DEMOCRATIC REPUBLIC OF ALGERIA PEOPLE
MOHAMED BOUDIAF UNIVERSITY OF M'SILA
FACULTY OF MATHEMATICS AND INFORMATICS
Computer Science Department

N° Ordre : .....

A THESIS

Presented for obtaining the degree of:

DOCTOR IN SCIENCE

In : Computer Science
Specialty : Computer Science

By :
Sellami Benaissi

Theme

Contribution au développement d'une méthode hybride pour le cryptage d'image numérique (Contribution to the development of a hybrid method for digital image encryption)

Thesis defended on : October 10, 2024

Defended to the jury:

M. Tahar Mehenni	**MCA**	**University of M'sila**	**President**
M. Noureddine Chikouche	**Prof**	**University of M'sila**	**Supervisor**
M. Labib Sadek Terrissa	**Prof**	**University of Biskra**	**Examiner**
M. Farid Nouioua	**Prof**	**University of Bordj Bou Arreridj**	**Examiner**
M. Lyamine Guezouli	**MCA**	**Higher National School of Renewable Energy, Environment & Sustainable Development**	**Examiner**
M. Belkacem Brahimi	**MCA**	**University of M'sila**	**Examiner**

Academic Year : 2024-2025

Acknowledgments

First, I am grateful to Allah, all thanks and praise be to Allah for His favors, His guidance, and His preservation.

I extend my warmest thanks and heartfelt condolences to the memory of Mr LAMICHE Chaabane -may Allah have mercy on him-, who originally proposed the topic of this work.

Grateful acknowledgement is also due to Mr Noureddine Chikouche for his invaluable guidance, patience, and valuable insights throughout my research journey. His contributions were instrumental in shaping this thesis and bringing it to fruition.

I would also like to extend my thanks and heartfelt appreciation to the members of the jury for their time and their graciousness in accepting to discuss my work.

I am also grateful to the administration of our university and college, especially Mr. Bachir GAGUI, for their unwavering support and the excellent facilities they provided, which greatly facilitated my research.

I owe a debt of gratitude to my dear friends who stood by me throughout this endeavour, notably Dif yazid, Belkacemi Kamal, Chikh Aziz, Nadir Zine AlAbideen, and Yagoubi Rachad, for their unwavering encouragement and support.

Lastly, I wish to express my deepest appreciation to my wife, who patiently stood by me during the lengthy research process. Her unwavering support was a constant source of motivation.

May Allah Almighty reward all those who have supported me with the best in this world and the hereafter.

إهداء

إلى من كانوا مصدر إلهامي ودعمي، إلى من تركوا بصمة لا تمحى في قلبي، أهدي هذا العمل المتواضع بكل حب ووفاء:

- إلى روح والدي الغالي، رجل عظيم رحل ولكنه ترك الأثر، كان نعم القدوة ونعم الأب، يامن علمني معنى القوة والشجاعة، رحمك الله وغفر لك، اللقيا الجنة.

- وإلى أخي العزيزمحمد، رفيق دربي و معلمي، إلى من رحل عنا مبكراً ولكنه ترك في قلوبنا حباً لا ينضب، كنت نعم السند والمربي و المرشد، أسكنك الله فسيح جناته. لطالما انتظرت هذه اللحظة لكن شاء الله ان ترحل قبلها، لقيانا الجنة بإذنه

- وإلى والدتاي الحبيبتان، أهديكما هذا الجهد راجياً أن يكون على قدر حبكما وعطائكما.

- وإلى زوجتي الغالية، شريكة حياتي ورفيقة دربي، إلى من منحتني السعادة والسكينة، إلى من كانت سنداً لي في أوقات الشدة والفرح، إلى زوجتي الرائعة التي غمرتني بحبها ودعمها، إلى قلبها النابض بالحب أهدي هذا العمل تعبيراً عن امتناني وتقديري.

- وأخيراً، إلى فلذات أكبادي، أولادي الأعزاء، إلى من منحوني أجمل لقب في الحياة، إلى من هم مستقبلي وأملي، إلى أولادي آمريم ، قتيبة، سيرين، محمد الفاتحْ الذين أتمنى لهم النجاح والفلاح، إليهم أهدي هذا الإنجاز داعياً الله أن يحفظهم ويرعاهم.

- إلى اخوتي جميعا، عمي ، خالي، إلى كل طالب علم

إليكم جميعاً أهدي هذا العمل المتواضع، فأنتم مصدر إلهامي وقوة دفعِي نحو الأمام. رحم الله من رحل منكم ، وأطال في عمر من بقي، وجمعنا الله بكم في جنات النعيم

بن عيسي سلامي

Abstract

One of the most important data currently shared on the Internet is images of various sizes and types, gray and color, private and public, civilian and military, medical and scientific, commercial, and others. The biggest challenge and problem is the secure transmission of these images in this insecure environment. In this context, this dissertation proposes a new image encryption algorithm using a hybrid of three modified and improved chaotic one-dimensional (1D) maps to avoid the shortcomings of 1D maps and multidimensional (MD) maps. A key image is used to initialize the chaotic maps and is also used as a mask in the diffusion phase with the eXclusive OR (XOR) operator. The encryption process involves using a variable called "ExtraParam" that is extracted from the original image, which is very sensitive to bit changes during the initialization of the chaotic maps. The proposed encryption process consists of two main phases: confusion and diffusion processes. The proposed image encryption algorithm has successfully passed numerous tests and cryptanalysis. We also measure the proposed algorithm with various analysis experiments, such as histogram and entropy analysis, the number of pixel change rate (NPCR), the unified average changing intensity (UACI), the mean square error(MSE), the peak signal-to-noise ratio (PSNR) and Key Space. The experimental results and comparative analysis show that the proposed encryption algorithm has excellent performance, is strong enough to withstand various attacks, and provides excellent privacy to digital images.

Key Words: Image encryption, chaotic system, cryptography, confusion–diffusion, Key image.

Résumé

L'une des données les plus importantes actuellement partagées sur Internet est constituée d'images de tailles et de types variés : grises et couleur, privées et publiques, civiles et militaires, médicales et scientifiques, commerciales, etc. Le plus grand défi et problème est la transmission sécurisée de ces images dans cet environnement non sécurisé.

Dans ce contexte, cette thèse propose un nouvel algorithme de cryptage d'images utilisant un hybride de trois cartes chaotiques unidimensionnelles (1D) modifiées et améliorées pour éviter les lacunes des cartes 1D et multidimensionnelles (MD). Une image clé est utilisée pour initialiser les cartes chaotiques et sert également de masque dans la phase de diffusion avec l'opérateur OU exclusif (XOR). Le processus de cryptage implique l'utilisation d'une variable appelée « ExtraParam » qui est extraite de l'image d'origine, qui est très sensible aux changements de bits lors de l'initialisation des cartes chaotiques. Le processus de cryptage proposé se compose de deux phases principales : la phase de confusion et la phase de diffusion.

L'algorithme de cryptage d'images proposé a subi avec succès de nombreux tests et analyses cryptographiques. Nous évaluons également l'algorithme proposé à l'aide d'expériences d'analyse diverses telles que l'analyse d'histogramme et d'entropie, le taux de changement de pixel (NPCR), l'intensité moyenne de changement unifié (UACI), l'erreur quadratique moyenne (MSE), le rapport signal sur bruit de crête (PSNR) et l'espace de clés. Les résultats expérimentaux et l'analyse comparative montrent que l'algorithme de cryptage proposé a d'excellentes performances, est suffisamment solide pour résister à diverses attaques et offre une excellente confidentialité aux images numériques.

Mots Clés : Encryption d'image, système chaotique, cryptographie, confusion - diffusion, image clé.

الملخص

تعد الصور بأحجامها وأنواعها المختلفة، الرمادية و الملونة، الخاصة و العامة، المدنية و العسكرية، الطبية و العلمية، التجارية وغيرها، من أهم البيانات التي يتم مشاركتها حاليا على الإنترنت. ويمثل النقل الآمن لهذه الصور في هذه البيئة غير الآمنة التحدي والمشكلة الأكبر. في هذا السياق، نقترح في هذه الاطروحة خوارزمية تشفير صور جديدة باستخدام مزيج من ثلاث خرائط فوضوية أحادية البعد (1D) معدلة ومحسنة لتجنب قصور الخرائط أحادية البعد والخرائط متعددة الأبعاد (MD) . يتم استخدام صورة مفتاح لتهيئة الخرائط الفوضوية وكذلك كقناع في مرحلة الانتشار باستخدام عامل (XOR) (أو عملية الخلاف الحصري). تتضمن عملية التشفير استخدام متغير يسمى ("ExtraParam") يتم استخراجه من الصورة الأصلية، وهو حساس للغاية لتغييرات البت أثناء تهيئة الخرائط الفوضوية. تتكون عملية التشفير المقترحة من مرحلتين رئيسيتين: مرحلة التشويش ومرحلة الانتشار. نجحت خوارزمية تشفير الصور المقترحة في اجتياز العديد من الاختبارات وتحليل التشفير. نقوم أيضًا بقياس الخوارزمية المقترحة باستخدام تجارب تحليل مختلفة مثل تحليل الهستوغرام والانتروبي، ومعدل تغيير البكسل (NPCR) ، ومتوسط تغيير شدة الإشارة الموحد (UACI) ، ومتوسط مربع الخطأ (MSE) ، ونسبة ذروة الإشارة إلى الضوضاء (PSNR) وحيز المفتاح. تظهر النتائج التجريبية والتحليل المقارن أن خوارزمية التشفير المقترحة تتمتع بأداء ممتاز، وقوية بما يكفي لتحمل الهجمات المختلفة، وتوفر خصوصية ممتازة للصور الرقمية.

كلمات مفتاحية :
تشفير الصور، نظام الفوضى، التشفير، التشويش والانتشار، صورة مفتاح.

Contents

List of Figures

List of Tables

Abreviations list

1D	One-dimensional
2D-LASM	2D Logistic-adjusted-Sine map
AES	Advanced Encryption Standard
BMP	Bitmap
bpc	Bits per channel
bpp	Bits per pixel
CA	Cellular automata
CMYK	Cyan, Magenta, Yellow, Black
DCT	Discrete Cosine Transform
DES	Data Encryption Standard
DFT	Discrete Fourier Transform
DNA	Deoxyribonucleic acid
DOC	Word Document
DPI	Dots per inch
ECC	Elliptic curve based encryption
EEG	Electroencephalogram
FFT	Fast Fourier Transform
FPGA	Field Programmable Gate Array
GAN	Genetic Adversarial Networks
GIF	Graphics Interchange Format
GPU	Graphics Processing Unit
HHT	Hilbert-Huang Transform
HSL	Hue, Saturation, Lightness
HSV	Hue, Saturation, Value
IFS	Iterated Function System
ILM	Improved Logistic map
ImproLS	improved Lorenz system
IoT	Internet of Things
ISM	Improved Sine Map
JPEG	Joint Photographic Experts Group
KLT	Karhunen-Loève Transform
LE	Lyapunov Exponent

LCM	Logarithmic Chaotic Map
LOMAS	Logistic-May System
LL-XZA	Liu Lorenz-XOR Zigzag Arnold
LZW	Lempel-Ziv-Welch
MD	Multidimensional
MD5	Message Digest Algorithm
MRI	Magnetic Resonance Imaging
MSE	Mean square error
NIST	National Institute of Standards and Technology
NPCR	Number of pixel change rate
OCR	Optical Character Recognition
PDF	Portable Document Format
PIPS	Pixels per inch
PSNR	Peak signal-to-noise ratio
PRG	Pseudo random generators
PRNS	Pseudo random number sequence
PNG	Portable Network Graphics
QKD	Quantum Key Distribution
QIR	Quantum Image Representation
RGB	Red, Green, Blue
RLE	Run-Length Encoding
RSA	Rivest Shamir Adleman
SCCT	Sine Cosine Chebyshev Transform
SHA-256	Secure Hash Algorithm
SNM	Single Neuron Model
STFT	Short-Time Fourier Transform
TIFF	Tagged Image File Format
TXT	Text File
UACI	Unified average changing intensity
WT	Wavelet Transform
WHT	Walsh-Hadamard Transform
XOR	eXclusive OR

General Introduction

Introduction

Digital devices have sparked a comprehensive change in all facets of life, encompassing social, scientific, cultural, and financial domains. It has become an essential element of everyone's or society's lives, the digital camera, in particular, has been instrumental in this shift, It has grown ubiquitous, appearing on phones, in people's hands, in homes, in hospitals, on military satellites, and everywhere else. Consequently, the image has acquired major importance for both individuals and organizations, which has consequently elevated the value of the information it contains [59]. We find that it contains personal information, medical information, bank accounts, and even military information and other confidential and very sensitive information.

In addition to this, there is the extensive and terrible proliferation of communication networks and their utilization in touching all aspects and areas of the globe. The majority, if not all, transactions are now conducted through networks, particularly the Internet. This poses an important challenge in ensuring the security of individuals' information and safeguarding their safety and privacy. The challenge is made more difficult by the fact that the majority of the current data flow consists of digital images, which have become the primary data used in numerous applications. These transactions and applications encompass different fields including medicine, internet banking, online commerce, communications, and military-themed images. And numerous other applications. Here is where the role of information protection methods is crucial in ensuring the security of data during its transfer or exchange. Encryption is the most significant technique in this regard, as it is highly effective in preserving data confidentiality and safeguarding it against unauthorized alterations or malicious targeting.

Cryptography has consistently demonstrated its efficacy in safeguarding data and preserving its confidentiality, especially text data [57]. This aspect has been extensively addressed by researchers in numerous studies and projects. Several robust and exceptionally effective methods and algorithms exist for encrypting text, including Advanced Encryption Standard (AES), Rivest Shamir Adleman (RSA) and the family of elliptic curve based encryption (ECC) and others. Each of them

has demonstrated their proficiency in encrypting and safeguarding files and textual data.

However, these techniques are inadequate in safeguarding digital image and their efficacy is compromised when applied to it. Moreover, they are susceptible to certain attacks and and exhibit slowness in implementation, particularly when shared in real-time. The reason for this is the characteristics that images have over text. Several features distinguish digital images from textual data. The most crucial are:

1. The very strong correlation between the image pixels,
2. The large volume of information,
3. The high frequency and redundancy of pixels, and others.

These characteristics make traditional methods of encrypting text data unsuitable for securing images. Traditional encryption methods require more time, computational resources, and high performance to encrypt images.

This necessitates the use of alternative methods and techniques that consider image properties and characteristics in order to build image encryption algorithms that are strong and resistant to all types of attacks, and can be implemented quickly in real time. Hence, researchers have developed a variety of image encryption algorithms. Various encryption algorithms have been proposed, each relying on distinct principles and based on different concepts, including : DNA computing [51, 6, 106], neural network [53], cellular automata [46], compressive sensing [12], optical transformation [41], quantum theory [60], chaotic maps [64, 40, 85, 41], the visually meaningful encryption [35, 98], asymmetric image encryption [36, 100],and multi-image encryption [98].

In recent years, chaotic maps have garnered significant interest from researchers because to their several advantages, including [102]: pseudorandomness, unpredictability, ergodicity, nonperiodicity, and high sensitivity to initial conditions and control parameters.

Furthermore, it is segmented into two distinct categories based on its dimensions: One-dimensional (1D) and multi-dimensional (MD). 1D maps are distinguished by their high execution speed and easy implementation, whereas MD maps are distinguished by their extensive key space and greater complexity. Nevertheless, chaotic systems are constrained by their chaotic performance, which includes drawbacks such as limited key spaces, inadequate complexity, and low security for 1D maps. Similarly, MD maps suffer from high computational costs and challenging implementation, resulting in deficiencies in encryption techniques. In order to overcome

these limitations, numerous studies choose for enhancing current maps or employing hybrid approaches that utilize multiple maps.

In addition to the study of encryption algorithms, cryptanalysis of existing chaotic image encryption algorithms is also constantly performed to demonstrate the level of security and to reveal vulnerabilities in the encryption algorithms. In particular, several chaotic-based encryption algorithms have been broken because they could not withstand a specific plaintext attack or a known plaintext attack [34, 94]. For this reason, the development of new technologies and algorithms to protect digital images from these innovative attacks is a constant necessity.

In this context, and to take advantage of the characteristics of chaotic maps, their extreme sensitivity to initial conditions and parameters, and their great potential in generating pseudorandom number strings. In this dissertation, we present our proposal of a method to encode grayscale images based on the hybridization of one-dimensional chaotic maps. The proposed work is based on the use of three improved and modified one-dimensional maps :

1. Improved Logistic Map ILM,
2. Logistic-May System LOMAS,and
3. Improved Sine Map ISM,

Which have excellent chaotic behavior and retain the advantages of one-dimensional maps such as simple structure, ease of implementation, and high execution speed. As proven by experiments and graphs.

