C_B[i,j] \oplus S_3[i,j] \end{cases} \tag{3.11}$$

Figure 3.4 shows some examples of plain images encrypted and decrypted from the explained steps in Figure 3.3.

Figure 3.4: plain images encrypted and decrypted with the explained steps. **(a)** The original plain images. **(b)** shows the effects of the encryption. **(c)** is the decryption of the ciphered images.

3.6 Security Analysis

The Security analysis are going to be tested with Lorenz 3D parameters values $a = 10$,$b = 28$, $c = 8/3$, and initial values of $x(0) = 0.1$, $y(0) = 0.2$, $z(0) = 0.3$, and for the step size and number

of iteration $h = 0.01$ and $t = 490$. For The Chen's system parameters values of $a' = 35$, $b' = 3$, $c' = 28$ and the initial values are: $x'(0) = 0.1$, $y'(0) = 0.2$, $z'(0) = 0.3$, and for the step size and number of iteration $h' = 0.01$ and $t' = 490$. The step size for the 6D-CNN is $h''(0) = 0.01$.

3.6.1 Key Space Analysis

The objective of the key space in security analysis is explained in Chapter 1 Section 1.5.1. To generate alternative pseudo-random sequences, we applied different initial values. The eleven parameters (x, y, z, t, x', y', z', t', h, h', and h'') in this technique can be any number of random digits. Its key space is determined by the computer's real accuracy. If a 64-bit machine is utilized. If calculate it. For Lorenz 3D we have (x, y, z, t, and h) which equals too 5×2^{64} or 2^{320}. for Chen's 3D the initial values are (x', y', z', t', and h') that equals too 5×2^{64} or 2^{320}. And for the 6D-CNN we have h'' is 2^{64}. Then the total is $2^{320} + 2^{320} + 2^{64} = 2^{704}$ or 11×2^{64}. The key space is extremely vast, allowing it to successfully withstand an exhaustive attack.

3.6.2 Histogram Analysis

The objective of histogram analysis is explained in Chapter 1 Section 1.5.2. Figure 3.5 shows the effect of encryption in colored images encrypted with the generated pseudo-random sequences. The histogram of the cipher images shows that all pixels are distributed equally, which makes the encryption can withstand histogram attack.

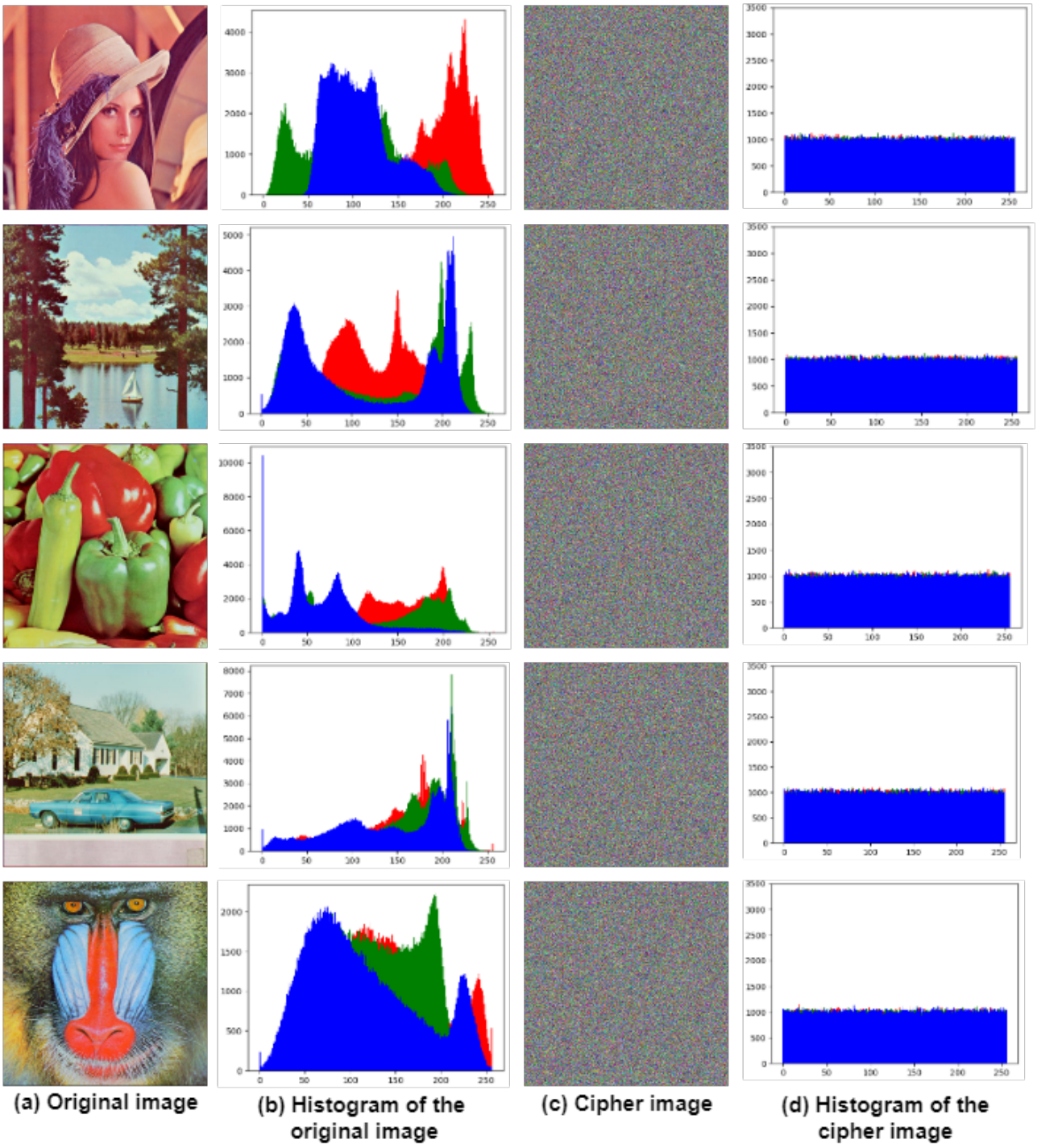

Figure 3.5: Histogram Analysis Tests to a Color Images

3.6.3 Correlation of Adjacent Pixels

The explanation about correlation coefficient and the way to evaluate it are in Chapter 1 Section 1.5.3. Table 3.1 shows the correlation of original plain images and encrypted images in the three components of a color image (R, G, and B) and in the three directions (horizontal, vertical, and diagonal). We see that the plain images have a high correlation, whereas the encrypted image has a correlation near to 0. That means that these generated pseudo-random sequences of a 6D-CNN have good cross-correlation characteristics. Table 3.2 shows a comparison between the scheme we implement of an encrypted Lena image with other schemes proposed in other references.

Table 3.1: Correlation of adjoining pixels of some original plain images and the encrypted cipher images.