The 1D maps (such as the logistic, May, and sine maps) exhibit certain limitations, including simple behavior and limited chaotic ranges, as clearly demonstrated in the bifurcation diagram and the Lyapunov exponent. These shortcomings can adversely affect certain chaos-based applications, particularly in encryption techniques [102]. Furthermore, these limitations serve as vulnerabilities that can be exploited by cryptanalysis to crack the cryptosystem, mostly because of the weaknesses in generating the encryption keys [102, 75]. Regarding multidimensional maps, they possess drawbacks such as a significant computational cost and difficulty of implementation. To address the previously mentioned shortcomings, we have proposed a hybrid approach that relies on 1D modified maps.

The structure of the remaining parts of this thesis is summarized as follows:

Chapter 1 provides a thorough overview of chaotic maps, which are integral components of dynamic systems. We discuss these systems and their significance and also the distinction between chaos and randomness. Next, we review chaotic

maps, including their definition, types, characteristics, and importance in diverse domains.

In Chapter 2, we discuss the characteristics, classifications, and significance of digital images. Additionally, we discuss the importance of grayscale images, which are the subject of the proposed method in this dissertation.

In the first section of Chapter 3, we comprehensively cover cryptography, including its various types, goals, fields, cryptanalysis, and attack methods. The following section focuses on image encryption and its different techniques with a general examination of image encryption algorithms, including their classifications, types, and examples. The third section discusses image encryption algorithms based on chaotic maps, which is highly significant. The following section of this chapter focuses on Image Encryption Evaluation Metrics. The final section of this chapter focuses on related works.

Chapter 4 encompasses a detailed discussion of the proposed image encryption algorithm, while the subsequent chapter provides a comprehensive focus on performance and security analysis.

Finally, we complete our thesis with a general and thorough conclusion, including an overview of my study effort and some proposals for future developments.

Chapter 1

Chaotic maps and their Applications

Introduction

The method described in this dissertation utilizes chaotic maps as a tool. Therefore, we provided the first chapter to explaining this technique, including its different types and applications, due to its significant importance.

Since chaotic maps are essential elements of dynamic systems, it is crucial to offer a preliminary overview of these systems. This includes defining them, describing their different types, and providing an explanation of their importance and applications in various fields.

1.1 Dynamic system

Dynamic systems refer to systems that change their state over time and provide a framework for evaluating these changes [19]. The state encompasses all pertinent data on the system at a specific instance, such as the pendulum's tilt and angular velocity or the population of rabbits in a forest. The evolution rule governs the transformation of the system's state, which can be either deterministic or stochastic, dependent on its current state.

A dynamical system is a mathematical construct composed of three components:

- **State Space (S):** The state space encompasses all conceivable states the system can occupy, ranging from an essential numerical continuum to a sophisticated, multi-dimensional realm.
- **Time Set (T)**: The time set is typically represented by real numbers ($\mathbb{R}$) to denote continuous time, although it can alternatively be discrete time steps for systems that progress in discrete intervals.
- **Evolution Rule (R)**: The evolution rule is a mathematical function that maps a state ($s \in S$) and a time point ($t \in T$) to the future state ($R(s,t) \in S$)

at that specific time. This rule encapsulates the fundamental nature of how the system dynamics evolve.

Dynamic systems have a wide range of real-world applications in many fields, demonstrating their richness. for example:

- Physics: Orbital mechanics (forecasting planetary motion)
- Studying the intricate dynamics of fluids in a state of chaos (to gain a deeper knowledge of weather patterns, e.g.)
- Biology: Predator-prey interactions (modeling population fluctuations)
- The study of epidemic modeling: analyzing how diseases spread and propagate.
- Economics: Examining the fluctuations in economic activity known as business cycles, which involve periods of expansion and contraction.
- modeling supply and demand to properly understand market dynamics.

1.1.1 Difference between chaos and randomness

In the context of dynamical systems, chaotic behaviour can sometimes be mistaken for random processes because they appear similar. However, chaos and randomness (or stochasticity) are related but two different and distinct concepts [107].

The key difference between chaos and randomness lies in their underlying mechanisms:

- Chaos is a deterministic phenomenon, where the behavior is governed by strict rules, but the complexity of these rules leads to unpredictability.
- Randomness is a fundamentally probabilistic phenomenon, where the behavior is influenced by chance or uncertainty.

Whereas

- Chaos refers to complex and unpredictable behaviour that can emerge in certain types of dynamic systems.
- Chaotic systems are deterministic, which means that their future evolution is entirely determined by their initial conditions and the rules governing their dynamics.

- Even small variations in initial conditions can lead to significant differences in future trajectories of the system, also known as sensitivity to initial conditions or the butterfly effect.
- Chaotic behaviour is characterised by extreme sensitivity to initial conditions, complex dependence on system parameters, and an appearance of disorder. Despite their random appearance, chaotic systems obey precise rules.

In contrast,

- randomness refers to the absence of patterns or regularities in a process or set of data.
- Random events are unpredictable and cannot be determined by rules or causal laws. They are characterised by a uniform distribution of possible outcomes without any discernible correlation or structure.
- Randomness is intrinsically independent of context or initial conditions.

1.2 Chaotic Maps

1.2.1 Definition of Chaotic Maps

Chaotic maps are mathematical functions that map an input value to an output value in an iterative fashion, generating a sequence of values over time. These maps are used to model and understand complex, unpredictable systems in various fields, such as physics, engineering, and computer science. They exhibit chaotic behavior, extreme sensitivity to initial conditions, unpredictable long-term behavior, and complex dynamics [11].

The general formula for a chaotic map can be written as:

$$x_{n+1} = f(x_n) \tag{1.1}$$

Where:

- $x_n is$ the current state of the system at time n
- x_{n+1} is the next state of the system at time $n + 1$
- $f(x_n)$ is the chaotic map function that takes the current state x_n and generates the next state x_{n+1}

1.2.2 Characteristics of chaotic maps

Chaos is a complex phenomenon with several distinguishing characteristics [23, 79, 81]:

1. **Sensitivity to initial conditions:** This is the most basic feature of chaos. Small changes in the initial conditions of a chaotic system can cause significant differences in its long-term behavior. This sensitivity is frequently demonstrated by the butterfly effect, in which the flapping of a butterfly's wings can trigger a hurricane on the other side of the world.

2. **Determinism:** Chaos is a deterministic phenomenon, which means that a chaotic system's future behavior is entirely determined by its starting conditions and the rules that govern its dynamics. However, sensitivity to initial conditions makes long-term predictions impossible, as minor errors in measuring initial conditions can result in large errors in predictions.

3. **Unpredictability:** Chaotic systems are unpredictable, making it impossible to predict their long-term evolution. Even if we know the system's initial conditions and rules, small initial variations can result in dramatically different outcomes, making predictions impossible.

4. **Non-periodicity:** Chaotic systems do not have periodic behavior. Their phase space trajectory is not a fixed point, a closed cycle, or a simple combination of the two. Instead, the trajectory is erratic and unpredictable, but it is limited to a specific region of phase space.

5. **Ergodicity:** Chaotic systems are ergodic, which means they progress through all phases over time. This property allows chaotic systems to be characterized by probability distributions rather than specific trajectories.

6. **Fractality:** Chaotic systems frequently have fractal structure, which means they contain self-similar details at different scales. This fractal structure is reflected in bifurcation diagrams and strange attractors found in chaotic systems.

7. **Pseudorandomness:** Chaotic Maps can generate pseudorandom sequences with desirable statistical properties. These sequences can be used in cryptographic algorithms, simulations, and various methods.

1.2.3 Butterfly effect

The butterfly effect is a concept in chaos theory that describes how very slight changes in initial conditions can result in huge changes in long-term outcomes [25]. It popularises the notion that a butterfly fluttering its wings can trigger a tornado on the other side of the earth. It was coined by meteorologist Edward Lorenz in 1961 while studying weather patterns. The premise is that even minor perturbations can result in enormous and unforeseen changes in a complex system, such as the weather. This great sensitivity to initial conditions is typical of chaotic systems. The butterfly effect illustrates complex dynamic systems' interconnectedness and unpredictable behavior, emphasizing the importance of understanding nonlinear dynamics and the limitations of deterministic modeling. The butterfly effect illustrates the inherent unpredictability and difficulty of long-term forecasting or predicting the behavior of specific systems. It suggests that even the most minor and seemingly insignificant factors can contribute to significant changes and outcomes, making it challenging to precisely determine or control the future behavior of complex systems.

1.2.4 Types of chaotic maps

Several types of chaotic maps have been studied in the field of chaos theory. They can be categorized based on their dimension into two categories (Figure 1.1):

- One-Dimensional Maps (1D).
- Multi-Dimensional Maps (MD).

1.2.4.1 One-dimensional maps

One-dimensional (1D) chaotic maps are mathematical models that describe the evolution of a single variable over time. They are defined by a simple equation that iteratively updates the value of a single variable, often with a nonlinear relationship [67].

Advantages:

- **Simplicity:** 1D chaotic maps are relatively simple to analyze and understand, making them a good starting point for studying chaotic dynamics.
- **Simple to implement:** 1D Chaotic Maps are easy to understand and implement.
- **Fast computation:** Since they only involve a single variable, 1D Chaotic Maps can be computed quickly, even for large numbers of iterations.

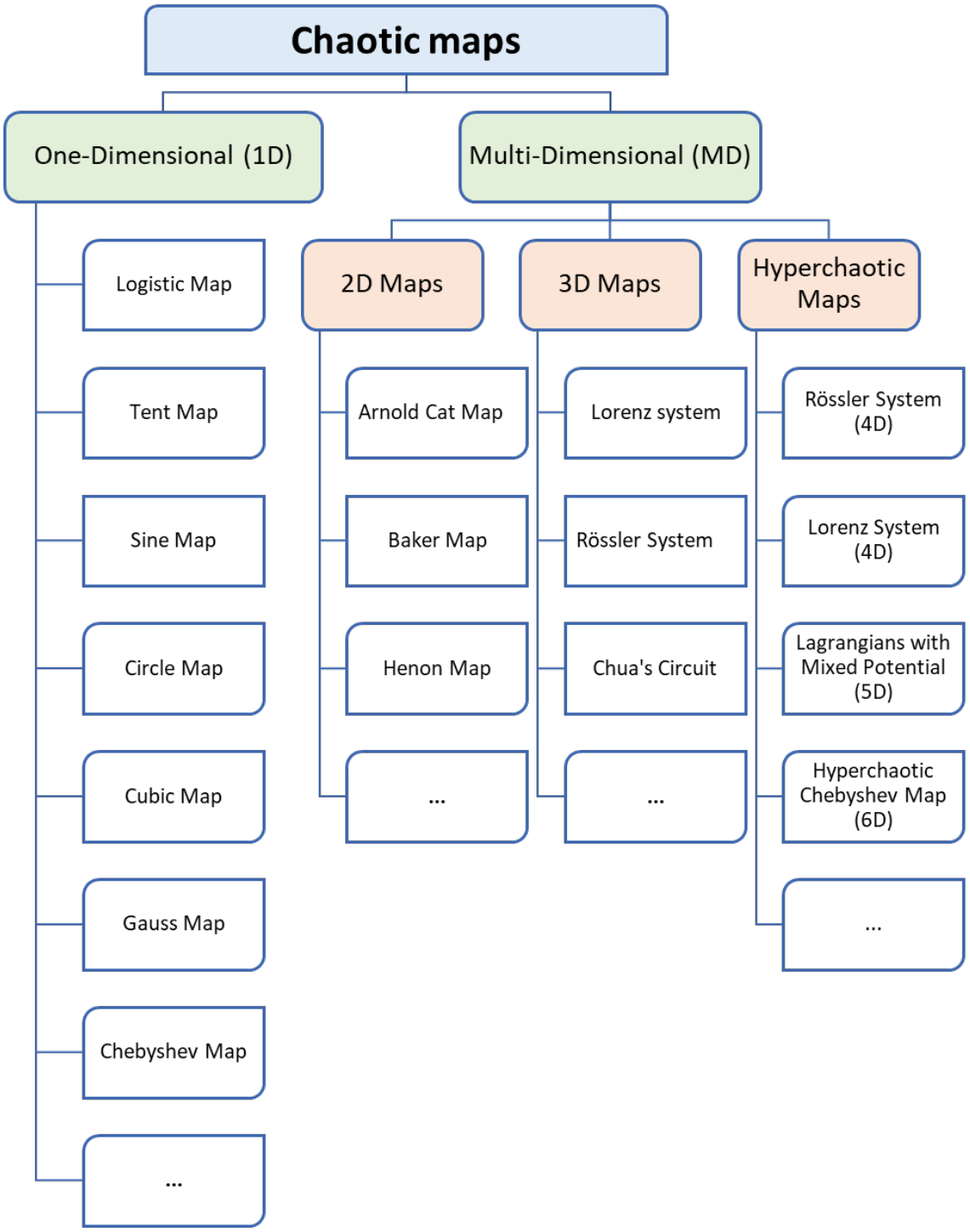


Figure 1.1: Types of Chaotic Maps

- **Rich dynamic behavior:** Despite their simplicity, 1D chaotic maps can display a wide range of complex and interesting dynamical behaviors, such as periodic orbits, bifurcations, and strange attractors.
- **Computational Efficiency:** 1D chaotic maps are computationally efficient, requiring only basic arithmetic operations to iterate from one time step to the next.

Disadvantages:

- **Limited Complexity:** While 1D chaotic maps can exhibit complex behavior, they are not as rich as higher-dimensional chaotic systems, which can exhibit more intricate and unpredictable dynamics.
- **Limited control:** The parameters of 1D Chaotic Maps can be difficult to adjust to achieve specific behavior, limiting their control and applicability.
- **Limited Flexibility:** 1D chaotic maps can only capture one-dimensional dynamics, which may be insufficient for modeling certain real-world phenomena involving multiple interacting variables. This limits their flexibility in modeling complex systems, as they only consider one variable.
- **Lack of Robustness:** Because they only have one variable, they may be less robust to perturbations and noise.
- **Lack of Spatial Structure:** 1D chaotic maps lack spatial structure, which can limit their applicability in certain fields, such as fluid mechanics and image processing.

Here are some examples of 1D chaotic maps:

1. **Logistic Map** A classic example of a chaotic map that demonstrates how complex, chaotic behavior can arise from simple non-linear dynamical equations.

$$x_{n+1} = r x_n (1 - x_n) \tag{1.2}$$

   Parameter: r (typically $0 < r \leq 4$)

2. **Tent Map** A piecewise linear map that exhibits chaotic behavior for certain parameter values.

$$x_{n+1} = \begin{cases} \mu x_n & \text{if } x_n < \frac{1}{2} \\ \mu(1 - x_n) & \text{if } x_n \geq \frac{1}{2} \end{cases} \tag{1.3}$$

Parameter: μ (typically $0 < \mu \leq 2$)

3. **Sine Map** Uses the sine function to generate chaotic behavior, particularly interesting for its smooth, continuous nature.

$$x_{n+1} = r \sin(\pi x_n) \tag{1.4}$$

Parameter: r (typically $0 < r \leq 1$)

4. **Cubic Map** A polynomial map that extends the logistic map to cubic nonlinearity, resulting in rich dynamical behavior.

$$x_{n+1} = r x_n (1 - x_n^2) \tag{1.5}$$

Parameter: r (typically $0 < r \leq 4$)

5. **Gaussian Map** Generates chaotic sequences using a Gaussian function.

$$x_{n+1} = \exp(-a x_n^2) + b \tag{1.6}$$

Parameters:a, b (typically $a > 0$, b is a small constant)

6. **Piecewise Linear Map** Simple piecewise linear function that can produce chaos for appropriate parameter choices.

$$x_{n+1} = \begin{cases} a x_n & \text{if } x_n < \frac{1}{2} \\ b(1 - x_n) & \text{if } x_n \geq \frac{1}{2} \end{cases} \tag{1.7}$$

Parameters: a, b (typically $0 < a, b \leq 2$)

7. **Quadratic Map** Another polynomial map, similar to the logistic map but with quadratic nonlinearity.

$$x_{n+1} = r x_n (1 - x_n^2) \tag{1.8}$$

Parameter: r (typically $0 < r \leq 2$)

8. **Iterated Function System (IFS) Map** Uses probabilistic switching between two linear functions to generate complex behavior.

$$x_{n+1} = \begin{cases} ax_n & \text{with probability } p \\ b(1-x_n) & \text{with probability } 1-p \end{cases} \tag{1.9}$$

Parameters: a, b, p (typically $0 < a, b \leq 2$, $0 \leq p \leq 1$)

9. **Chebyshev Map** Based on Chebyshev polynomials, this map exhibits chaotic behavior for certain values of n.

$$x_{n+1} = \cos(n \cos^{-1}(x_n)) \tag{1.10}$$

Parameter: n (typically a positive integer)

10. **Circle Map** Models the behavior of a point on a circle subjected to periodic driving, leading to chaos for certain parameter values.

$$x_{n+1} = x_n + \Omega - \frac{K}{2\pi}\sin(2\pi x_n) \mod 1 \tag{1.11}$$

Parameters: Ω, K (typically $0 < K \leq 1$, Ω is a constant)

1.2.4.2 Multi-Dimensional Maps

Unlike One-dimensional; Multi-Dimensional (MD) chaotic maps are mathematical models that describe the evolution of several variables over time. They are defined by a set of equations to update the values of variables, MD Chaotic Maps are often used to model complex systems with multiple interacting variables [71, 5].

Advantages:

- **More complexity** : MD Chaotic Maps frequently incorporate complex equations and nonlinear dynamics, allowing them to capture elaborate and unpredictable behaviors. This makes them very useful for modeling phenomena that exhibit chaotic features. They can also model real-world systems with several interacting variables, such as fluid dynamics, weather patterns, or biological systems.

- **Fractal Generation**: Many MD Chaotic Maps can form fractal patterns, complicated and self-similar structures in nature and mathematics. Fractals have uses in computer graphics, picture compression, and the study of complex systems.

- **A greater number of parameters**: can make it more complex to detect and study the range of chaotic behavior. This property can also be considered an excellent feature for MD maps.

Disadvantages :

- **Increased complexity**: which can make them more difficult to understand and analyze.
- **Computational Complexity**: MD Chaotic Maps involve complex equations and require significant computational resources to simulate or analyze. This can limit their practicality in real-time applications or systems with limited computational capabilities. MD Chaotic Maps can be computationally expensive, especially for high-dimensional systems.
- **Stability and Control**: Chaotic systems, including MD Chaotic Maps, can be difficult to control or stabilize due to their inherent sensitivity to initial conditions. This can make it challenging to harness their chaotic properties for practical applications that require stability and predictability.

Here are some examples of multidimensional chaotic maps with their equations:

1. **Lorenz Attractor**: The Lorenz attractor is a three-dimensional chaotic system [9], described by the following set of differential equations:

$$\begin{cases} \frac{dx}{dt} & = \sigma(y - x) \\ \frac{dy}{dt} & = x(\rho - z) - y \\ \frac{dz}{dt} & = xy - \beta z \end{cases} \tag{1.12}$$

Where σ, ρ, and β are parameters that control the system's behavior.

2. **Rössler Attractor**: The Rössler attractor is another three-dimensional chaotic system, described by the following set of differential equations:

$$\begin{cases} \frac{dx}{dt} & = -y - z \\ \frac{dy}{dt} & = x + ay \\ \frac{dz}{dt} & = b + z(x - c) \end{cases} \tag{1.13}$$

Where a, b, and c are parameters that control the system's behavior.

3. **Hénon Map**: The Hénon map is a two-dimensional chaotic map, defined by the following equations:

$$\begin{cases} x(n+1) & = 1 - ax(n)^2 + y(n) \\ y(n+1) & = bx(n) \end{cases} \tag{1.14}$$

Where $a > 1.2$ and $b > 0.3$ are parameters that control the system's behavior.

4. **Ikeda Map**: The Ikeda map is a two-dimensional chaotic map, defined by the following equations:

$$\begin{cases} x(n+1) & = 1 + u[x(n)\cos(t) - y(n)\sin(t)] \\ y(n+1) & = u[x(n)\sin(t) + y(n)\cos(t)] \\ t & = 0.4 - \frac{6}{1+x(n)^2+y(n)^2} \end{cases} \tag{1.15}$$

Where u is a parameter that controls the system's behavior.

5. **4D Lü System**: This system demonstrates hyperchaotic behavior with four dimensions [69].

$$\begin{cases} \dot{x} & = a(y - x) \\ \dot{y} & = x(c - z) - y \\ \dot{z} & = xy - bz \\ \dot{w} & = -ex + fy \end{cases} \tag{1.16}$$

Here, $\dot{x}$, $\dot{y}$, $\dot{z}$, and $\dot{w}$ represent the time derivatives, and a through f are parameters. Common values for hyperchaotic behavior include: $a = 36$, $b = 3$, $c = 20$, $d = 0.05$, $e = 4$, and $f = 0.06$.

6. **5D Generalized Chua System**: This map extends the classic Chua's circuit to five dimensions.

$$\begin{cases} \dot{x} &= \alpha(y - x - h(x)) \\ \dot{y} &= x - ay - z \\ \dot{z} &= y \\ \dot{u} &= mx - ny - pu \\ \dot{v} &= qx - ry + sv \end{cases} \tag{1.17}$$

Here, $\dot{x}$ through $\dot{v}$ represent the time derivatives, $h(x)$ is a piecewise linear function defining non-linearity, and α, a, m through s are parameters. Specific parameter values can be found in research papers for achieving hyperchaotic behavior.

7. **4D Scroll Chaotic System**: This system exhibits hyperchaotic behavior with a scroll-like trajectory in four dimensions.

$$\begin{cases} \dot{x} &= a(y - x) \\ \dot{y} &= dx - y - xz \\ \dot{z} &= xy - bz \\ \dot{w} &= ky - cz \end{cases} \tag{1.18}$$

Here, $\dot{x}$ through $\dot{w}$ represent the time derivatives, and a through k are parameters. Common values for hyperchaotic behavior include: $a = 35$, $b = 3$, $c = 28$, $d = -10.5$, and $k = 8/3$.

1.2.5 Overview of 1D seed chaotic maps

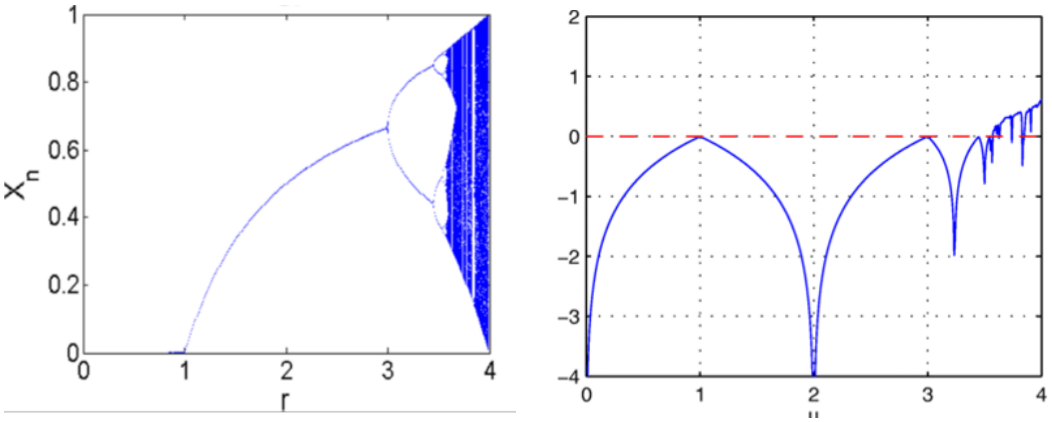
The chaotic maps used in our proposed algorithm are improved and modified from elementary ones. For this reason, we thought it worth mentioning an overview of seed Chaotic Maps at the beginning. Next, we will touch on the maps that were used.

1.2.5.1 Logistic maps

The logistic map is the best-known chaotic map and one of the best-studied chaotic systems. It is often used in image encryption systems because of its ease of generation. Its mathematical equation is shown below (1.19):

$$x_{n+1} = rx_n(1 - x_n) \tag{1.19}$$

Where x refers to the random generated set of numbers,x_0 is the initial value of the map and x_n is the output sequence with $x_n \in [0, 1]$, r (chaos multiplier) is the control parameters with range of $(0 < r \leq 4)$. The chaotic behaviour of the Logistic map is observed in the range $[3.5, 4]$. However, its chaotic properties are not so good, Which is clearly shown by the bifurcation and Lyapunov exponent shown in the figure 1.2a 1.2b



(a) The bifurcation diagrams. (b) The Lyapunov exponent.

Figure 1.2: The bifurcation and The Lyapunov exponent of Logistic map [86].

1.2.5.2 May maps

It is published by Robert May [55, 81], the May map has behaviour and properties similar to that of the Logistic map and is expressed by the following equation (1.20) [65]:

$$x_{n+1} = x_n \exp(a(1 - x_n)) \tag{1.20}$$

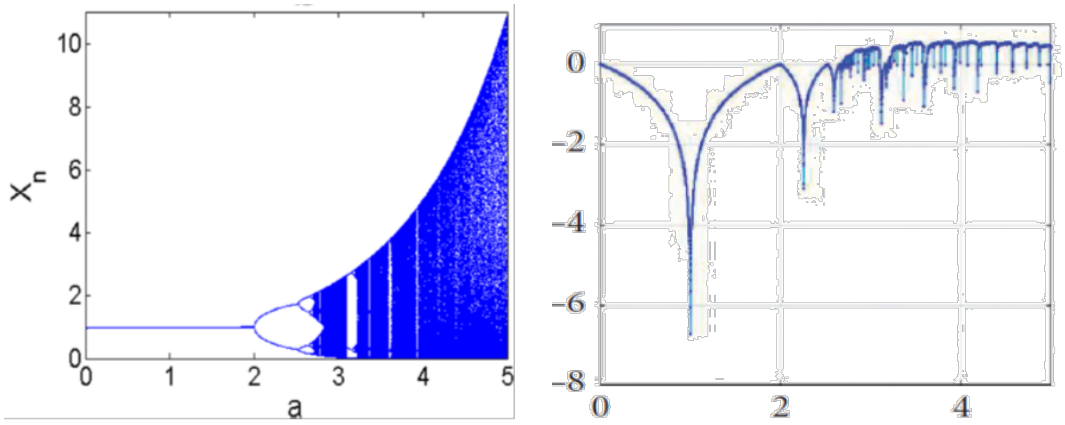
Where $x_n \in [0, 10.9]$and the control parameter a belongs to the range $[0, 5]$. Figure 1.3a illustrates the bifurcation diagram of May map in which, we can observe a non uniform data output distribution and periodicity (expressed by blank space) in the range of [2.6, 5]. The Lyapunov exponent are shown in Figure 1.3b [33]

1.2.5.3 Sine map

Sine map is one of the 1D chaotic maps that has similar chaotic behavior to the Logistic map, its structure is defined as:

$$x_{n+1} = K \sin(\pi . x_n) \tag{1.21}$$

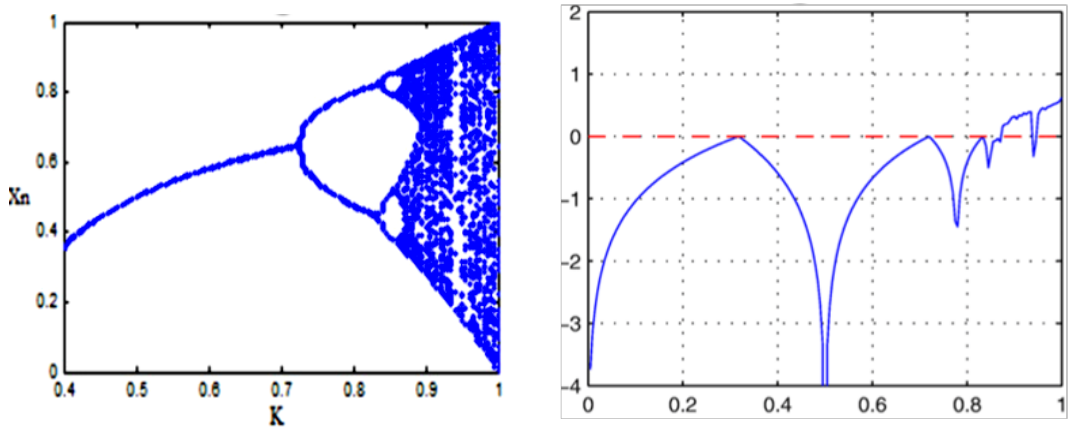
Where $K \in (0, 1]$ and x_n is in $(0, 1)$. The bifurcation diagram and Lyapunov



(a) The bifurcation diagrams. (b) The Lyapunov exponent.

Figure 1.3: The bifurcation and The Lyapunov exponent of May map [4].

exponent are shown in Figure 1.4a and Figure 1.4b, respectively. Then, We can see that chaotic properties are weak, especially if K is less than 0.8. Some isolated values such as K = 0.941 appear to show non-chaotic behavior and generates periodic sequences at output which are not random and unsuitable for encryption [66].



(a) The bifurcation diagrams. (b) The Lyapunov exponent.

Figure 1.4: The bifurcation and The Lyapunov exponent of Sine map [66].

1.2.6 Evaluation of chaotic maps

Chaotic maps are a fundamental concept in chaos theory, where dynamic systems produce random states governed by initial seed conditions. To analyze and measure the chaotic behavior of these maps, several key tools are employed. These tools include:

1.2.6.1 Bifurcation diagram

The bifurcation diagram is a powerful tool for analyzing the behaviour of chaotic maps. It is a graphical representation that shows the different behaviours of a dynamic system for each value of a bifurcation parameter. It illustrates how the system's behaviour changes as the parameter is varied.

In a bifurcation diagram, the parameter of the chaotic map is plotted on the horizontal axis, while the resulting values of the map are plotted on the vertical axis. As the parameter varies, the diagram shows how the map's behaviour changes, revealing the different regions of stability, periodic behaviour, and chaos. chaotic regions appear as dense, irregularly shaped areas where the map exhibits sensitive dependence on initial conditions and unpredictable behaviour.

To create a bifurcation diagram for a chaotic map, you can use the following steps:

- Choose a range of values for the bifurcation parameter.
- For each value of the parameter, iterate the system several times to let it converge to the attractor.
- Collect points on the attractor for each value of the parameter.
- Plot the points on the y-axis against the parameter values on the x-axis.

The bifurcation diagram is particularly useful for understanding the transition from order to chaos in chaotic systems. It can help identify the critical parameter values at which the system undergoes bifurcations, where the dynamics change abruptly from one type of behaviour to another.

1.2.6.2 Lyapunov Exponent

The Lyapunov Exponent (LE) is a mathematical tool measuring sensitivity to initial conditions in chaotic maps[66, 99]. it is crucial in analyzing and characterizing chaotic dynamic systems. It quantifies the rate of divergence of nearby trajectories in a dynamical system. It provides insights into these systems' stability, predictability, and complexity. Hence, we can decide whether or not the system will be chaotic. A positive Lyapunov exponent indicates chaotic behaviour. It can be calculated as:

$$\lambda = \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln|f'(x_i)|. \tag{1.22}$$

A high Lyapunov exponent indicates fewer iterations and transient effects, resulting in two totally different PRNS from two very close initial conditions with the same control parameter.

1.2.7 The importance of chaotic maps

Chaotic maps can serve as a connection between randomness and determinism due to their distinct attributes. Through the examination of their apparently unpredictable behaviors, scientists can gain insight into complex systems in the real world. Therefore, chaotic maps play a crucial role in numerous scientific fields, as they have significant influence and allow us to:

- Generating Pseudo-Random Numbers: Chaotic maps excel at creating sequences of numbers that appear random. While not truly random like flipping a coin, these sequences are good enough for many applications. The secret lies in their extreme sensitivity to initial conditions. Minute differences in the starting point of a chaotic map can lead to drastically different outputs down the line. This very property makes them excellent tools in cryptography. In encryption algorithms, chaotic maps can be used to generate unpredictable keys that scramble data, making it extremely difficult to crack. Their role extends beyond cryptography as well. They can be used in simulations where random numbers are needed, such as modeling complex physical phenomena or financial markets.

- Understanding Complex Systems: Chaotic maps act as simplified models for real-world systems that exhibit chaotic behavior. Studying these maps allows us to understand how chaos unfolds in these complex systems. Chaos, though seemingly random, often follows underlying rules. By analyzing chaotic maps, scientists can gain insights into how these systems evolve over time, even if predicting their exact future state remains elusive.