Images	Horizontal			Vertical			Diagonal		
	R	G	B	R	G	B	R	G	B
Lena (512×512)	0.9628	0.9397	0.8844	0.9883	0.9819	0.9514	0.9827	0.9730	0.9408
Encrypted Lena	0.0011	0.0008	0.0016	0.0012	0.0005	0.0006	5e-06	0.0001	0.0015
Lake (512×512)	0.9208	0.9675	0.9606	0.9388	0.9729	0.9599	0.9601	0.9823	0.9834
Encrypted Lake	0.0009	0.0020	0.0036	0.0002	-0.0010	-0.0003	-0.0022	-0.0020	-0.0015
Peppers (512×512)	0.9757	0.9883	0.9731	0.9750	0.9905	0.9773	0.9771	0.9883	0.9734
Encrypted Peppers	0.0041	0.0006	0.0012	0.0009	-0.0010	-0.0014	0.0025	-0.0022	0.0004
House (512×512)	0.9573	0.9517	0.9710	0.9374	0.9415	0.9600	0.9529	0.9466	0.9670
Encrypted House	0.0017	0.0011	0.0017	0.0033	-0.0005	-0.0018	0.0026	-0.0011	-0.0027
Baboon (512×512)	0.8711	0.8513	0.8692	0.8208	0.7608	0.8259	0.8746	0.8179	0.9038
Encrypted Baboon	0.0021	0.0025	0.0023	-0.0006	-0.0002	-0.0005	0.0011	8e-06	-0.0008
San Diego (1024×1024)	0.7709	0.7814	0.7298	0.9140	0.9079	0.9037	0.9142	0.9037	0.8816
Encrypted San Diego	0.0004	0.0001	-0.0003	-7e-05	-0.0002	-0.0021	-0.00031	0.0003	-0.0014

Table 3.2: Comparison correlation coefficients between other referenced schemes of encrypted Lena image.

Scheme	Horizontal	Vertical	Diagonal
Our	0.0012	0.0010	0.0005
[30]	0.0076	–0.0125	0.0101
[72]	0.0076	0.0130	0.0138
[64]	0.0445	–0.0168	–0.0022
[66]	0.0195	0.0086	–0.0260
[28]	0.0113	0.0173	0.0099
[73]	0.1410	0.1967	0.1116
[74]	0.0172	0.0039	0.0277
[75]	–0.0909	0.2389	0.0126

3.6.4 Information Entropy

The details about information entropy and the evaluation equation are in Chapter 1 Section 1.5.4. Table 3.3 shows examples of plain images and cipher images tested with the information entropy in all three channels (R, G, and B). Table 3.4 shows a comparison between the scheme we implement of an encrypted Lena image and encrypted peppers image with other schemes proposed in other references.

Table 3.3: Plaintext and ciphertext images information entropy test.

Images	Plain Image			Cipher Image		
	R	G	B	R	G	B
Lena (512×512)	7.2531	7.5940	6.9684	7.9992	7.9992	7.9991
Lake (512×512)	7.3260	7.6358	7.3268	7.9992	7.9992	7.9992
Peppers (512×512)	7.3575	7.5929	7.1290	7.9991	7.9992	7.9992
House (512×512)	7.4645	7.2701	7.5084	7.9993	7.9993	7.9992
Baboon (512×512)	7.7324	7.4825	7.7570	7.9993	7.9992	7.9992
San Diego (1024×1024)	7.7575	7.3387	6.9561	7.9998	7.9998	7.9997

Table 3.4: Comparison of information entropy between other referenced schemes of encrypted Lena image and encrypted peppers image.

Scheme	**Encrypted Lena**			**Encrypted Peppers**		
	R	**G**	**B**	**R**	**G**	**B**
Our	7.9992	7.9992	7.9991	7.9991	7.9992	7.9992
[30]	7.9997	7.9937	7.9976	7.9932	7.9824	7.9969
[66]	7.9971	7.9977	7.9975	7.9971	7.9968	7.9974
[56]	7.9278	7.9744	7.9705	7.9391	7.9693	7.9805
[6]	7.9914	7.9914	7.9902	7.9910	7.9910	7.9911

3.6.5 Plaintext Sensitivity Analysis

The two plaintext sensitivity analysis techniques ($NPCR$ and $UACI$) are explained in details in Chapter 1 Section 1.5.5. We run a test of $NPCR$ and $UACI$ in some examples of colored images, and the result shows in Table 3.5. Table 3.6 shows a comparison of $NPCR$ and $UACI$ of Lena's image with the implemented method and other proposed scheme references.

Table 3.5: Tests of the number of pixel of change rate ($NPCR$) and unified average changing intensity ($UACI$) of color images.

Images	**NPCR(%)**			**UACI(%)**		
	R	**G**	**B**	**R**	**G**	**B**
Lena	99.7840	99.7200	99.7028	33.6114	33.6704	33.6007
Lake	99.7840	99.7200	99.7028	33.6407	33.6597	33.5380
Peppers	99.7840	99.7200	99.7028	33.6708	33.6927	33.5725
House	99.7840	99.7200	99.7028	33.5805	33.5957	33.5613
Baboon	99.7840	99.7200	99.7028	33.5981	33.6433	33.5494

Table 3.6: Comparison of $NPCR$ and $UACI$ of Lena image between other proposed scheme references.

Scheme	**NPCR(%)**	**UACI(%)**
Our	99.7356	33.6275
[76]	99.6168	33.4460
[4]	99.6302	33.4521
[59]	99.9133	33.3633
[6]	99.57	33.42

3.6.6 Robustness Analysis

The purpose of robustness analysis is explained in Chapter 1 Section 1.5.6. We demonstrate in Figure 3.6 the efficacy of our technique for a noisy cipher image with additional Gaussian noise with zero means and variance ranging from 0.01 to 1.

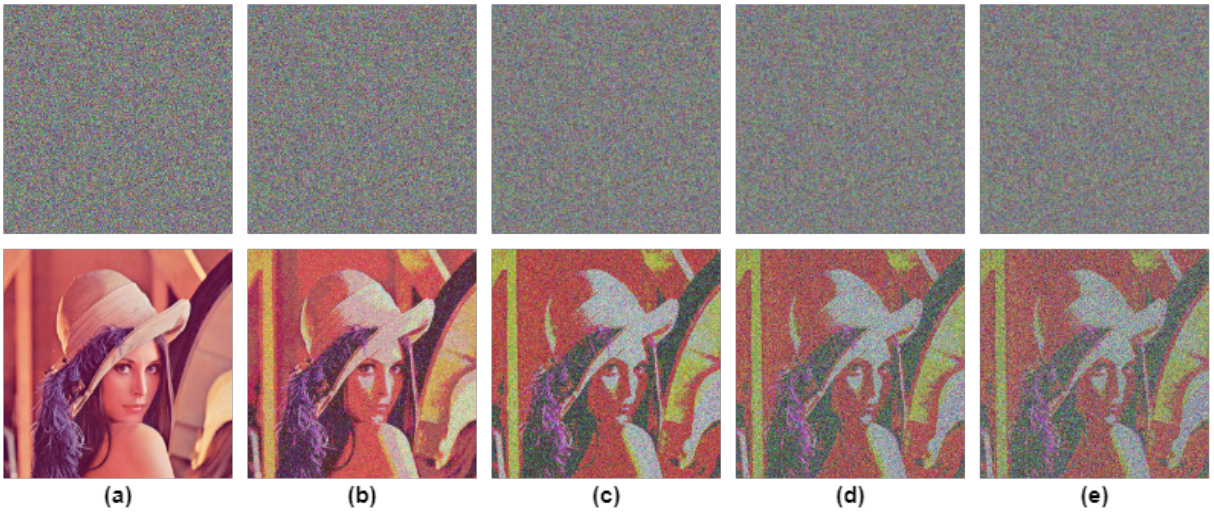

Figure 3.6: Decryption tests when the cipher image is corrupted with noise.

In Figure 3.6: **(a)** Gaussian blur with $\sigma^2 = 0.01$, **(b)** Gaussian blur with $\sigma^2 = 0.2$, **(c)** Gaussian blur with $\sigma^2 = 0.4$, **(d)** Gaussian blur with $\sigma^2 = 0.7$, and **(e)** Gaussian blur with $\sigma^2 = 1$.

3.7 Conclusions

In this chapter, we introduced an efficient neural networks generator to generate a pseudo-random number based on a hyper-chaotic system of six-dimensional cellular neural networks (6D-CNN). We proposed an efficient technique to improve the initial key conditions for the 6D-CNN. We run some tests to evaluate the quality of the generated sequences. The results show that the 6D-CNN has very good sensitivity to the change in the initial conditions, possesses good randomness, and has hyper-chaos characteristics.