- Modeling Chaos: Chaotic maps, such as the Logistic Map and the Lorenz Attractor, provide simplified models that capture the essence of chaotic behavior. By studying these maps, researchers can gain insights into the mechanisms that generate chaos in real-world dynamic systems.

- Understanding Unpredictability: Chaotic maps illustrate how minor changes in initial conditions can lead to drastically different long-term outcomes. This underscores the inherent unpredictability of chaotic systems, which is crucial for fields like weather forecasting, ecology, and finance.

- Generating Complex Patterns: Surprisingly simple chaotic maps can produce intricate, fractal-like patterns that mimic the structures observed in nature, from coastlines to cloud formations. Analyzing these patterns can lead to a better understanding of the underlying principles governing complex systems.
- Applications in Cryptography: The sensitivity to initial conditions in chaotic systems has led to their use in cryptographic applications. Chaotic maps can be leveraged to generate unpredictable sequences for encryption, enhancing the security of data transmission.

These effects apply in various fields. Some examples of the use of chaotic maps and their application in some fields can be listed, including:

- Physics: Chaotic maps can be used to model chaotic phenomena like turbulent fluid flow or the erratic motion of a bouncing ball. Understanding these chaotic behaviors is crucial in various areas of physics, from designing airplanes to predicting weather patterns.
- Biology: Population dynamics, the growth and decline of species in an ecosystem, can exhibit chaotic behavior. Chaotic maps can help model these dynamics, allowing biologists to study how factors like predation and competition affect populations.
- Economics: Economic systems are inherently complex, and chaotic maps can be used to model fluctuations in markets, price changes, and even economic crashes. While not perfect predictors, these models can provide valuable insights into how economies behave under different conditions.

1.2.8 Conclusion

This chapter provides a comprehensive exploration of chaotic maps, delving into their fundamental importance in the field of nonlinear dynamics. We examined the diverse classifications of chaotic maps, highlighting their unique characteristics and behaviors. Additionally, the chapter delved into the dynamic systems, discussing their essential attributes and exploring the various classifications that categorize them. By understanding these concepts, we aim to establish a solid foundation for further exploration of chaotic phenomena and their applications in various scientific disciplines.

Chapter 2

Digital images

Introduction

Digital images play a critical role in today's world, significantly impacting fields such as medicine, military, social media, and entertainment. The advancement of camera technology has made high-quality image capture widely accessible, while robust networking infrastructures have enabled the seamless sharing of images globally. In medicine, digital images are vital for diagnostics, treatment planning, and telemedicine. In the military, they are essential for surveillance, intelligence, and strategic operations. Digital images also enhance scientific research, entertainment, marketing, and education by providing detailed visualization and engagement. As technology advances, the importance of digital images continues to grow, further shaping our reality and driving innovation.

In this chapter, we will discuss digital images, focusing on their definition, types, importance, and other related aspects.

2.1 Image definition

A digital image is a visual representation of an object or scene, converted into a format for storage, processing, and display on digital devices. It is composed of a grid of tiny pixels, each with a specific color value, typically represented by a combination of red, green, and blue (RGB) values. Digital images are created using binary data, created through scanning physical images or capturing scenes with digital cameras.

In mathematics, particularly in the context of digital image processing, a "digital image" is defined as a discrete representation of a visual image or an array of quantized values derived from the visual spectrum. More formally,

Mathematically, let I be a digital image. Then I can be represented as:

$$I = \{I(i,j) | 1 \leq i \leq M, 1 \leq j \leq N\} \longmapsto V \tag{2.1}$$

Where

- M and N are the dimensions of the image (height and width, respectively), and
- Each pair (i, j) corresponds to a pixel location.
- $I(i, j)$ is the intensity value at the pixel located in the $i - th$ row and $j - th$ column.
- For grayscale images, $I(i, j)$ is typically a single value representing the brightness.
- For color images, $I(i, j)$ might be a vector, such as $[R(i, j), G(i, j), B(i, j)]$, representing the red, green, and blue color channels, respectively.
- V is the set of possible pixel values. For grayscale images, V typically consists of integers within a certain range, such as $[0, 255]$, where 0 represents black and 255 represents white.

The Figure 2.1 is a visual representation of a grayscale digital image. Each pixel in this 10x10 grid has an intensity value ranging from 0 (black) to 255 (white), with varying shades of gray in between. Each element in the image corresponds to the intensity value of a pixel at that coordinate, illustrating the mathematical concept of a digital image, and how digital images are represented using discrete pixel density values.

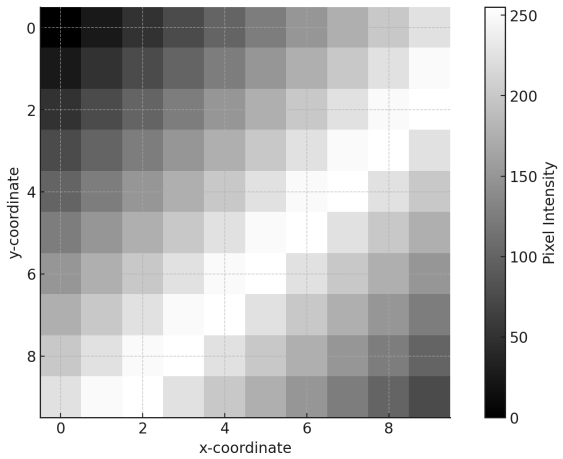


Figure 2.1: Grayscale Digital Image Representation

2.2 Types of Images

Two main categories of digital images exist [13]: raster (or bitmap) images and vector images.

2.2.1 Raster images

Raster images are generated using pixels or dots, and their quality is determined by their resolution. They are composed of a grid of pixels, each with a specific color value.They are most appropriate for intricate photographic photographs, as they can capture a broad spectrum of colors and gradients. Raster image file formats such as JPG, TIFF, PNG, and GIF are some examples.

2.2.2 Vector images

Conversely, vector images are derived from mathematical equations that define the characteristics of shapes, colors, and positioning. Instead of using pixels.These objects are resolution-independent, indicating that they can be resized to various dimensions without any loss in quality. Vector pictures are well-suited for designing layouts, logos, illustrations, and visuals that need to be resized frequently. EPS and SVG are two examples of vector image file formats.

The table 2.1 shows more differences between the two types

Feature	**Raster Images**	**Vector Images**
Building	Blocks Pixels (colored squares)	Mathematical paths (lines, curves)
Composition	Individual pixels arranged in a grid	Mathematical formulas defining shapes and lines
Resolution	Dependent on DPI	Independent of resolution
Scalability	Limited - Loses quality when enlarged	High - Can be resized infinitely without quality loss
Transparency	Limited support	Transparent background by default
Suitability	Photographs, complex illustrations, realistic details	Logos, icons, cartoons, line art, scalable graphics
Editing	More challenging (Manipulating individual pixels)	Easily edited and manipulated (Adjusting shapes, paths, and fills)
Rendering	Require more processing power to render	Render quickly and smoothly at any size
File Formats	JPEG, PNG, GIF, BMP, TIFF	EPS, SVG, AI
File Size	Generally larger	Generally smaller

Table 2.1: Raster vs. Vector Images: A Comparison

2.3 Color modes

Color modes are systems that define the ways colors are represented in digital images. Different color modes are used depending on the type of image and its intended use. Here's a detailed look at the primary color modes and their applications [13]:

2.3.1 RGB (Red, Green, Blue)

RGB is an additive color model used in electronic displays, such as computer monitors, TVs, and smartphones. It works by combining different intensities of red, green, and blue light to create a wide range of colours.

2.3.2 CMYK (Cyan, Magenta, Yellow, Black)

CMYK is a subtractive color model used in print media, such as magazines, books, and posters. It works by using varying amounts of cyan, magenta, yellow, and black inks to absorb and reflect light, creating the desired colours.

2.3.3 Grayscale

Grayscale is a monochromatic colour mode that uses various shades of grey, ranging from pure black to pure white. It is often used for black-and-white photography, technical illustrations, and documents where colour is not necessary.

2.3.4 Indexed Color

This colour mode uses a limited palette of colours, typically 256 or fewer. It is often used in older or smaller digital images, as it can result in smaller file sizes compared to RGB or CMYK.

2.3.5 Bitmap

Bitmap (binary colour) Consists of black and white pixels, with no colours or shades of grey. Suitable for creating the effect of hand-drawn sketches or line drawings.

2.3.6 HSL (Hue, Saturation, Lightness)

HSL and HSV These are alternative colour models that represent colour in terms of hue, saturation, and lightness or value. They can be more intuitive for some users when working with colour. where

- **Hue:** Represents the color on the color wheel (e.g., red, green, blue). **Saturation:** Represents the intensity/purity of the color (e.g., how much white is mixed in). **Value:** Represents the brightness of the color (e.g., how much black is mixed in).

2.3.7 HSV (Hue, Saturation, Value)

HSV This is a cylindrical-coordinate representation of RGB colours. Hue represents the colour wheel, saturation is the intensity/purity of the colour, and value is the brightness.

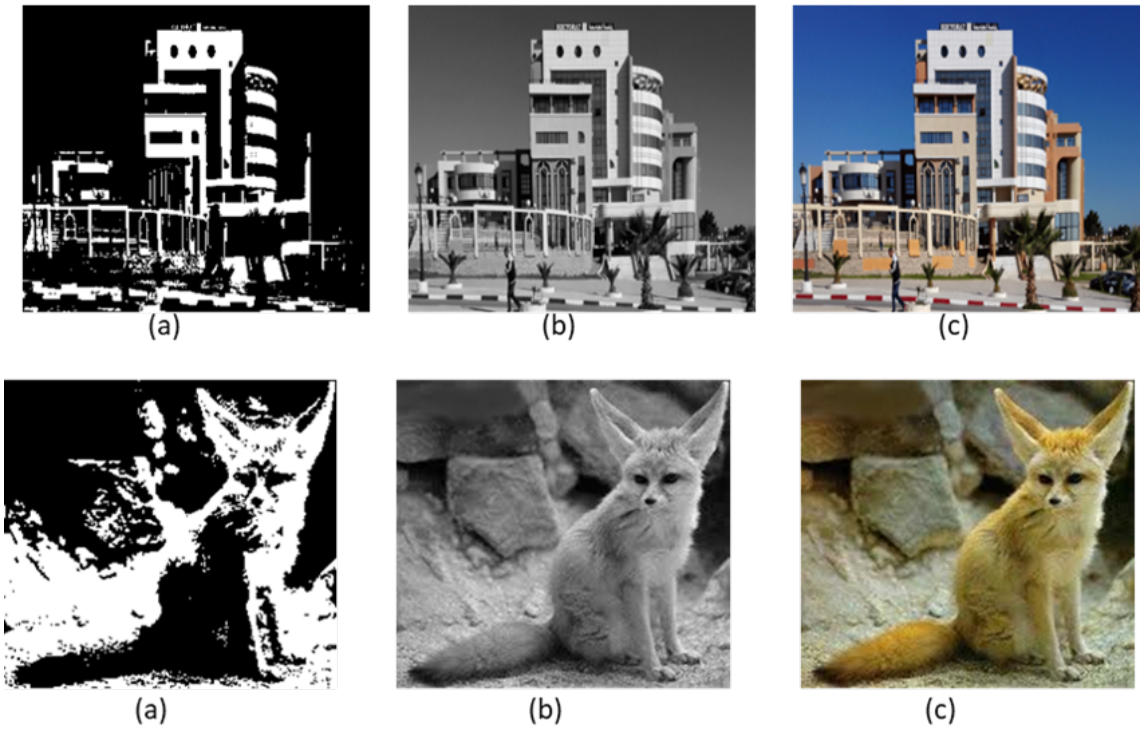


Figure 2.2: Different version of the same image: (a) black and white, (b) grey-level, (c) colour.

2.4 The resolution

The resolution of a digital image is the number of pixels within the image, typically measured in pixels per inch (PPI) or dots per inch (DPI). A higher resolution results in more pixels, resulting in greater detail and clarity. The resolution can also be described in terms of the overall number of megapixels contained in the image, obtained by multiplying the height and width figures together and dividing by 1 million.

Higher resolution images have more pixels and can display finer details, while lower resolution images have fewer pixels and appear more pixelated or blurry. The higher the PPI/DPI, the sharper and more detailed the image will appear.

Image resolution impacts file size [13], as higher resolutions result in larger file sizes, which can affect storage and transmission speeds

2.5 Color depth

Color depth, also known as bit depth, is the number of bits used to represent the color of each pixel in a digital image. It determines the range of colors that can be represented and recorded, quantifying the level of color variation and fidelity captured by a camera or stored in a file.

It is typically measured in bits per pixel (bpp) or bits per channel (bpc), with each color component having the same number of bits.

Higher color depths allow for a wider range of colors and greater color fidelity, but result in larger file sizes. Lower color depths may lead to visible artifacts and limited color accuracy.The appropriate color depth to use depends on the specific application and the desired balance between image quality and resource requirements.

There are several popular color depth values:

- **1-bit (Black and White)**: This is the simplest color depth, representing each pixel as either black or white.An image with 1 bpp is called a binary image
- **8-bit color**: This allows for 256 unique colors (2^8). This is the minimum color depth used for most digital images and is common for basic graphics and user interfaces.
- **16-bit color**: Also known as high color, this allows for 65,536 unique colors (2^{16}). It provides more color information than 8-bit but is less common than 24-bit or 32-bit.
- **24-bit color**: Also known as true color, this allows for 16.7 million unique colors (2^{24}). This is the standard for most digital images, photographs, and high-quality graphics.
- **32-bit color**: This includes 24 bits for color information plus an additional 8 bits for an alpha channel, which represents transparency. The extra 8 bits allow for 256 levels of transparency, resulting in a total of over 16 million unique color-transparency combinations.

2.6 File formats

Digital image file formats refer to the several methods by which picture data is saved and structured within a computer file. They exhibit differences in features, benefits,

and drawbacks, and have an important effect on defining the quality, compatibility, and functioning of digital images.

These file formats impact factors such as image quality, file size, transparency, and editing capabilities. Gaining a comprehensive understanding of various file formats will assist you in selecting the most appropriate one for your particular requirements, be it for online use, printing, or editing intentions.

The optimal image file format depends on intended use, quality requirements, file size limitations, and compatibility with software and devices.

File formats can be classified according to their loss of data into two categories:

1. **Lossless formats:** are file formats that compress picture data without sacrificing any information. As a consequence, this leads to reduced file sizes without compromising the quality of the image. Some examples of image file formats include PNG, GIF, and TIFF.

2. **Lossy formats:** compress image data by selectively eliminating some information. This leads to reduced file sizes, but it can also result in a degradation in image quality, particularly when using high compression levels. Some examples of image file formats include JPEG, WebP, and HEIF.

The following is a list of some of the most prevalent file formats for digital images:

- **JPEG (Joint Photographic Experts Group):** This is the most widely used image format, especially for photographs and web images. It offers a good balance between file size and image quality, but it is a lossy format. it Supports 24-bit color depth (16.7 million colors). and Uses lossy compression, which can reduce file size but may introduce some image quality loss.

- **PNG (Portable Network Graphics):** It is a lossless format commonly used for web graphics and images with large solid color areas. It supports transparency and interlacing, allowing images to be displayed gradually as they load. PNG supports up to 48-bit color depth (16.7 million colors with alpha channel) and is suitable for images with text or graphics over backgrounds. It offers lossless compression, supports transparency, and provides sharper images compared to JPEG. PNG is commonly used for web graphics, logos, and transparent elements.

- **GIF(Graphics Interchange Format):** It is an older format used for simple animations and web graphics, offering transparency and a limited color palette

of 256 colors. It supports up to 8-bit color depth and is ideal for small, animated web graphics and icons. GIF is a lossless format, often used for small, animated images with limited colors, such as icons or social media animations.

- **TIFF (Tagged Image File Format):** It is a versatile format used for storing high-quality images and scanned documents, offering a wide range of color depths and compression options. It is widely used in professional photography and desktop publishing, supporting both lossless and lossy compression. TIFF can handle high bit depths, up to 48-bit color and 16-bit grayscale, and is ideal for high-quality printing and image editing workflows. It maintains image data regardless of compression or editing, making it an essential tool for professional photography and graphic design.
- BMP **(Bitmap):** It is a lossless format commonly used for storing high-quality images with a large color palette, but is less efficient in terms of file size.
- **WebP:** It is a Google-developed image format, offers smaller file sizes and comparable image quality, and is supported by most modern web browsers.

Type of image	Resolution	Number of pixels	Image size
VGA	640 x 480	307,200	921,600 B
NTSC	720 x 480	345,600	1.036 MB
Super VGA	1024 x 768	786,432	2.305 MB
HD DVD	1280 x 720	921,600	2.764 MB
HDV	1920 x 1080	2,073,600	6.22 MB
4K	3840 x 2160	8,294,400	24.883 MB
8K	7680 x 4320	33,177,600	99.532 MB
16K	15360 x 8640	132,710,400	398.131 MB
64K	61440 x 34560	2.12×10^9	6.37 GB

Table 2.2: Sizes of different types of images [24]

2.7 Compression techniques

Digital image compression techniques are essential for reducing the size of image files, making them easier to store, transmit, and manage. There are two main categories of image compression techniques [43]: lossless and lossy.

2.7.1 Lossless Compression

Lossless compression techniques reduce file size without any loss of quality. This means the original image can be perfectly reconstructed from the compressed data.

These methods are crucial when image quality cannot be compromised, such as in medical imaging or technical drawings. Here are some common lossless compression methods:

- **Run-Length Encoding (RLE)**: RLE is a simple compression technique where sequences of the same data value (pixels) are stored as a single data value and count. Example: A row of pixels [255, 255, 255, 0, 0, 255] would be encoded as [(255, 3), (0, 2), (255, 1)].
- **Huffman Coding:** uses variable-length codes to represent data. It assigns shorter codes to more frequently occurring pixel values. Example: In an image with pixel values A, B, C, D and frequencies 45, 13, 12, 16, 9, 5, a Huffman tree is built, and each pixel is encoded based on the tree structure.
- **Lempel-Ziv-Welch (LZW)**: Utilized in formats like GIF and TIFF, LZW compression creates a dictionary of pixel sequences and encodes repeated sequences as single codes. Example: The string "ABABABAB" could be compressed by representing repeated patterns with dictionary references.

2.7.2 Lossy Compression

Lossy compression techniques reduce file size by permanently eliminating some image data. This results in some loss of image quality, but the reduction in file size can be significant, these methods achieve much higher compression ratios than lossless techniques. These techniques are widely used for web images, streaming, and other applications where perfect fidelity is not essential. Here are some common Lossy compression methods:

- **JPEG**: JPEG is the most widely used lossy compression technique for photographic images. It works by transforming the image into the frequency domain using the Discrete Cosine Transform (DCT), then quantizing the coefficients, and finally encoding them.
- **Vector Quantization**: This method divides the image into blocks and replaces these blocks with the closest match from a set of block patterns (codebook). Example: An image is divided into 4x4 pixel blocks, and each block is replaced with the closest pattern from a predefined codebook, reducing the amount of data needed to represent the image.
- **Fractal compression**: uses self-similarity in images to compress data. It encodes the image as a set of mathematical fractal functions.

2.7.3 Hybrid Compression

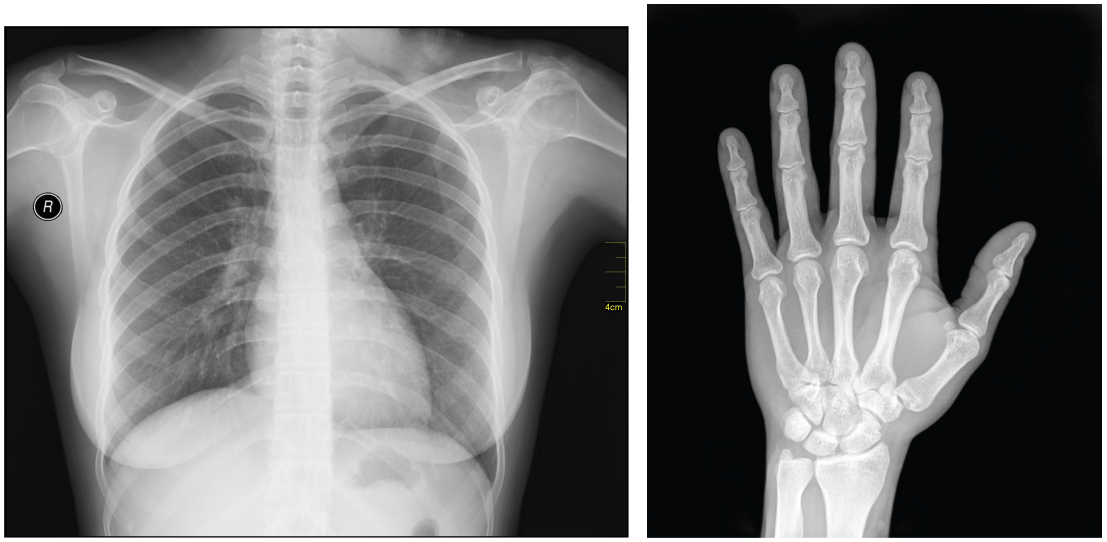
Some compression techniques combine both lossless and lossy methods to optimize compression efficiency and quality, such as:

- **JPEG 2000**: uses wavelet transform instead of DCT and offers both lossless and lossy compression within the same framework. It allows progressive transmission by quality or resolution. The compression can be adjusted to be either lossless or lossy depending on the application.
- **WebP**: WebP supports both lossy and lossless compression. It uses predictive coding for lossy compression, predicting pixel values based on neighboring pixels and encoding only the differences.

2.8 Applications of grayscale images

Despite the advancements in high-quality color photography, grayscale images continue to hold significant importance in various fields due to their simplicity, effectiveness, and ability to highlight certain features without the distraction of color. Here are some reasons why grayscale images are crucial in the medical, military, and other fields:

1. **Medical Field :** Grayscale Images is especially important in medical imaging and pathology
    - **X-rays:** Grayscale images are crucial in diagnosing fractures, infections, and tumors. The different shades of gray represent varying densities of tissues and bones, allowing for detailed examination (Figures 2.3a,2.3b).
    - **CT Scans:** These provide cross-sectional images of the body in grayscale, which helps in visualizing internal organs, blood vessels, and detecting abnormalities such as tumors, clots, and internal injuries (Figures 2.4a, 2.4b).
    - **Magnetic Resonance Imaging (MRI):** MRI scans produce detailed grayscale images of soft tissues. These images are essential for diagnosing conditions related to the brain, spinal cord, and musculoskeletal system, providing high contrast between different tissue types (Figures 2.5a,2.5b).
    - **Ultrasound:** Grayscale ultrasound images allow for visualizing internal body structures such as muscles, tendons, and organs. It is widely used in obstetrics for monitoring fetal development and in cardiology for assessing heart function (Figures 2.6a, 2.6b).

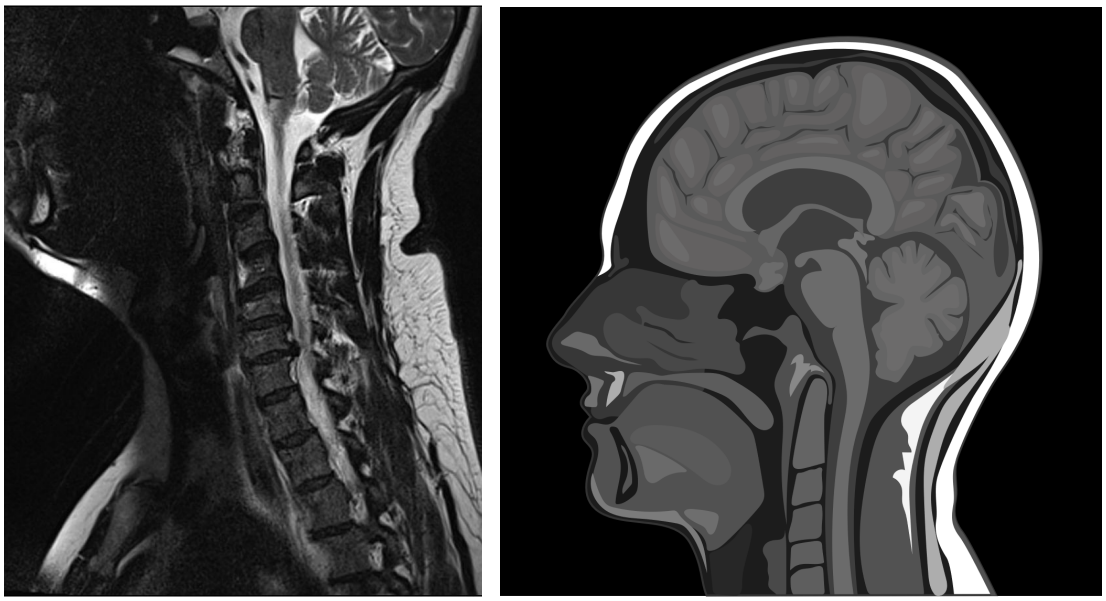


(a) Chest X-ray. (b) Hand X-ray.

Figure 2.3: X-ray Images.

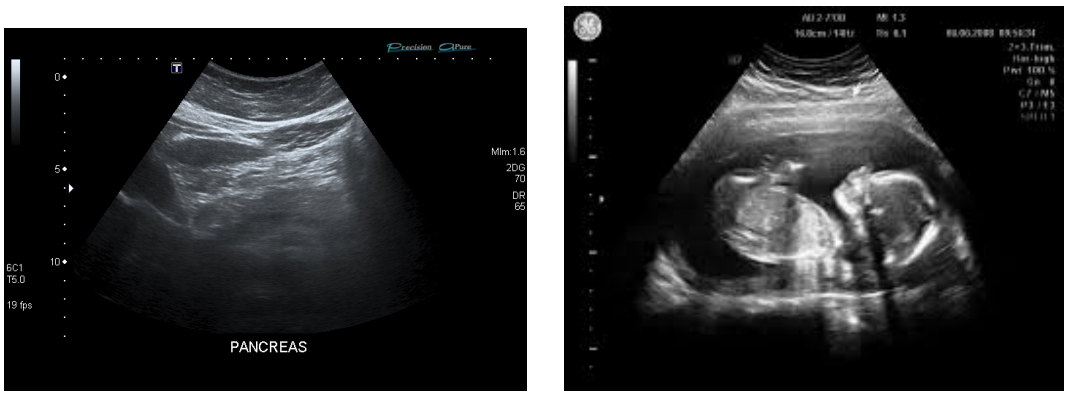
(a) CT scan of a pregnancy. (b) CT showing spine and kidneys.

Figure 2.4: CT scan Images.



(a) C6-C7 disc herniation MRI. (b) MRI of the brain.

Figure 2.5: MRI scan Images.



(a) Pancreas ultrasound. (b) Pregnancy ultrasound.

Figure 2.6: Ultrasound Images.

- **Pathology:** (Histology Slides) Grayscale imaging is used to examine tissue samples under a microscope, highlighting cellular structures and abnormalities that can indicate diseases such as cancer (Figure 2.7).

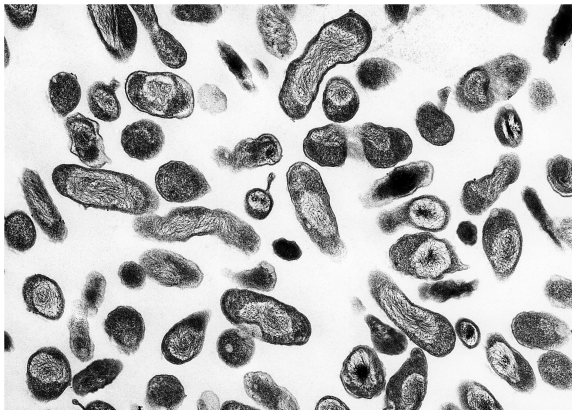


Figure 2.7: Histology Slides

2. **Military Field :** Grayscale images can be used in the military field, especially in Reconnaissance and Surveillance and targeting applications:

   - **Satellite and Aerial Imagery:** Grayscale images from satellites and aerial drones are used for detailed terrain analysis, identifying structures, and monitoring changes in landscapes, and identifying objects or targets in various lighting conditions. These images can penetrate certain weather conditions better than color images (Figure 2.8).
   - **Night Vision:** Grayscale images are fundamental in night vision equipment, where the intensity of infrared light is converted into shades of gray, allowing soldiers to see in low-light conditions (Figures 2.9a, 2.9b).
   - **Target Recognition:** Grayscale images are used in various targeting systems to identify and track objects based on their shapes and contrasts rather than colors, which can be misleading in different lighting conditions or camouflage situations.
   - **Thermal Imaging:** Grayscale thermal images detect heat emitted by objects and individuals, useful for locating hidden targets, monitoring equipment, and assessing damage in military operations (Figure 2.10).

Figure 2.8: Aerial Image

(a) Night Vision (Gazelle). (b) Night Vision (Mountain lion).

Figure 2.9: Night Vision Images.

Figure 2.10: Thermal Imaging

3. **Other Fields:**

   - **Astronomy:** Many astronomical images are captured in grayscale, particularly those from telescopes that detect non-visible wavelengths (e.g., radio, X-ray). These images are essential for studying celestial objects and phenomena, such as black holes, galaxies, and star formations (Figure 2.11).
   - **Microscopy:** Grayscale images are used in different types of microscopy (electron, optical) to study the fine details of materials, biological specimens, and nanostructures, enabling researchers to observe phenomena at a microscopic level.
   - **Security and Surveillance:** Grayscale images from CCTV cameras are widely used for monitoring and security purposes. They offer sufficient detail for identifying movements and activities while reducing data storage requirements compared to color images.
   - **OCR (Optical Character Recognition):** Converting documents into grayscale simplifies the process of recognizing and digitizing text from printed materials, improving accuracy and reducing data storage requirements.
   - **Machine Vision and Robotics:** (Pattern Recognition) Grayscale im-

Figure 2.11: Astronomy photography

ages are often used in machine vision systems for object detection, quality control, and robotic guidance, as they reduce computational complexity while retaining essential information.

- **Art and Design:** Grayscale images are used in various artistic and design applications to focus on composition, light, and shadow without the influence of color. This technique is often used in preliminary sketches and studies.

2.9 Conclusion

In the previous chapter, we delved deeply into the realm of digital images. We thoroughly examined various aspects, starting with the different classifications and types of digital images. We also explored the various formats available, discussing their unique characteristics and the contexts in which they are most effectively used. Additionally, we analyzed the essential properties of digital images, such as resolution, color depth, and compression, providing a comprehensive understanding of their technical foundations and practical applications.

Chapter 3

Cryptography and Image encryption

The digital revolution has fundamentally transformed how we create, transmit, and store information. Images, once confined to physical mediums, are now ubiquitous in our digital world. From personal photos shared online to medical scans transmitted for diagnosis, the secure handling of visual data has become paramount. This chapter delves into the realm of cryptography, specifically focusing on its application in image encryption.

3.1 Cryptography

In the current era, during the worldwide digital revolution, when data is continuously exchanged across the Internet, ensuring information protection has become a concern for both individuals and institutions. Cryptography has emerged as a dependable technique to ensure the confidentiality, integrity, and privacy of data in the face of increasing risks. It protects its security from curious individuals and cyber criminals. in this section of the thesis, we will discuss cryptography and its principles,

Protecting sensitive and private information is necessary, especially during its exchange and storage. Therefore, we can divide information security into two important parts (Figure 3.1):

- A section that deals with hiding information within other information, This section can be divided into two branches:
  - **Steganography:** Steganography is a technique used to conceal secret information within a non-secret message, image, or other medium. It is a form of covert communication, where the goal is to hide the existence of the secret message itself.
  - **Watermarking:** Digital watermarking is a technique used to embed information into digital media (such as images, audio, or video) in a way that is imperceptible to the human senses but can be detected or

extracted by a computer. Unlike steganography, which focuses on hiding the existence of a message, watermarking is often used to protect content ownership and verify authenticity.

- Second section that focuses on protecting information by camouflaging it and transforming it into an incomprehensible and unreadable form. This section is called Cryptography.