We used the generated sequences in image cryptography to show to test the efficacy of the encryption. We used a simple operator, which is the XOR operator to encrypt and decrypt the given image. This algorithm is simple, efficient, and has low complexity. We did a comparison of the security testing results with other referenced algorithm of CNN's pseudo-random number generator and the implemented algorithm shows that our approach gives a good effect and is better than related work. It has good initial value sensitivity, plaintext sensitivity, correlation coefficient properties, and information entropy. All the experiments and security testing prove that encryption based on 6D-CNN works effectively for secure communication, and that it can satisfy network information security requirements while also increasing the use of the chaotic system in cryptography. Which gives nonlinear systems more options for producing random numbers that are independent, uniform, and complicated.

By proving that the generated pseudo-random number sequences of 6D-CNN have a great sensitivity and good randomness and showing that the 6d-CNN can be implemented to cryptography encryption. We proposed in the next Chapter 4, a novel image encryption based on cellular neural network and chaos.

CHAPTER 4

A NOVEL IMAGE ENCRYPTION ALGORITHM BASED ON CHAOS AND CELLULAR NEURAL NETWORKS

4.1 Introduction

By proving that cellular neural networks (CNN) have the chaos effect, and by having good results on our approach in chapter 3, we have made an Image cryptosystem that uses a combination of well-known chaos function "Arnold's Cat map" and Cellular Neural Networks. The cat map is used for the Substitution part, while CNN is going to be used to generate a pseudo-random sequence to be implemented in the Diffusion part. Making sure that our cryptosystem is Secure and solid enough we run it on various tests to demonstrate its efficiency.

Having some various approaches to use the CNN we stick to ours (6D-CNN), which gave us good results. And by applying the algorithms Lorenz 3D and Chen's 3D in chapter 3, this approach became very efficient.

In this chapter, we are going to walk through our approach to making a Cryptosystem. In Section 4.2 we discuss and explain the architecture of image encryption based on chaos and CNN that we have used. Following that, in Section4.3 we run our approach to various known Security tests to prove its quality and efficiency. Lastly, we draw our Conclusion in 4.4.

4.2 Architecture of Image Encryption Bases on Chaos and Cellular Neural Network

A general architecture of image encryption based on Chaos and Cellular Neural Network is shown in Figure 4.1.

The architecture of image encryption based on Chaos and Cellular neural Network consist of two steps "Substitution-Diffusion", First, is the substitution part, in which we take the plaintext image (PI) and scramble all the pixels coordinates using a chaotic maps substitutions

technique. the result is a scrambled image (SI). Secondly, we take the scrambled image we get in the substitution part and operate a chaotic diffusion technique to change the pixel values, which in our case, scrambled image XORed with pseudo-random number sequences generated using Six-Dimension Cellular Neural Network.

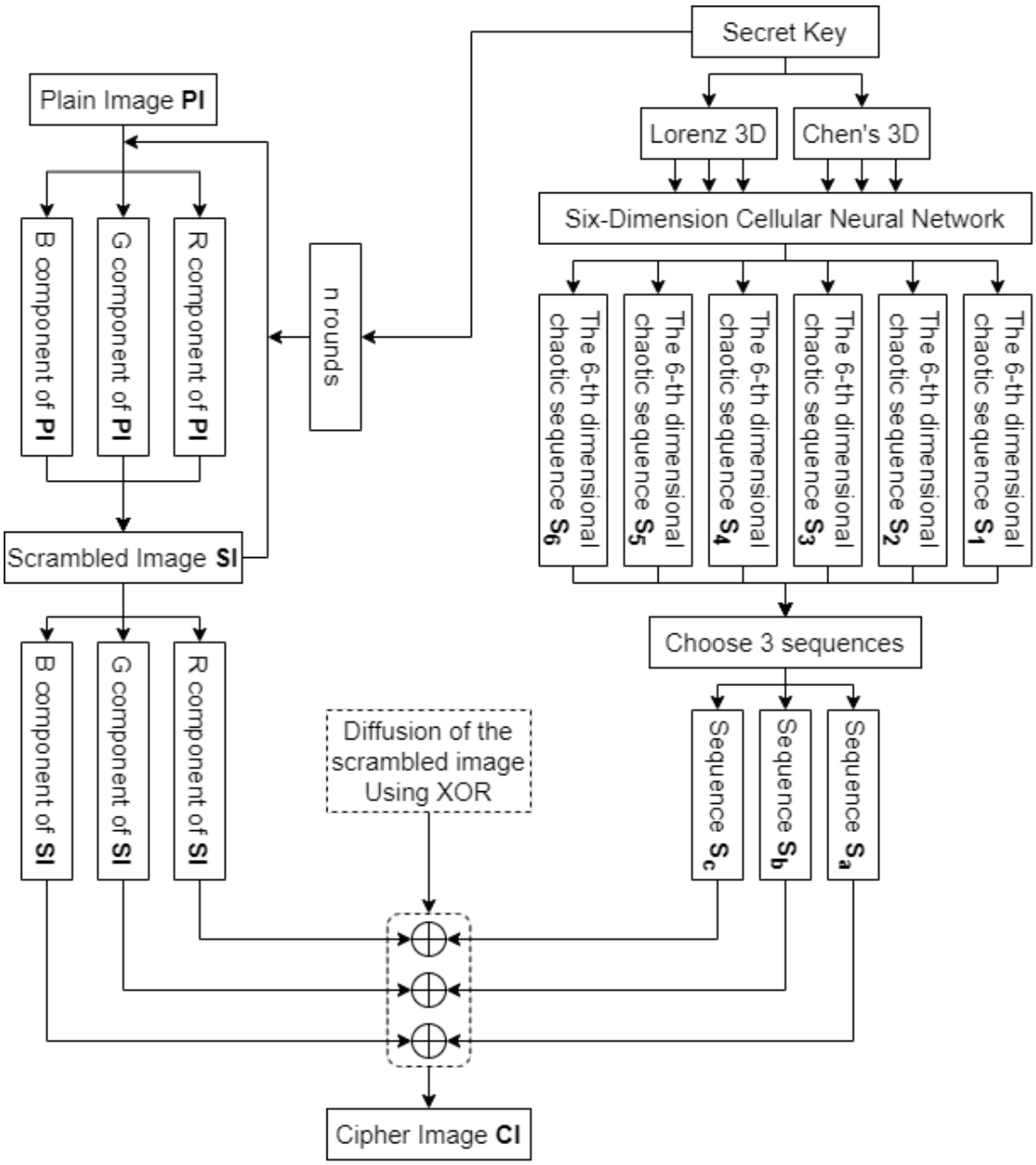

Figure 4.1: Architecture of Image Encryption Based Chaos and Cellular Neural Network

4.2.1 Substitution Plaintext Image Encryption

There are lots of techniques in the substitutions part, for example, we have the Baker map, and Standard map and the one we are going to use is the Arnold's Cat map. It is defined by the following equation 4.1:

$$(x_{t+1}, y_{t+1}) = (x_t + 2y_t mod N, x_t + y_t mod N) \tag{4.1}$$

x_t and y_t are the coordinates of a pixel image, x_t and y_t are the new coordinates, N represent the size of an image ($N \times N$). The key in Arnold's cat map in the number of iteration n. Figure

4.2 shows some plaintext image encrypted using Arnold's Cat map with different n iterations.

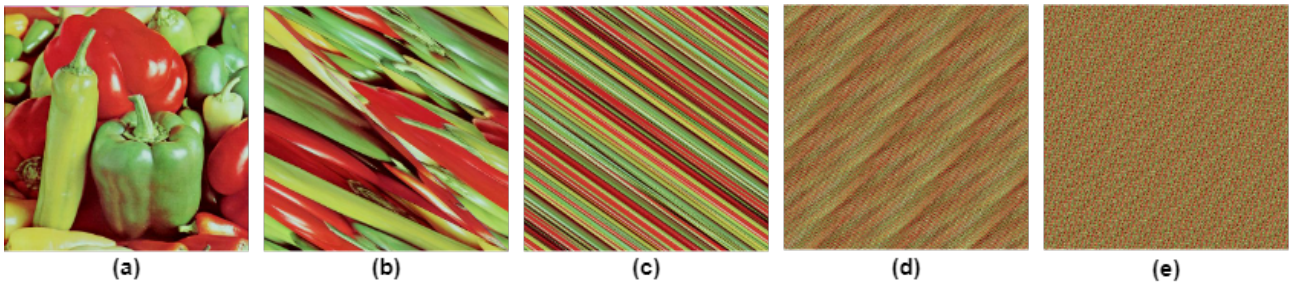

Figure 4.2: Plaintext images encrypted using Arnold's Cat map. **(a)** represent the plaintext images. **(b)** encryption based on Arnold's Cat map where $n = 1$. **(c)** encryption based on Arnold's Cat map where $n = 3$. **(d)** encryption based on Arnold's Cat map where $n = 10$. **(e)** encryption based on Arnold's Cat map where $n = 20$.

The decryption of Arnold's Cat map, the image return to it original state where number of iteration n is $n = N$. Which means, to encryption a ciphered image we need to iterate t times, where $t = |(n \mod N) - N|$. Figure 4.3 demonstrate an example image size of 512×512 ($N = 512$) encrypted with number of iteration $n = 300$, and decrypted with number of iteration $t = |(300 \mod N - 1) - 512 - 1|$ that equals too $t = 211$.

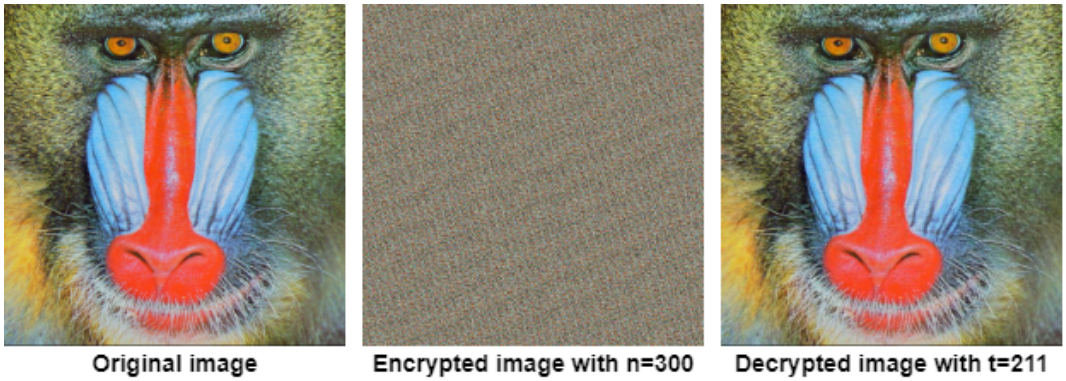

Figure 4.3: Plaintext image encryption and decryption.

4.2.2 Diffusion Image Encryption

In this part, we encrypt the scrambled image from the substitution part. The encryption goes by generating a pseudo-random sequences using a six-dimension cellular neural network as it explained in Chapter 3, then we XOR the scrambled image with the generated sequences. Figure 4.4 an experiment on some plaintext images encrypted and decrypted based on chaos and CNN. with secret key of Arnold's cat map $n = 300$. $x = 0.1$, $y = 0.2$, $z = 0.3$, step size $h = 0.01$, and number of iterations $t = 490$ for Lorenz 3D. And $x' = 0.1$, $y' = 0.2$, $z' = 0.3$, step size $h' = 0.01$ and number of iterations $t' = 490$ for Chen's 3D. For the 6D-CNN, step size of $h'' = 0.01$.

Figure 4.4: Plaintext Images Encrypted and Decrypted Based on Chaos and CNN. **(a)** is the plaintext images. **(b)** Substitution part encrypted using Arnold's cat map. **(c)** Diffusion part encrypted using Cellular Neural Network. **(d)** The decrypted Image.

4.3 Security Analysis

In this section we run all the tests with secret key of $n = 300$ iteration for Arnold's cat map. $x = 0.1$, $y = 0.2$, $z = 0.3$, step size $h = 0.01$, and number of iterations $t = 490$ for Lorenz 3D. x'$= 0.1$, y'$= 0.2$, z'$= 0.3$, step size h'$= 0.01$ and number of iterations t'$= 490$ for Chen's 3D. For the 6D-CNN, step size of $h'' = 0.01$.

4.3.1 Key Space

The key space in this proposition is the same as the one in Chapter 3 with the addition of n iteration of Arnold's cat map. which means, $2^{704} + 2^{64}$. Then the key space is 2^{768}.

4.3.2 Histogram Analysis

Figure 4.5 shows the histograms of the encrypted colored images in two parts (substitution and diffusion). In the substitution part the histogram stays the same as the original image, while in the diffusion part, all the pixels spread equally in the histogram, which makes the encryption can withstand histogram attack.