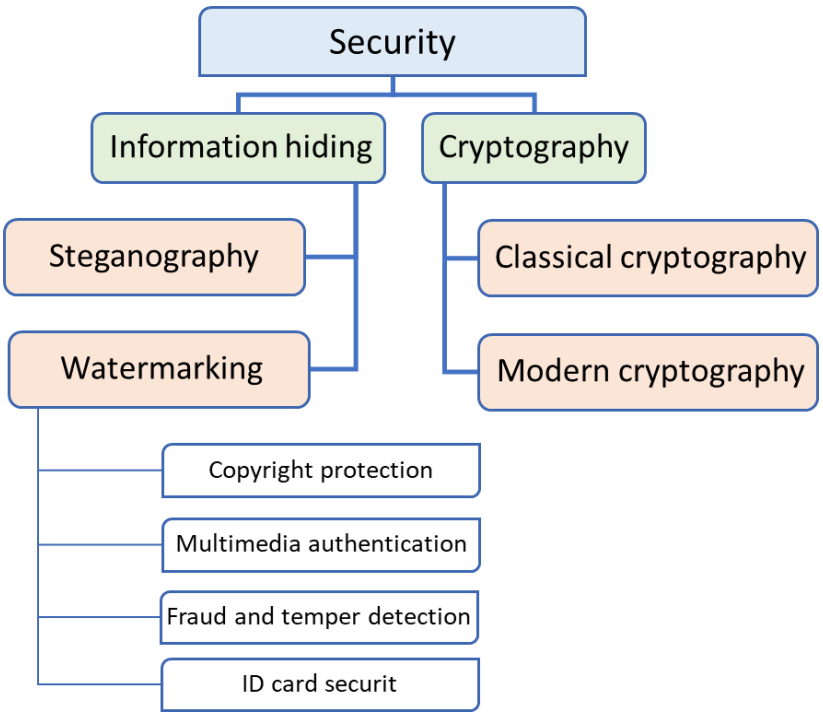


Figure 3.1: Security categories [77, 32]

3.1.1 Definition

Cryptography is the discipline that involves the techniques and principles used to ensure the confidentiality and integrity of information by transforming it into a coded form that can only be understood by authorized individuals. It involves developing and examining strategies to protect private communication from being intercepted or understood by unauthorized parties or individuals. Thus, cryptography is the art and science of keeping information hidden and secure by encoding and decoding messages to prevent unauthorized access.

3.1.2 Goals of Cryptography

Cryptography is an advanced technique developed to accomplish many essential objectives, guaranteeing secure transmission and safeguarding of information in an increasingly digital environment. The notion includes several crucial objectives, such as maintaining confidentiality, ensuring integrity, verifying authentication, preventing repudiation, and guaranteeing unaltered delivery. The objectives can be succinctly expressed in the following manner:

1. **Confidentiality:** One purpose of cryptography is to safeguard the privacy of information (called Confidentiality). Confidentiality ensures that the information contained in a communication is not understandable to unauthorized individuals and can only be accessible by authorized receivers, thereby preventing unlawful access or disclosure.

2. **Integrity:** Cryptography ensures data integrity by prohibiting unauthorized alterations during transmission and storage. This allows the receiver to verify that the message they got is identical to the one sent by the sender. Any data modifications will be identifiable, avoiding unwanted tampering or manipulation. This acts as a deterrent against unauthorized modifications carried out by malicious individuals. The Integrity examines if data remains unchanged during its transit or storage.

3. **Authentication:** Cryptography offers techniques for confirming the identity of the source or recipient of data. Authentication means the verification of the identity of the sender of the communication, allowing the recipient to authenticate that the message indeed came from the claimed sender. This is essential to prevent impersonation, verify the message's origin from a reliable source, and establish confidence in online interactions.

4. **Non-repudiation:** Digital signatures and other cryptographic processes make it impossible for a message's sender to repudiate sending it, which ensures accountability. This is known as non-repudiation, and it ensures that the sender cannot subsequently deny sending a message since it stops the sender from disavowing a previously transmitted message. This stops senders from taking back their acts, which is vital in situations where responsibility is required.

3.1.3 Classification of cryptography

Cryptography has been considered one of the most essential technologies produced by humans in their constant desire to safeguard their messages and private infor-

mation throughout history. Therefore, many types of techniques used in encryption have been invented over time, up to the present, and thus these techniques can be divided into two basic parts (Figure 3.2):

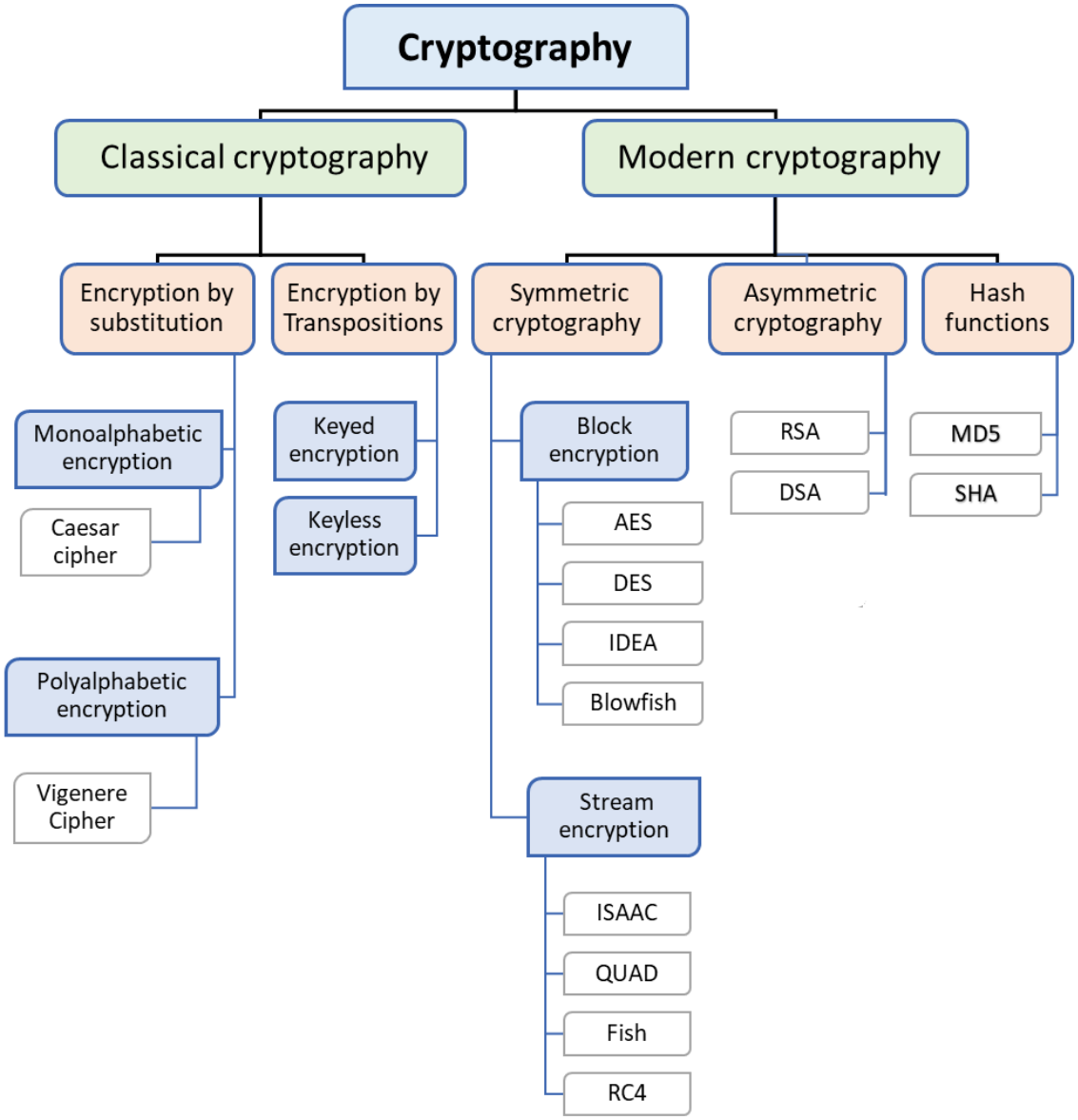


Figure 3.2: Classification of Cryptography

3.1.3.1 Classical cryptography

Classical cryptography, developed before the 20th century, is a traditional method of encryption and decryption that involves manual cryptographic algorithms and relies on manual techniques or simple mechanical tools. It is characterized by reliance on

encryption algorithm secrecy, limited key space, susceptibility to brute-force attacks, and lack of mathematical underpinnings and formal security analysis.

It can be divided into two categories:

1. **Transposition cryptography**, which alters the order of characters in plaintext to create a ciphertext, including Rail Fence,Row Transposition, Columnar Transposition, and Scytale.

2. **Substitution cryptography**, which replaces characters in plaintext to create a ciphertext.including Caesar, Monoalphabetic, and Polyalphabetic Substitution Ciphers,

3.1.3.2 Modern cryptography

Modern cryptography, on the other hand, refers to the encryption techniques and methods that have been developed and used since the 20th century, especially after the development of digital computers. This Cryptography can be classified into three primary categories:

1. **Symmetric key cryptography**, often referred to as private key cryptography, secret key cryptography, or single-key encryption, employs a solitary shared key for both the encryption and decryption processes (Figure 3.3). This implies that the same key is used to both encrypt and decrypt the message, resulting in the message being scrambled (encrypted) and unscrambled (decrypted) at the receiving end. Some examples of encryption algorithms are AES (Advanced Encryption Standard), DES (Data Encryption Standard), and Blowfish.

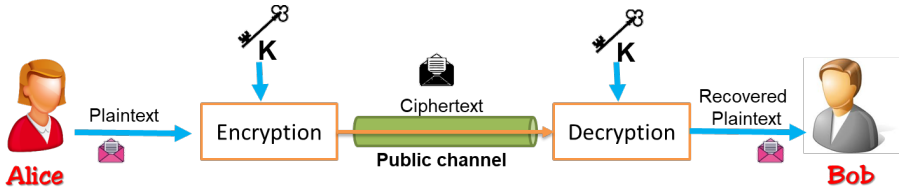


Figure 3.3: Symmetric key cryptography

2. **Asymmetric Key Cryptography** (also known as public key cryptography or two-key cryptography) uses a pair of mathematically linked keys (Figure 3.4): a public key and a private key. While the public key is freely available, the private key remains secret. Only the corresponding private key can decrypt data encrypted with the public key, and vice versa. Examples

include RSA (Rivest-Shamir-Adleman), ECC (Elliptic Curve Cryptography), and DSA (Digital Signature Algorithm).

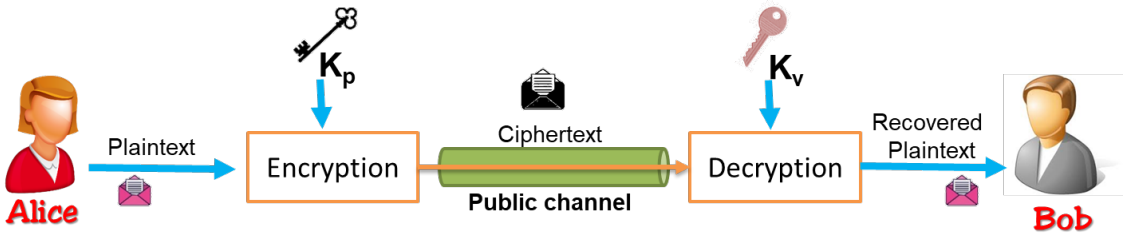


Figure 3.4: Asymmetric key cryptography

3. **Hash functions** are one-way mathematical functions that take any size input and generate a fixed-size output called a hash value (Figure 3.5). Hash functions play a crucial role in maintaining data integrity and confirming the absence of tampering during transmission or storage. Consider a hash function as a digital fingerprinting tool; it generates a distinct "fingerprint" (hash) for a specific piece of data, enabling subsequent authentication verification. Examples of hash functions include SHA-256 (Secure Hash Algorithm) and MD5 (Message Digest Algorithm).

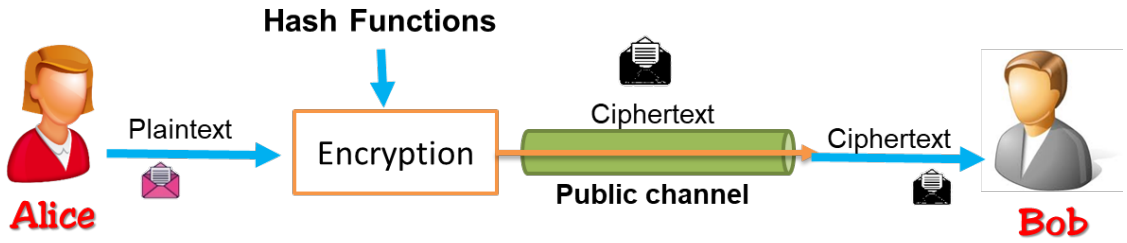


Figure 3.5: Hash Function

3.1.4 Cryptanalysis

cryptanalysis is a discipline that detects vulnerabilities in Cryptography systems and exploits them to breach encryption. This requires comprehension of the system's algorithms, protocols, and implementations to detect flaws that could be exploited to impersonate users, recover plaintext, or forge signatures. Cryptanalysis is the study of methods for breaking cryptographic algorithms . It involves analyzing and exploiting weaknesses in encryption algorithms, key management, and other

cryptographic systems to gain unauthorized access to sensitive information.The investigation of cryptographic systems facilitates comprehension and exploitation of these vulnerabilities, thereby assuring the confidentiality of sensitive data.

Attackers are always seeking opportunities to take advantage of weaknesses in cryptographic systems. The tools and methods of attack vary according to the variety of aims and knowledge about the targeted system. Hence, several types of cryptographic attacks may be used to circumvent and undermine the security of the cryptosystem. Presented here are many of the most common types.

3.1.5 Types of cryptanalysis attacks

Cryptanalysis includes a variety of techniques, which are essentially classed as passive and active attacks.

3.1.5.1 Passive cryptanalys

Passive cryptanalys is a technique in which an attacker passively monitors and analyzes encrypted communication without actively interfering. The attacker aims to decrypt the plaintext message or gain valuable information about the cryptographic system without altering the transmitted data. Passive attacks rely on the cryptanalyst's ability to detect patterns, statistical anomalies, or other weaknesses in the ciphertext or the cryptographic algorithm itself. Passive attacks provide a challenge in terms of detection and may be executed over a long duration, as the target stays oblivious to the eavesdropping while the attackers persistently collect information. There are several examples of passive attacks, including: Traffic analysis, Ciphertext-only Attack, Known-plaintext Attack, Chosen-plaintext Attack.

3.1.5.2 Active cryptanalys

Active attacks entail interacting with the cryptographic system in order to compromise it. The attacker seeks to exploit vulnerabilities in the system's implementation or underlying algorithm. The purpose is to extract information or harm the system. This may include changing, injecting, or deleting messages in order to control the system and disclose vulnerabilities. Active attacks usually demand more resources, capabilities, and system access than passive attacks since the attacker must interact with the system directly. They can, however, be more effective in cracking cryptographic systems, particularly if the system contains implementation-level weaknesses. Active assaults are easier to detect than passive attacks since they are disruptive. However, they can inflict enormous harm by interfering with commu-

nication, stealing important information, or altering system behavior. There are several examples of Active attacks, including: Man-in-the-Middle Attack, Replay Attack,Chosen-ciphertext Attack, Denial-of-service attack.

Attackers are always seeking opportunities to take advantage of weaknesses in cryptographic systems. The tools and methods of attack vary according to the variety of aims and knowledge about the targeted system. Hence, several types of cryptographic attacks may be used to circumvent and undermine the security of the cryptosystem. Presented here are many of the most common types:

3.1.5.3 Brute-force attacks

Brute-force attacks involve trying every possible key until the right and proper one is discovered. This technique can be effective but is computationally intensive and may not be feasible for very long keys. It is effective against weak or short encryption keys but becomes exponentially more difficult as the key length increases.

3.1.5.4 Ciphertext-only Attack

The attacker obtains a set of encrypted ciphertext. The goal is to find the key or plaintext only based on the ciphertext. Implementing this attack is highly challenging, yet it is commonly used because it only requires the ciphertext. Although the attacker does not have direct access to the plaintext, they can nevertheless deduce the ciphertext from the dataset.

3.1.5.5 Known plaintext attack

A known plaintext attack is a method where an attacker gains access to both the ciphertext and the associated plaintext to deduce the encryption key used for the transformation. This method entails identifying or acquiring the original text of some sections of the encrypted message by employing information-collecting techniques, such as linear cryptanalysis in block ciphers. These attacks are more feasible to execute than ciphertext-only attacks since there is more accessible information.

3.1.5.6 Chosen plaintext attacks

Chosen plaintext attacks entail the deliberate selection of a plaintext and the subsequent observation of its associated ciphertext. This information can then be utilized to determine the secret key or system specifics. This attack enables the attacker to get insights into the encryption process and take advantage of vulnerabilities in the

algorithm.An illustrious instance of this sort of attack is the differential cryptanalysis executed on block ciphers.

3.1.5.7 Chosen ciphertext attack

in contrast with the chosen plaintext attack in that the attacker picks the ciphertext and obtains the associated plaintext. The chosen ciphertext attack allows the attacker to determine the encryption key by choosing the ciphertext and obtaining the corresponding plaintext.

3.1.5.8 Man-in-the-middle attacks

Man-in-the-middle attacks refer to malicious activity in which an attacker intercepts and modifies communication between two parties. The attacker possesses the ability to read, alter, or inject messages, inducing the parties to harbour the false belief that they are engaging in direct communication. This attack involves intercepting communication between two parties and impersonating one of them to steal cryptographic keys or other sensitive information.

3.2 Image encryption

In the digital age, where visual data permeates every aspect of our lives, Image encryption is a crucial tool, protecting sensitive information such as personal photos, medical scans, financial transactions, and military reconnaissance imagery. Unlike traditional text encryption, which manipulates characters, image encryption addresses pixels and their intricate correlation within an image. This chapter explores prevalent image encryption methodologies.

3.2.1 Challenges and considerations

Image encryption techniques differ from text encryption techniques due to the unique characteristics of image data (see table 3.1). These differences pose unique challenges and issues that need to be considered and must be addressed when designing image encryption algorithms. in this section, some key differences and considerations are listed:

1. **Data characteristics:**

    - **Multi-dimensional data:** Unlike text, which is typically 1-dimensional, image data is 2-dimensional (or even 3-dimensional in case of multi-spectral images), adding complexity to the encryption process. This structure must be considered when designing encryption algorithms.
    - **Pixel values:** Images are represented as pixel values, which can range from 0 to 255 in grayscale images or include multiple channels (e.g., RGB) in color images (Images may have multiple color depths). This differs significantly from text, which is typically has a binary representation and represented as ASCII or Unicode characters.
    - **Strong Correlation:** Neighboring pixels in an image are often correlated, which can expose patterns and weaken encryption.
    - **Large size:** Images typically consist of much larger data compared to text (see Table 2.2), requiring more computational power and memory for encryption and decryption processes.
    - **high redundancy:** Image data is highly redundant because of the strong correlations between neighboring pixels. This means that a lot of the information in an image can be predicted from the information already known. For example, if you know the color of one pixel in a smooth

gradient, you can likely guess the colors of the pixels around it. This redundancy makes image data more susceptible to certain types of attacks, especially statistical attacks that try to exploit these patterns.

- **Data Distribution:** Image data tends to have a more continuous distribution compared to the discrete distribution of text data.

2. **Data compression:**

   - **Image compression :** Unlike text data, where text compression is less important for security purposes, as text files are generally smaller and can be easily compressed after encryption. Image data often undergoes compression before encryption since image compression is essential. Image compression techniques, like JPEG and PNG, are widely used to reduce file sizes.Encryption techniques must be compatible with these compression standards.
   - **Lossless vs. lossy compression :** Images are often compressed using lossy or lossless compression techniques. which can be challenging given the different characteristics of these methods. Compression algorithms can introduce specific patterns, which can be exploited by attackers. Encrypted images should ideally be compatible with these compression methods without compromising security or compression efficiency.
   - **Size Increase:** Encryption can increase the size of image files, especially if they interfere with the compression algorithms, which is a lesser concern for text.

3. **Quality preservation**

   - **Visual quality:** Image quality is crucial, it does not tolerate any distortions during encoding, but text is more tolerant; text encryption can tolerate some distortion without compromising readability. For instance, a few misplaced characters in a text message might still be understandable. Hence, Preserving image quality is essential, especially for applications requiring high visual fidelity.
   - **Sensitivity to distortion** Images are often more sensitive to errors and distortion than text. Even minor errors in the decryption process can lead to noticeable visual artifacts. Any degradation or artifact introduced during encryption can significantly impact results and usability. Image encryption must ensure that the decrypted image is of high quality and closely matches the original.

4. **Real-Time requirements**

   - **Latency:** Latency refers to the delay between the initiation and completion of a process. In the context of image encryption and decryption, many applications, such as video streaming and real-time surveillance, require these processes to be performed with minimal latency to happen almost instantaneously.
   - **Throughput:** Throughput refers to the amount of data processed in a given amount of time. Handling large volumes of image data in real-time scenarios demands high throughput, which is less of an issue with text data. Real-time applications involving images require encryption and decryption processes to handle large volumes of data quickly.

5. **Hardware and software constraints**

   - **Hardware acceleration:** Image encryption may benefit from specialized hardware like GPUs or FPGAs to manage the large data size and real-time requirements. Text encryption generally requires less specialized hardware.
   - **Implementation complexity:** Implementing efficient and secure image encryption algorithms can be more complex due to the need to handle large data sizes and ensure visual quality.
   - **Resource constraints:** Devices with limited computational power and storage (e.g., IoT devices, mobile phones) pose additional challenges for implementing efficient image encryption.

6. **Security concerns**

   - **Attack vectors:** Images can be vulnerable to specific attacks such as chosen-plaintext and known-plaintext attacks, where attackers use the inherent redundancy and patterns in images to break encryption.
   - **Statistical attacks:** The high redundancy in image data can be exploited in statistical attacks, which are less effective against the more random nature of text data.
   - **Perceptual attacks:** Attackers may use visual perception characteristics to infer information about the encrypted image, which can be challenging to counter.

Given these distinctions and various other factors unique to images compared to text, it becomes imperative to explore the development of dedicated encryption techniques tailored specifically for images.

Aspect	**Image Data**	**Text Data**
Data Size	Generally large, often in megabytes or more	Typically smaller, often in kilobytes or less
Data Structure	Multi-dimensional arrays (pixels)	Linear sequence of characters
Data Representation	Pixel values (e.g., RGB values, grayscale)	ASCII or Unicode characters
Redundancy	High redundancy, spatial correlation between pixels	Lower redundancy, less correlation between characters
Compression	Often compressed before encryption (e.g., JPEG, PNG)	Can be compressed after encryption, if necessary
Perceptual Quality	Quality can degrade with compression or encryption	No perceptual quality concerns, characters are discrete
Sensitivity to Errors	More sensitive, errors can affect visual quality	Less sensitive, minor errors may not affect readability
File Formats	Multiple formats (JPEG, PNG, BMP, etc.)	Fewer formats (TXT, DOC, PDF, etc.)
Processing Requirements	Higher computational and memory requirements	Lower computational and memory requirements
Key Management	More complex due to larger data size and structure	Simpler due to smaller and more uniform data
Security Concerns	High sensitivity to chosen-plaintext attacks	Standard security practices generally sufficient
Visual Artifacts	Possible during encryption/decryption	Not applicable
Storage and Transmission	Requires more storage and bandwidth	Requires less storage and bandwidth
Application-Specific Needs	Critical in fields like medical imaging and surveillance	Less critical, mostly standard across applications

Table 3.1: The key differences between image data and text data

3.3 Image encryption methods

In the literature, a multitude of algorithms are specifically designed for encrypting images, addressing the considerations and challenges outlined in the preceding section. These algorithms can be categorized into several classifications (Figure 3.7), including:

- Classifications by Domain.
- Classifications by Key Types.
- Classifications by Techniques.

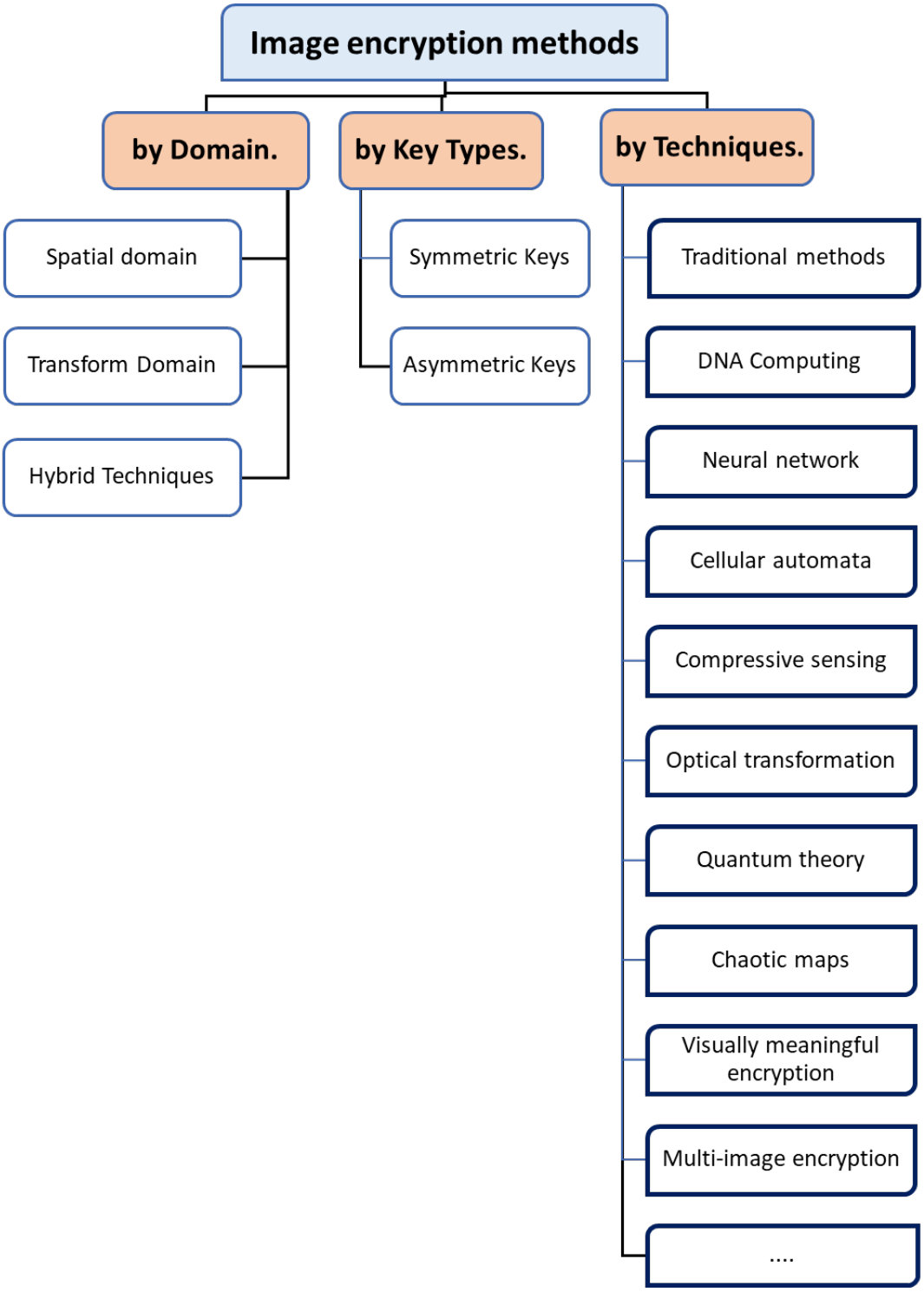


Figure 3.6: Classification of Image encryption methods

3.3.1 Classifications by Domain

This categorization focuses on the domain in which the encryption operations occur. there are two main categories (Table 3.2):

Domain	Techniques	Advantages	Disadvantages
Spatial	Permutation, Substitution	Simple to implement, Fast processing	Vulnerable to statistical attacks, Error propagation
Frequency	Transform image to the frequency domain, Manipulate frequency components	Better resists statistical attacks	Computationally expensive, Potential for distortions
Hybrid	Combines spatial and frequency techniques	Potentially stronger security	Increased complexity, Computational cost

Table 3.2: Types of Image encryption techniques by Domain

3.3.2 Spatial domain

Spatial domain encryption techniques work directly on an image's pixel values and positions within its raw pixel grid, bypassing the need for transformation into other domains. These methods achieve encryption by altering pixel values and rearranging their positions, such as through pixel scrambling (permutation) or intensity modification (substitution). Common approaches include chaos-based, cellular automata-based, DNA-based, and metaheuristic-based methods, all of which directly manipulate the image's pixel values.

There are three types of spatial domain image encryption techniques:

1. Position Permutation (Transposition) Based Algorithm.
2. Value Transformation (diffusion) Based Algorithm.
3. Position-Substitution Based Algorithm.

3.3.2.1 Position Permutation Based Algorithm

Position Permutation-Based Algorithms or Transposition algorithms can rearrange image elements like bits, pixels, or blocks. They use a permutation or keying sequence to shuffle these elements' positions. The process involves generating or obtaining a permutation key or sequence, which may be a pseudorandom sequence or derived from a user-provided key. This key or sequence dictates the swapping or transposition of element positions, ensuring effective image manipulation.

There are three levels of rearranging elements:

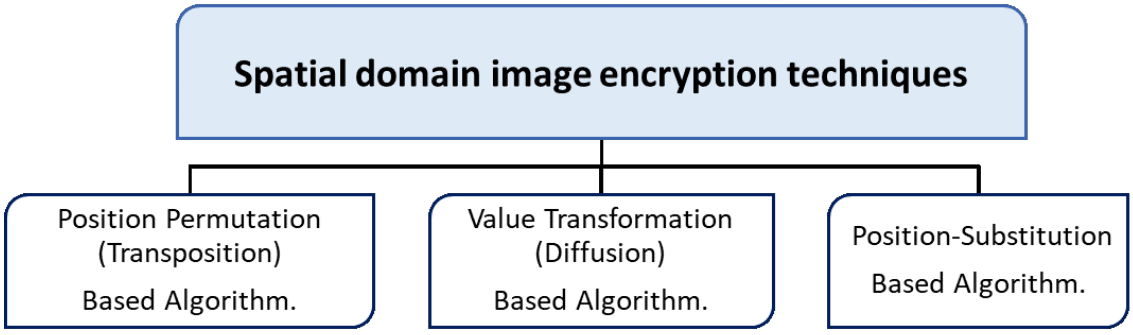


Figure 3.7: Spatial domain image encryption techniques

1. **Bit-level permutation:** This involves shuffling the individual bits within each pixel. While it might reduce some visual information,
2. **Pixel-level permutation:** This involves rearranging entire pixels within the image.
3. **Block-level permutation:** This involves dividing the image into blocks of pixels and then shuffling those blocks .

3.3.2.2 Value Transformation Based Algorithm

The value transformation algorithm, also known as diffusion, is a technique that complements the position permutation-based algorithm in image encryption. Unlike the position permutation-based algorithm, which rearranges the positions of image elements (bits, pixels, or blocks), the value transformation (diffusion) algorithm obscures image data by altering the values of these elements. This algorithm generates or uses a diffusion key or sequence similar to the permutation key in the position permutation-based approach.

- The image is segmented into parts (bits, pixels, or blocks) based on the chosen diffusion level (bit-level, pixel-level, or block-level).
- Each element undergoes a diffusion operation using the diffusion key or sequence. This operation can involve bitwise operations (such as XOR, rotation, or substitution) or mathematical operations (such as modular addition or multiplication).

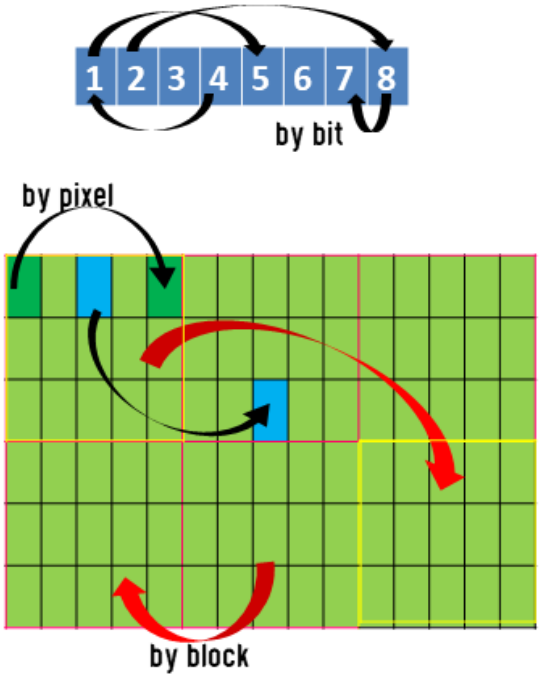


Figure 3.8: Confusion methods

- The diffused elements are then combined to create the encrypted image.

The diffusion technique increases the complexity and dispersion of the image data, making it more difficult to determine the original values of the elements or their relationships. An effective diffusion algorithm should ensure that even a slight change in the input, whether the original image or the key, results in a significant impact on the output.