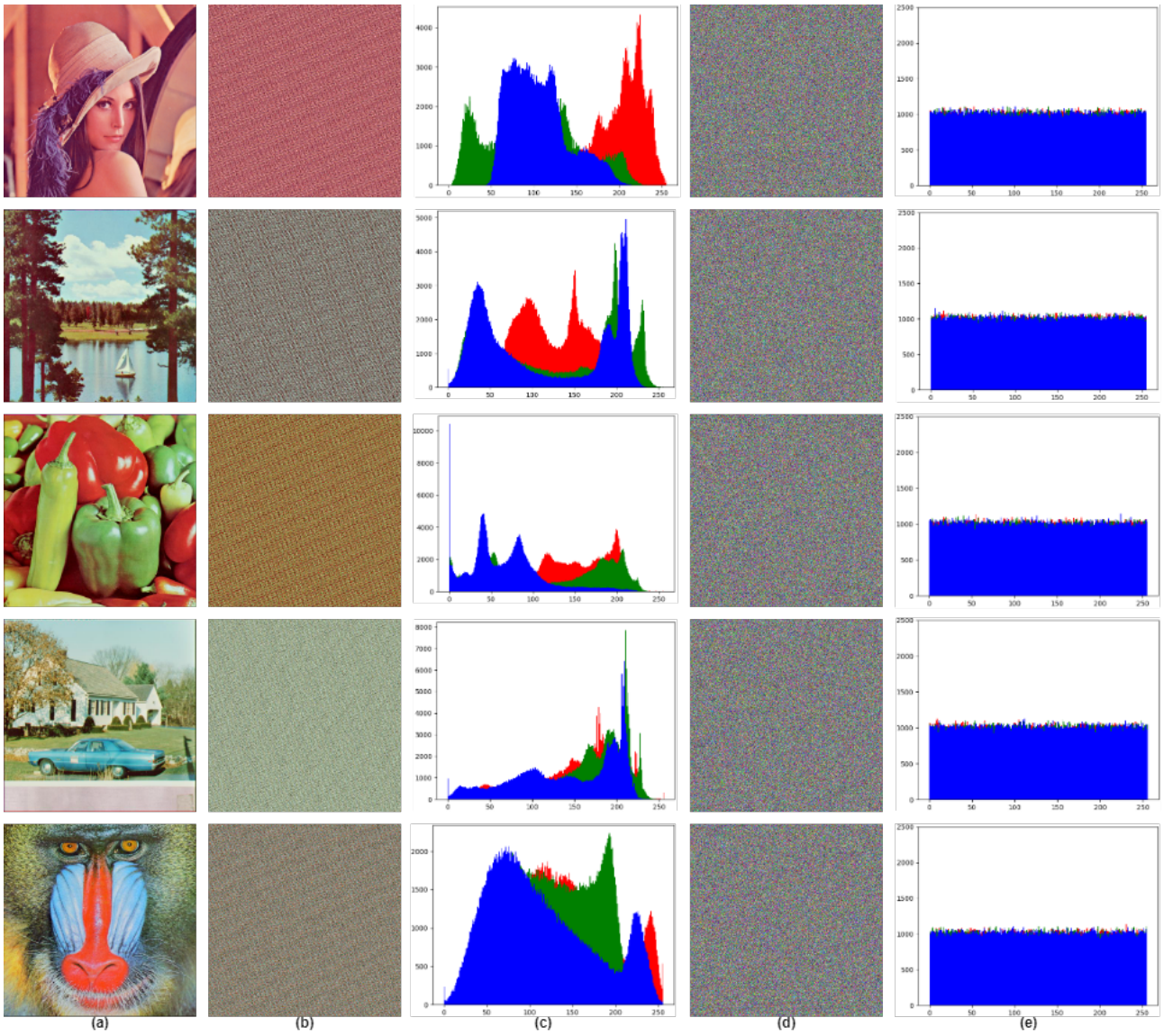

Figure 4.5: Histogram Analysis Tests to a Colored Plaintext Images

In Figure 4.5: **(a)** The original plaintext images. **(b)** Image encryption in the substitutions part. **(c)** Histograms of original plaintext images and the encrypted in substitution part. **(d)** Image encryption in the diffusion part. **(e)** Histograms images encrypted in the diffusion part.

4.3.3 Correlation of Adjacent Pixels

Table 4.1 shows the correlation of plaintext images and encrypted images using the three components of a colored image (R, G, and B) and in the three directions (horizontal, vertical, and diagonal). We see that the plaintext images have a high correlation, whereas the encrypted image has a correlation near to 0. That means that these image encryption based on chaos and CNN have good cross-correlation characteristics. Table 4.2 shows a comparison between the scheme we implement for an encrypted Lena image with other schemes proposed in other references.

Table 4.1: Correlation of adjoining pixels of plaintext images and the encrypted images using the explained method.

Images	**Horizontal**			**Vertical**			**Diagonal**		
	R	**G**	**B**	**R**	**G**	**B**	**R**	**G**	**B**
Lena (512×512)	0.9628	0.9397	0.8844	0.9883	0.9819	0.9514	0.9827	0.9730	0.9408
Encrypted Lena	-0.0021	-0.0006	-0.0030	0.0024	0.0003	-0.0010	-0.0024	0.0003	-0.0030
Lake (512×512)	0.9208	0.9675	0.9606	0.9388	0.9729	0.9599	0.9601	0.9823	0.9834
Encrypted Lake	-0.0015	-0.0003	0.0010	-0.0008	0.0011	0.0020	-0.0009	5e-05	-0.0023
Peppers (512×512)	0.9757	0.9883	0.9731	0.9750	0.9905	0.9773	0.9771	0.9883	0.9734
Encrypted Peppers	0.0016	-0.0001	0.0009	0.0009	0.0026	-0.0037	0.0004	0.0028	0.0009
House (512×512)	0.9573	0.9517	0.9710	0.9374	0.9415	0.9600	0.9529	0.9466	0.9670
Encrypted House	-0.0025	-0.0035	-0.0030	-0.0008	0.0019	0.0004	-0.0021	-0.0043	-0.0029
Baboon (512×512)	0.8711	0.8513	0.8692	0.8208	0.7608	0.8259	0.8746	0.8179	0.9038
Encrypted Baboon	0.0008	-0.0016	-0.0006	0.0015	0.0028	0.0021	0.0017	0.0008	-0.0003
San Diego (1024×1024)	0.7709	0.7814	0.7298	0.9140	0.9079	0.9037	0.9142	0.9037	0.8816
Encrypted San Diego	-0.0003	-6e-05	0.0002	-0.0004	0.0006	-0.0010	-0.0004	0.0012	-0.0005

Table 4.2: Comparison correlation coefficients between other referenced schemes of encrypted Lena image.

Scheme	**Horizontal**	**Vertical**	**Diagonal**
Our	-0.0024	0.0012	-0.0017
[30]	0.0076	–0.0125	0.0101
[72]	0.0076	0.0130	0.0138
[64]	0.0445	–0.0168	–0.0022
[66]	0.0195	0.0086	–0.0260
[28]	0.0113	0.0173	0.0099
[73]	0.1410	0.1967	0.1116
[74]	0.0172	0.0039	0.0277
[75]	–0.0909	0.2389	0.0126

4.3.4 Information Entropy

The details about information entropy and the evaluation equation are in Chapter 1 Section 1.5.4. Table 4.3 shows examples of plaintext images and ciphertext images encrypted using the explained method tested with the information entropy in all three channels (R, G, and B). Table 4.4 shows a comparison between the scheme we proposed of an encrypted Lena image and encrypted peppers image with other schemes proposed in other references.

The tasted results show that the information entropy is near 8 in the encrypted images, which means that these image encryption based on chaos and CNN have good information entropy encryption characteristics.

Table 4.3: Information entropy test of plaintext images and encrypted images using the explained encryption.

Images	**Plaintext Image**			**Ciphertext Image**		
	R	**G**	**B**	**R**	**G**	**B**
Lena (512×512)	7.2531	7.5940	6.9684	7.9992	7.9994	7.9992
Lake (512×512)	7.3260	7.6358	7.3268	7.9994	7.9993	7.9992
Peppers (512×512)	7.3575	7.5929	7.1290	7.9992	7.9992	7.9993
House (512×512)	7.4645	7.2701	7.5084	7.9993	7.9991	7.9993
Baboon (512×512)	7.7324	7.4825	7.7570	7.9993	7.9991	7.9990
San Diego (1024x1024)	7.7575	7.3387	6.9561	7.9998	7.9998	7.9997

Table 4.4: Comparison of information entropy between other referenced schemes of encrypted Lena image and encrypted peppers image.

Images	**Encrypted Lena**			**Encrypted Peppers**		
	R	**G**	**B**	**R**	**G**	**B**
Our	7.9992	7.9994	7.9992	7.9992	7.9992	7.9993
[30]	7.9997	7.9937	7.9976	7.9932	7.9824	7.9969
[66]	7.9971	7.9977	7.9975	7.9971	7.9968	7.9974
[56]	7.9278	7.9744	7.9705	7.9391	7.9693	7.9805
[6]	7.9914	7.9914	7.9902	7.9910	7.9910	7.9911

4.3.5 Plaintext Sensitivity Analysis

The two plaintext sensitivity analysis techniques ($NPCR$ and $UACI$) are explained in details in Chapter 1 Section 1.5.5. We run a test of $NPCR$ and $UACI$ in some examples of encrypted colored images, and the result shows in Table 4.5. Table 4.6 shows a comparison of $NPCR$ and $UACI$ of Lena's image with the proposed method and other proposed scheme references.

Table 4.5: Tests of the number of pixel of change rate ($NPCR$) and unified average changing intensity ($UACI$) of colored images.

Images	**NPCR(%)**			**UACI(%)**		
	R	**G**	**B**	**R**	**G**	**B**
Lena	99.6063	99.6131	99.6257	33.7233	33.7586	33.7586
Lake	99.6063	99.6131	99.6257	33.6766	33.7655	33.7648
Peppers	99.6063	99.6131	99.6257	33.7019	33.7676	33.6647
House	99.6063	99.6131	99.6257	33.6604	33.7970	33.8074
Baboon	99.6063	99.6131	99.6257	33.6518	33.8196	33.6655

Table 4.6: Comparison of $NPCR$ and $UACI$ of encrypted Lena image between other proposed scheme references.

Scheme	**NPCR(%)**	**UACI(%)**
Our	99.6150	33.7325
[76]	99.6168	33.4460
[4]	99.6302	33.4521
[59]	99.9133	33.3633
[6]	99.57	33.42

4.3.6 Robustness Analysis

The purpose of robustness analysis is explained in Chapter 1 Section 1.5.6. Figure 4.6 demonstrates the efficacy of our technique for a noisy cipher image with additional Gaussian noise with zero means and variance ranging from 0.01 to 1.

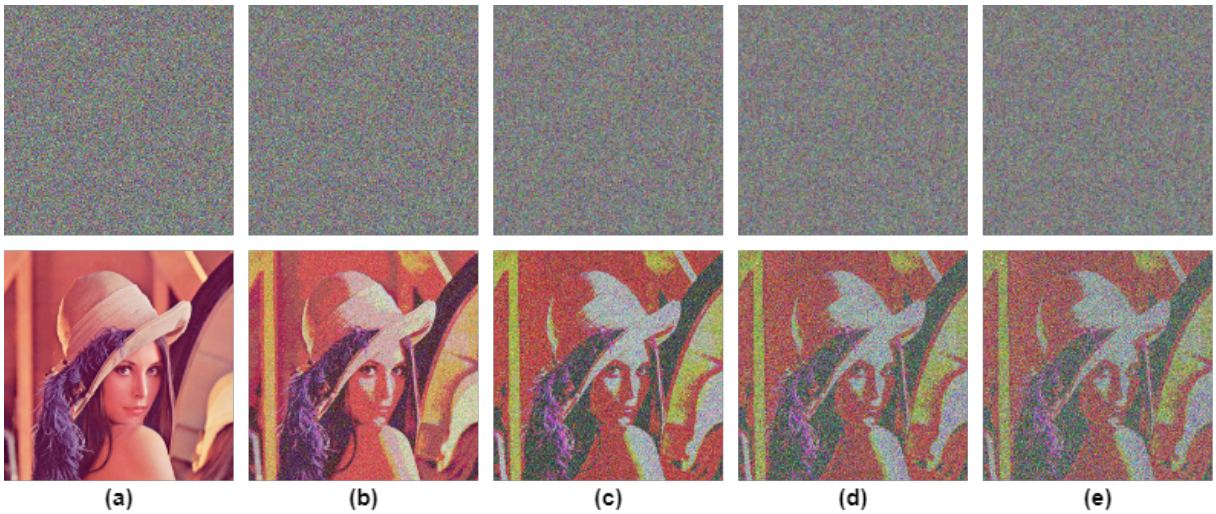

Figure 4.6: Decryption tests when the encrypted image corrupted with noise. **(a)** Gaussian blur with $\sigma^2 = 0.01$, **(b)** Gaussian blur with $\sigma^2 = 0.2$, **(c)** Gaussian blur with $\sigma^2 = 0.4$, **(d)** Gaussian blur with $\sigma^2 = 0.7$, and **(e)** Gaussian blur with $\sigma^2 = 1$.

4.4 Conclusion

In this chapter, we proposed novel image encryption based on Chaos and Cellular Neural Network to encrypt colored images. For the first part, which is the substitution part, we used Arnold's cat map to shuffle the pixels and get a scrambled image. Secondly, we generate pseudo-random sequences using 6D-CNN as explained in Chapter 3 and we XOR scrambled image with the generated sequence to get the final result, the ciphered image. This method is simple, efficient, and has good encryption results. we run and tested our algorithm for security analysis and we did a comparison between other image encryption schemes and our proposed approach method shows a greater effect compared to related works. It has great key space, good initial value sensitivity, a remarkable plaintext sensitivity, correlation coefficient characteristics, information entropy, and robustness. All the testing and encryption experiment shows that this approach has good security communication and that it can network information security requirements while also increasing the use of the chaotic system in cryptography. In conclusion, the suggested method can stand attacks that are specifically designed to damage its confidentiality and integrity.

GENERAL CONCLUSION

Synthesis

Cryptography is still the most efficient way to obtain better data security. For a long time, mankind has used this technique to ensure the confidentiality of messages, they have developed it in a simple but effective way. In this work, we did an application of neural networks to image cryptography.

In this work, we introduced a Six-Dimensional cellular neural network (6D-CNN) to generate a pseudo-random number sequences. We proposed in this work an improvement to generate the initial key conditions for the 6D-CNN. Then we took this general idea and proposed a novel image encryption based on Chaos "Substitution-Diffusion". In the substitution, we shuffle the image pixels coordinate to create a scrambled image. Then after, in the diffusion, we encrypt the scrambled image using the generated pseudo-random sequences by the 6D-CNN.

For the improvement in the initial key conditions for the 6D-CNN, we used 3D Lorenz chaos system and 3D Chen's chaos system methods. Which they have already proven that they have chaos characteristics. With this improvement, the key space became bigger and the initial key condition for the 6D-CNN are now generated in a chaotic way.

In the proposition of the generating of pseudo-random sequences using 6D-CNN, we run an analysis and evaluation to test the sensitivity of the initial key conditions and the randomness of the generated sequences. The test result shows that the 6D-CNN has good sensitivity to the change in the initial condition and possesses good randomness. The 6D-CNN proves that have chaos characteristics.

We took the generated 6D-CNN pseudo-random sequences, and we apply them to image encryption to prove they can be used in image cryptography. We run some test of security. All the results of the correlation coefficient, information entropy, histogram analysis, key space, plaintext sensitivity, and robustness analysis show very good results. We even did a comparison with other reference-related work's results. The comparison shows that it has good security performance and sometimes better.

After proving the chaos of 6D-CNN and indicating that offer good encryption security. We took it to another cryptosystem, and proposed novel image encryption based on chaos "Substitution-Diffusion". In the substitution, we used a method known as Arnold's Cat map, which plays the role of image pixels coordinates shuffling and obtaining the scrambled image to

get encrypted in the diffusion part by the generated 6D-CNN pseudo-random sequences. we run a security test on the proposed encryption scheme. The results indicate the key space became larger. which is 2^{768}. That means the security knows even better and harder for brute force attacks. We run the other test of the correlation coefficient, information entropy, histogram analysis, plaintext sensitivity, and robustness analysis. The results show that the proposed scheme offers good quality of security. we did a comparison with other related schemes and the results show that the proposed image encryption scheme possesses really good and efficient security and can withstand attacks that harm its confidentiality.

Perspectives

The work presented in this manuscript can be extended to accomplish other objectives. In future perspectives of this work, we plan to:

- Try to extend the dimension of the CNN to even more than six dimensions or make a change in the CNN function inputs which we think can give even better chaotic results.
- applying the CNN cryptosystem to other types of information like videos and sounds.
- using CNN in other types of application cryptosystems like visual cryptography.

BIBLIOGRAPHY

[1] Zhenlong Man, Jinqing Li, Xiaoqiang Di, Yaohui Sheng, and Zefei Liu. Double image encryption algorithm based on neural network and chaos. *Chaos, Solitons & Fractals*, 152:111318, 2021.

[2] Manel Dridi, Mohamed Ali Hajjaji, Belgacem Bouallegue, and Abdellatif Mtibaa. Cryptography of medical images based on a combination between chaotic and neural network. *IET Image Processing*, 10(11):830–839, 2016.

[3] Sakshi Patel, V Thanikaiselvan, Danilo Pelusi, B Nagaraj, Rajendran Arunkumar, and Rengarajan Amirtharajan. Colour image encryption based on customized neural network and dna encoding. *Neural Computing and Applications*, 33(21):14533–14550, 2021.

[4] Xingyuan Wang, Yining Su, Chao Luo, and Chunpeng Wang. A novel image encryption algorithm based on fractional order 5d cellular neural network and fisher-yates scrambling. *Plos one*, 15(7):e0236015, 2020.

[5] SJ Sheela, KV Suresh, Deepaknath Tandur, and A Sanjay. Cellular neural network-based medical image encryption. *SN Computer Science*, 1(6):1–11, 2020.

[6] Jingshuai Wang, Fei Long, and Weihua Ou. Cnn-based color image encryption algorithm using dna sequence operations. In *2017 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*, pages 730–736. IEEE, 2017.

[7] Michael Anshel and Kent D Boklan. Introduction to cryptography with coding theory, second edition. *The Mathematical Intelligencer*, 29(3):66–69, June 2007.

[8] Sean-Philip Oriyano and Shimonski. *Penetration Testing Essentials*. John Wiley & Sons, Incorporated, Indianapolis, UNITED STATES, 2016.

[9] Hans Delfs and Helmut Knebl. *Introduction*, pages 1–10. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[10] Gary C Kessler. An overview of cryptography. *Online: http://www. garykessler. net/library/crypto. html,[Last accesed: 01/10/09]*, 2020.

[11] Duncan Buell. *Fundamentals of Cryptography: Introducing Mathematical and Algorithmic Foundations*. Springer, Columbia, SC, USA, 2021.

[12] Matthew J.B. Robshaw Lars R. Knudsen. *The Block Cipher Companion*. Heidelberg ; New York : Springer-Verlag Berlin Heidelberg, Denmark and France, 2011.

[13] Christof Paar and Jan Pelzl. *Understanding Cryptography: A Textbook for Students and Practitioners*. Springer-Verlag Berlin Heidelberg, Germany, 2009.

[14] Douglas R. Stinson Philip A. Eisen. Threshold visual cryptography schemes with specified whiteness levels of reconstructed pixels, methods, and applications. *Designs, Codes and Cryptography*, 2002.

[15] Wei Qi Yan (auth.) Feng Liu. *Visual Cryptography for Image Processing and Security: Theory, Methods, and Applications*. Springer International Publishing, New Zealand, China, 2014.

[16] Q. Kester. A cryptographic image encryption technique based on the rgb pixel shuffling a cryptographic image encryption technique based on the rgb pixel shuffling. *International Journal of Advanced Research in Computer Engineering & Technology*, 2013.

[17] Chandana Gamage Mohan Harshana Perera Ranmuthugala. *Chaos Theory Based Cryptography in Digital Image Distribution*, page 33. International Conference on Advances in ICT for Emerging Regions (ICTer), Colombo, Sri Lanka, 2010.

[18] KW Wong. *Image Encryption Using Chaotic Maps*, pages 334–338. Springer, Berlin, Heidelberg, In: Kocarev, L., Galias, Z., Lian, S, 2009.

[19] Robert Peharz. *Neurocomputing*. Elsevier, Amsterdam, 2012.

[20] Shyi-Kae Yang, Chieh-Li Chen, and Her-Terng Yau. Control of chaos in lorenz system. *Chaos, Solitons & Fractals*, 13(4):767–780, 2002.

[21] GUANRONG CHEN and TETSUSHI UETA. Yet another chaotic attractor. *International Journal of Bifurcation and Chaos*, 1999.

[22] Chen. Guanrong Zhou. Tianshou, Tang. Yun. Complex dynamical behaviors of the chaotic chen's system. *International Journal of Bifurcation and Chaos*, 2003.

[23] Zhong Li Shujun Li, Gonzalo Alvarez and Wolfgang A. Halang. Analog chaos-based secure communications and cryptanalysis: A brief survey. *3rd International IEEE Scientific Conference on Physics and Control*, page 2, October 2007.

[24] S. Amini and A. L. Steele. *Digital Implementation*, pages 54–59. IEEE, Klagenfurt, Austria, 2009.

[25] Robert M Corless. What good are numerical simulations of chaotic dynamical systems? *Computers & Mathematics with Applications*, 28(10-12):107–121, 1994.

[26] Fathi E Abd el Samie, Hossam Eldin H Ahmed, Ibrahim F Elashry, Mai H Shahieen, Osama S Faragallah, El-Sayed M El-Rabaie, and Saleh A Alshebeili. *Image encryption: a communication perspective*. CRC Press, 2013.

[27] Wei Zhang, Kwok-wo Wong, Hai Yu, and Zhi-liang Zhu. An image encryption scheme using reverse 2-dimensional chaotic map and dependent diffusion. *Communications in Nonlinear Science and Numerical Simulation*, 18(8):2066–2080, 2013.

[28] Akram Belazi, Ahmed A Abd El-Latif, and Safya Belghith. A novel image encryption scheme based on substitution-permutation network and chaos. *Signal Processing*, 128:155–170, 2016.

[29] Xiuli Chai, Yiran Chen, and Lucie Broyde. A novel chaos-based image encryption algorithm using dna sequence operations. *Optics and Lasers in engineering*, 88:197–213, 2017.

[30] Renxiu Zhang, Longfei Yu, Donghua Jiang, Wei Ding, Jian Song, Kuncheng He, and Qun Ding. A novel plaintext-related color image encryption scheme based on cellular neural network and chen's chaotic system. *Symmetry*, 13(3):393, 2021.

[31] Manjit Kaur and Vijay Kumar. A comprehensive review on image encryption techniques. *Archives of Computational Methods in Engineering*, 27(1):15–43, 2020.

[32] Leon O Chua and Lin Yang. Cellular neural networks: Theory. *IEEE Transactions on circuits and systems*, 35(10):1257–1272, 1988.

[33] Leon O Chua and Lin Yang. Cellular neural networks: Applications. *IEEE Transactions on circuits and systems*, 35(10):1273–1290, 1988.

[34] T Schmidt, H Rahnama, and A Sadeghian. A review of applications of artificial neural networks in cryptosystems. In *2008 World Automation Congress*, pages 1–6. IEEE, 2008.

[35] Pranita P Hadke and Swati G Kale. Use of neural networks in cryptography: a review. In *2016 World Conference on Futuristic Trends in Research and Innovation for Social Welfare (Startup Conclave)*, pages 1–4. IEEE, 2016.

[36] Arooj Nissar and Ajaz Hussain Mir. Classification of steganalysis techniques: A study. *Digital Signal Processing*, 20(6):1758–1770, 2010.

[37] Liu Shaohui, Yao Hongxun, and Gao Wen. Neural network based steganalysis in still images. In *2003 International Conference on Multimedia and Expo. ICME'03. Proceedings (Cat. No. 03TH8698)*, volume 2, pages II–509. IEEE, 2003.

[38] Yun Q Shi, Guorong Xuan, Dekun Zou, Jianjiong Gao, Chengyun Yang, Zhenping Zhang, Peiqi Chai, Wen Chen, and Chunhua Chen. Image steganalysis based on moments of characteristic functions using wavelet decomposition, prediction-error image, and neural network. In *2005 IEEE International Conference on Multimedia and Expo*, pages 4–pp. IEEE, 2005.

[39] Guanshuo Xu, Han-Zhou Wu, and Yun-Qing Shi. Structural design of convolutional neural networks for steganalysis. *IEEE Signal Processing Letters*, 23(5):708–712, 2016.

[40] Moni Naor and Adi Shamir. Visual cryptography. In *Workshop on the Theory and Application of of Cryptographic Techniques*, pages 1–12. Springer, 1994.

[41] Rajat Bhatnagar and Manoj Kumar. Visual cryptography: A literature survey. In *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, pages 78–83. IEEE, 2018.

[42] Wolfgang Kinzel and Ido Kanter. Neural cryptography. In *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP'02.*, volume 3, pages 1351–1354. IEEE, 2002.

[43] Wolfgang Kinzel and Ido Kanter. Interacting neural networks and cryptography. In *Advances in solid state physics*, pages 383–391. Springer, 2002.

[44] DA Karras and V Zorkadis. Improving pseudorandom bit sequence generation and evaluation for secure internet communications using neural network techniques. In *Proceedings of the International Joint Conference on Neural Networks, 2003.*, volume 2, pages 1367–1372. IEEE, 2003.

[45] Saraswati D Joshi, VR Udupi, and DR Joshi. A novel neural network approach for digital image data encryption/decryption. In *2012 International Conference on Power, Signals, Controls and Computation*, pages 1–4. IEEE, 2012.

[46] Pavi Saraswat, Kanika Garg, Rajan Tripathi, and Ayush Agarwal. Encryption algorithm based on neural network. In *2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)*, pages 1–5. IEEE, 2019.

[47] Eva Volna, Martin Kotyrba, Vaclav Kocian, and Michal Janosek. Cryptography based on neural network. In *ECMS*, pages 386–391, 2012.

[48] Rafael Valencia-Ramos, Luis Zhinin-Vera, Gissela E Pilliza, and Oscar Chang. An asymetric-key cryptosystem based on artificial neural network. In *ICAART (3)*, pages 540–547, 2022.

[49] Tao Dong and Tingwen Huang. Neural cryptography based on complex-valued neural network. *IEEE transactions on neural networks and learning systems*, 31(11):4999–5004, 2019.

[50] Sooyong Jeong, Cheolhee Park, Dowon Hong, Changho Seo, and Namsu Jho. Neural cryptography based on generalized tree parity machine for real-life systems. *Security and Communication Networks*, 2021, 2021.

[51] Sayantica Pattanayak and Simone A Ludwig. Encryption based on neural cryptography. In *International Conference on Hybrid Intelligent Systems*, pages 321–330. Springer, 2017.

[52] Yuetong Zhu, Danilo Vasconcellos Vargas, and Kouichi Sakurai. Neural cryptography based on the topology evolving neural networks. In *2018 Sixth International Symposium on Computing and Networking Workshops (CANDARW)*, pages 472–478. IEEE, 2018.

[53] Nooshin Bigdeli, Yousef Farid, and Karim Afshar. A novel image encryption/decryption scheme based on chaotic neural networks. *Engineering Applications of Artificial Intelligence*, 25(4):753–765, 2012.

[54] Wenwu Yu and Jinde Cao. Cryptography based on delayed chaotic neural networks. *Physics Letters A*, 356(4-5):333–338, 2006.

[55] Gururaj Maddodi, Abir Awad, Dounia Awad, Mirna Awad, and Brian Lee. A new image encryption algorithm based on heterogeneous chaotic neural network generator and dna encoding. *Multimedia Tools and Applications*, 77(19):24701–24725, 2018.

[56] Abdurahman Kadir, Askar Hamdulla, and Wen-Qiang Guo. Color image encryption using skew tent map and hyper chaotic system of 6th-order cnn. *Optik*, 125(5):1671–1675, 2014.

[57] Florian Döhler, Florian Mormann, Bernd Weber, Christian E Elger, and Klaus Lehnertz. A cellular neural network based method for classification of magnetic resonance images: towards an automated detection of hippocampal sclerosis. *Journal of neuroscience methods*, 170(2):324–331, 2008.

[58] Vahid Jamshidi. A vlsi majority-logic device based on spin transfer torque mechanism for brain-inspired computing architecture. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 28(8):1858–1866, 2020.

[59] Kuru Ratnavelu, M Kalpana, P Balasubramaniam, K Wong, and Paramesran Raveendran. Image encryption method based on chaotic fuzzy cellular neural networks. *Signal Processing*, 140:87–96, 2017.

[60] Shuming Jiao, Changyuan Zhou, Yishi Shi, Wenbin Zou, and Xia Li. Review on optical image hiding and watermarking techniques. *Optics & Laser Technology*, 109:370–380, 2019.

[61] Qingmei Huang, Xueyan Zhao, and Guodong Li. Research on the application of video encryption technology based on 7 dimensional cnn hyper chaos. In *2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, pages 448–451. IEEE, 2018.

[62] Wei Zhang, Jun Peng, Huaqian Yang, and Pengcheng Wei. A digital image encryption scheme based on the hybrid of cellular neural network and logistic map. In *International Symposium on Neural Networks*, pages 860–867. Springer, 2005.

[63] Qingdu Li, Xiao-Song Yang, and Fangyan Yang. Hyperchaos in a simple cnn. *International Journal of Bifurcation and Chaos*, 16(08):2453–2457, 2006.

[64] Jun Peng, Du Zhang, and Xiaofeng Liao. A digital image encryption algorithm based on hyper-chaotic cellular neural network. *Fundamenta Informaticae*, 90(3):269–282, 2009.

[65] Wang Xingyuan, Xu Bing, and Zhang Huaguang. A multi-ary number communication system based on hyperchaotic system of 6th-order cellular neural network. *Communications in nonlinear science and numerical simulation*, 15(1):124–133, 2010.

[66] Jinqing Li, Fengming Bai, and Xiaoqiang Di. New color image encryption algorithm based on compound chaos mapping and hyperchaotic cellular neural network. *Journal of Electronic Imaging*, 22(1):013036, 2013.

[67] Yong Wang, Yi Zhao, Qing Zhou, and Zehui Lin. Image encryption using partitioned cellular automata. *Neurocomputing*, 275:1318–1332, 2018.

[68] Farhan Musanna, Deepak Dangwal, and Sanjeev Kumar. Novel image encryption algorithm using fractional chaos and cellular neural network. *Journal of Ambient Intelligence and Humanized Computing*, 13(4):2205–2226, 2022.

[69] Z He, Y Zhang, and H Lu. The dynamic character of cellular neural network with applications to secure communication. *JOURNAL-CHINA INSTITUTE OF COMMUNICATIONS*, 20:59–67, 1999.

[70] G-p Jiang and S-P Wang. Synchronization of hyperchaos of cellular neural network with applications to secure communication. *JOURNAL-CHINA INSTITUTE OF COMMUNICATIONS*, 21(9):79–85, 2000.

[71] Gangyi Hu, Jin Peng, and Weili Kou. A novel algorithm for generating pseudo-random number. *Int. J. Comput. Intell. Syst.*, 12(2):643–648, 2019.

[72] Chong Fu, Zhou-feng Chen, Wei Zhao, and Hui-yan Jiang. A new fast color image encryption scheme using chen chaotic system. In *2017 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, pages 121–126. IEEE, 2017.

[73] Ri-Gui Zhou, Ya-Juan Sun, and Ping Fan. Quantum image gray-code and bit-plane scrambling. *Quantum Information Processing*, 14(5):1717–1734, 2015.

[74] Xiaoling Huang and Guodong Ye. An image encryption algorithm based on irregular wave representation. *Multimedia Tools and Applications*, 77(2):2611–2628, 2018.

[75] Juan Deng, Shu Zhao, Yan Wang, Lei Wang, Hong Wang, and Hong Sha. Image compression-encryption scheme combining 2d compressive sensing with discrete fractional random transform. *Multimedia Tools and Applications*, 76(7):10097–10117, 2017.

[76] Farhan Musanna, Deepak Dangwal, and Sanjeev Kumar. Novel image encryption algorithm using fractional chaos and cellular neural network. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–22, 2021.