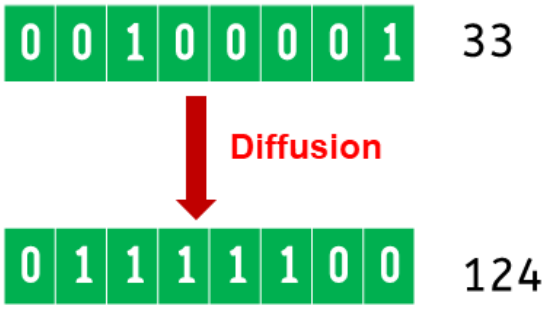


Figure 3.9: Diffusion method

3.3.2.3 Position- Substitution Based Algorithm

In image encryption, a position-substitution-based algorithm combines position permutations and value transformations. This method permutes and substitutes the

positions and values of visual elements such as bits, pixels, or blocks. The pixels are rearranged, and their values are replaced using a key generator.

The image is divided into components (bits, pixels, or blocks) based on the chosen substitution level (bit-level, pixel-level, or block-level).

Each element undergoes two operations:

1. Position permutation: The element's position is rearranged or shuffled based on a permutation sequence, similar to the position permutation algorithm.

2. Value substitution: The substitution key or sequence determines the new value for the element, involving either mathematical operations (such as modular addition or multiplication) or bitwise operations (such as XOR or substitution boxes).

The permuted and substituted elements are then combined to create the encrypted image.

This combined approach of substitution and permutation provides a higher security level than using either method alone.

3.3.3 Transform Domain

In the spatial domain, Each pixel in the image represents a specific location with a certain brightness or color value. This domain provides a direct, point-to-point representation of the visual information.

The frequency domain, on the other hand, offers a more abstract view. Here, the image is decomposed into its fundamental building blocks: frequencies. Just like musical notes can be combined to create complex sounds, various frequencies combine to form the image you see. Each frequency has two key characteristics:

1. **Amplitude:** This represents the strength or intensity of a particular frequency component. High amplitudes in the frequency domain correspond to sharp details or pronounced features in the image (like edges or noise). Conversely, low amplitudes indicate smoother, less defined areas, it represent smooth variations in the image.

2. **Phase Shift:** This describes the relative position of a frequency component within the wave cycle. While it doesn't directly affect brightness, the phase relationships between frequencies influence how they combine to create the final image.

The attraction of frequency domain analysis lies in its connection to the spatial domain. Any alteration applied to the image in the spatial domain, (e.g., blurring edges, sharpening details), would indirectly result in a change in the distribution of frequencies in the frequency domain to reflect the altered image characteristics. Likewise, altering the frequency components (amplitude or phase) will have a corresponding effect on the visual representation of the image when transformed back to the spatial domain. Techniques like noise reduction or image sharpening often involve targeted adjustments to specific frequency bands.

Within the frequency domain, different frequency ranges carry distinct information:

- **High frequencies:** These frequencies hold information about the image's fine details, such as sharp edges, textures, and high-frequency noise. Higher frequencies correspond to finer and more rapid changes in intensity within the image. As the frequency increases, the details become finer and more intricate.
- **Low frequencies:** Conversely, low frequencies represent the broad, smooth variations in intensity across the image. They capture the overall image structure and low-frequency details like gradual changes in brightness or background colors.

Transforming an image to the frequency domain provides valuable insights into its composition. It allows us to:

- **Separate image content from noise:** By analyzing the frequency distribution, we can identify and target noise components for removal while preserving the underlying image information.
- **Perform selective editing:** We can manipulate specific frequency bands to achieve targeted effects. For example, enhancing edges or textures by boosting certain high frequencies or smoothing the image by reducing them.
- **Compress images efficiently:** By discarding less important high-frequency information that may not be visually perceptible, we can achieve significant image compression without sacrificing too much detail.

Transform domain techniques in image encryption involve converting the image into a different domain, typically the frequency domain, using transforms like Discrete Fourier Transform (DFT) or Discrete Cosine Transform (DCT), before applying encryption, before applying encryption. This transformation aims to disrupt

spatial correlations and patterns within the image, making it more challenging for attackers to exploit them.

Transforming the image from the spatial domain to the frequency domain involves representing the image in terms of its constituent frequencies, similar to decomposing a musical piece into individual notes. Encryption operations are then performed on the image's frequency components, manipulating these frequencies to obscure the original image content. Transform domain techniques can provide higher security and resilience against attacks compared to spatial domain techniques.

The following are some prevalent transform techniques employed in the image encryption by the frequency domain :

3.3.3.1 Discrete Fourier Transform (DFT)

This fundamental mathematical operation decomposes an image into its basic sinusoidal components, revealing the presence of various frequencies. Each component is represented by its amplitude and phase, providing a complete picture of the image's frequency content. In encryption, manipulations like modifying the coefficients or introducing phase shifts can disrupt the image's visual information.

3.3.3.2 Discrete Cosine Transform (DCT)

Similar to DFT, DCT decomposes an image into its frequency components. However, DCT exhibits better energy compaction properties, concentrating most of the image's information in the lower frequency bands. This characteristic makes DCT particularly useful for natural images, where significant details reside in the low-frequency components. By selectively manipulating these crucial bands, encryption can target specific image features.

3.3.3.3 Wavelet Transform (WT)

Unlike DFT and DCT, which offer global frequency analysis, WT provides both spatial and frequency information. It decomposes the image into wavelet coefficients, capturing spatial details at different frequency scales. This ability to localize frequency components makes WT valuable for encryption scenarios where spatial information preservation is crucial. Targeted encryption can be achieved by strategically modifying wavelet coefficients in specific locations and frequencies.

The following table 3.3 describes a set of Transform techniques and their applications

Transform Technique	**Description**	**Applications**
Discrete Fourier Transform (DFT)	Decomposes a discrete signal into its fundamental frequency components.	* Signal analysis (e.g., audio processing, vibration analysis) * Image processing (e.g., image compression, frequency domain filtering) * Telecommunications (e.g., spectral analysis for data transmission)
Discrete Cosine Transform (DCT)	Similar to DFT, focuses on cosine components for efficient signal representation.	* Image and video compression (e.g., JPEG, MPEG) * Signal coding (e.g., audio compression)
Wavelet Transform (WT)	Provides both spatial and frequency information by decomposing signals into wavelet coefficients.	* Image compression with good spatial localization (e.g., JPEG 2000) * Signal denoising (e.g., removing noise from audio or images) * Feature extraction in image and signal processing
Short-Time Fourier Transform (STFT)	Analyzes short segments of a signal to capture time-varying frequencies.	* Speech processing (e.g., analyzing speech formants) * Music analysis (e.g., identifying musical notes) * Time-frequency analysis of non-stationary signals
Fast Fourier Transform (FFT)	Efficient algorithm for computing the DFT, making large-scale signal processing feasible.	* All applications that utilize the DFT, but with significantly reduced computational cost.
Laplace Transform	Converts a function from the time domain to the Laplace domain (complex s-plane).	* Analysis of linear systems with constant coefficients (e.g., electrical circuits, control systems) * Solving differential equations
Z-Transform	Transforms discrete-time signals into the z-domain, enabling analysis of digital filters and systems.	* Analysis and design of digital filters (e.g., low-pass filters, high-pass filters) * Stability analysis of digital systems
Haar Transform	Simple orthogonal transform with efficient implementation, useful for specific applications.	* Image compression (e.g., early image compression methods) * Edge detection in images
Sine Cosine Chebyshev Transform (SCCT)	Offers properties between DFT and DCT. Utilizes Chebyshev polynomials for alternative frequency domain analysis.	* Emerging application in image encryption due to potential for improved security.
Karhunen-Loève Transform (KLT)	Data-driven transform that maximizes energy compaction based on the specific signal statistics.	* Principal component analysis (PCA) for dimensionality reduction * Image compression
Walsh-Hadamard Transform (WHT)	Fast and efficient transform with applications in coding and cryptography.	* Error correction in coding schemes * Stream cipher design in cryptography
Hilbert-Huang Transform (HHT)	Analyzes non-stationary and nonlinear signals by decomposing them into Intrinsic Mode Functions (IMFs).	* Analysis of biomedical signals (e.g., EEG) * Oceanographic and atmospheric data analysis

Table 3.3: Transform Techniques and their Applications

3.3.4 Classifications by techniques

Encrypting images involves various approaches and methods. Thus, image encryption methods can be classified based on the technique operated. In this part, we outline the key techniques used, which can be combined in several ways to achieve different levels of security and efficiency.

3.3.4.1 Traditional encryption methods

Despite the large difference between image and text data, many image encryption methods use traditional encryption methods, such as AES (Advanced Encryption Standard), DES (Data Encryption Standard), and Blowfish. Due to potential shortcomings in their effectiveness, hybrid approaches are used in conjunction with other strategies to improve performance.

3.3.4.2 DNA computing

DNA computing is a field that uses DNA molecules to perform computational tasks, such as image encryption. It uses biological principles and operations of DNA sequences, such as DNA encoding, DNA sequence operations, and Genetic Adversarial Networks (GAN)[51, 6, 106]. The process involves mapping image pixels to DNA nucleotides (A, T, C, and G) and applying biological operations like crossover, mutation, and splicing. DNA operations such as DNA hybridization and strand displacement can be encrypted with high complexity and security, despite the complexity of implementation and processing.

3.3.4.3 Cellular automata

Cellular automata (CA) is a data security method that uses the principles of cellular automata, a computational model consisting of a grid of cells with a finite number of states (e.g., 0 and 1). A set of rules determines each cell's next state based on the states of its neighboring cells [46]. This process is repeated for each cell in the grid, creating a dynamic pattern of states over time. This method leverages cellular automata's complex, chaotic behavior to create cryptographic systems. CA-based encryption offers several advantages, including the ability to generate large and complex keys that are difficult to predict or break, as well as the ease of implementation using parallel or distributed computing, which improves the speed and efficiency of encryption and decryption.

3.3.4.4 Compressive sensing

Compressive sensing-based image encryption is a new method that combines sampling, compression, and encryption. It captures the sparse representation of images by sampling a subset of their measurements. In the encryption process, the image is compressed using a sensing matrix, and encryption algorithms such as chaotic maps or cryptographic techniques are applied to the compressed measurements [12]. The decryption process reconstructs the original image from the encrypted measurements. Compressive sensing is useful for encrypting images while reducing storage size, allowing for efficient storage and transmission while maintaining security. However, it requires specialized algorithms that can be computationally intensive, especially for high-resolution images.

3.3.4.5 Optical transformation

Optical transformation-based image encryption is a method that uses optical principles to secure digital images, providing speed and security for high-speed applications. This method manipulates light patterns within the image rather than relying solely on computational methods, passing light through optical elements like lenses and mirrors as encryption keys. The decryption process reverses these transformations to recover the original image. Various optical techniques, like Fourier transform and double random phase encoding, are employed for encryption. The decryption process involves applying inverse optical transformations [41]. Due to complex light interactions, optical encryption operates at the speed of light, facilitating real-time encryption and offering high security. It is suitable for demanding applications and can be implemented in hardware using specialized components.

3.3.4.6 Neural network

Traditional encryption algorithms rely on well-defined mathematical operations and pre-defined keys. Neural network-based encryption, on the other hand, is a new type of encryption method that uses artificial neural networks to learn complex encryption patterns [53] . These networks are trained on a dataset of images and their corresponding encrypted versions, learning the relationships between the original and encrypted images. The trained network can encrypt new images by applying the learned patterns and decrypt encrypted images by reversing these patterns. The network itself becomes an integral part of the encryption process [10]. This method has several advantages, including the ability to learn complex and non-linear relationships, potentially achieving higher security than traditional methods, and the

ability to retrain the network with new data to adapt to evolving threats. However, there are disadvantages, such as the need for significant computational resources and vulnerability to adversarial attacks, where attackers manipulate the input data to cause the network to make incorrect predictions. It can be challenging to analyze the security properties of a trained neural network due to the difficulty in understanding its internal workings.

3.3.4.7 Chaotic map

Chaotic map-based image encryption methods are a category of encryption techniques that leverage the properties of chaotic systems to secure digital images. Due to their unpredictable behavior and sensitivity to initial conditions, chaotic maps have become a popular tool for image encryption. These methods are beneficial in applications where high security and computational efficiency are crucial [64, 40, 85, 41]. This technique is the basis of our research in this thesis, and therefore, we will discuss it in detail in the next section.

3.3.4.8 Quantum theory

Quantum theory-based image encryption is a rapidly developing field that uses quantum mechanics principles to create theoretically unbreakable encryption schemes [60]. The field explores properties like superposition and entanglement, which haveno classical counterparts. These properties allow for encoding complex information and intertwining two qubits, even at vast distances.

Researchers are exploring several theoretical approaches for quantum image encryption, such as Quantum Key Distribution (QKD), Quantum Image Representation (QIR), and Quantum Image Processing. These techniques use entangled qubits to securely distribute encryption keys, ensuring the integrity of encrypted data. However, significant challenges remain before widespread practical implementation, such as quantum hardware complexity, error correction, and scalability. Quantum cryptography is still in its early stages and requires specialized equipment.

3.3.4.9 Visually meaningful encryption

Visually meaningful encryption is a method that encrypts an image while preserving its visual content, such as edges, shapes, and textures, in a way that is recognizable to the human eye. This approach differs from conventional encryption techniques, which can obscure the visual content, making it unsuitable for certain applications. One popular method is to divide an image into multiple regions and encrypt each

region individually using a technique that preserves the visual content [35, 98]. Important concepts encompass partial encryption, feature retention, and progressive encryption. Common techniques include edge-based encryption, region-based encryption, and layer-based encryption. Edge-based encryption encrypts the edges of objects while keeping the main structure visible; region-based encryption encrypts specific regions based on importance or sensitivity, such as faces or text areas; and layer-based encryption separates different layers of an image, allowing some layers to remain visible, such as color layers (RGB). Visually meaningful encryption offers a trade-off between security and visual content preservation, making it useful in applications like medical imaging, where doctors and medical professionals need to recognize the visual content of the image.

3.3.4.10 Multi-image encryption

Multi-image encryption is a technique that uses a single key to encrypt multiple images simultaneously, resulting in efficient encryption and decryption. It is especially advantageous for extensive collections of images and data compression. These approaches seek to increase both security and efficiency. This approach is especially valuable in applications necessitating safeguarding extensive datasets, such as multimedia communication systems, medical imaging, and surveillance systems [98]. Important principles in multi-image encryption encompass concurrent encryption, data insertion, and hierarchical encryption. Simultaneous encryption is a process that encrypts several images simultaneously, whereas data embedding involves the embedding of many images into a single encrypted image. Layered encryption is a technique that encrypts images by applying many layers of encryption, resulting in a single composite encrypted image. In general, multi-image encryption provides strong solutions for protecting extensive datasets and enhancing storage efficiency.

3.4 Encrypting images using chaotic maps

The unique properties of chaotic maps (see 1.2.2), particularly their extreme sensitivity to initial conditions, ability to generate pseudo-random number sequences, ergodicity, and ease of implementation, make them a highly powerful and effective tool for image encryption.

In recent years, numerous techniques have been developed that utilize chaotic maps for image encryption. The diversity and variation in the types of chaotic maps, including one-dimensional (1D) and multi-dimensional (MD) maps, have significantly contributed to the development of a wide range of innovative algorithms in this field.

Chaotic encryption algorithms are highly adaptable in their implementation. These approaches may use one-dimensional chaotic maps or multi-dimensional maps, each presenting distinct benefits regarding complexity and security. Moreover, these approaches may be hybridized, combining multiple chaotic maps to enhance the encryption's unpredictability and resilience against attacks. Additionally, chaos-based encryption can be combined with other encryption methods, creating a multi-layered approach that further enhances security. A hybrid method can include image encryption techniques, such as those we discussed in the previous section (see 3.3.4), adding another layer of protection and leveraging the strengths of both chaotic and other established methods.

3.4.1 Typical architecture of chaos based image cryptosystems

Chaos-based image cryptosystems utilize the properties of chaotic maps to achieve secure encryption of digital images. The typical architecture of such a cryptosystem consists of several key components and processes that ensure both confusion and diffusion of image data [93]:

Figure 3.10 illustrates a typical architecture for a Chaos-Based Image Cryptosystem. The system consists of two main stages: the confusion stage and the diffusion stage.

During the confusion stage, the image pixels are secretly rearranged to create confusion and make it difficult for attackers to analyze the encrypted image.

In the diffusion stage, the influence of each pixel is spread throughout the entire image using mathematical operations such as XOR, addition, or subtraction.

The confusion and diffusion stages are repeated n times, with n typically being greater than 1, followed by a single round of diffusion repeated m times.

Multiple rounds of confusion and diffusion are conducted until a satisfactory degree of security is achieved.

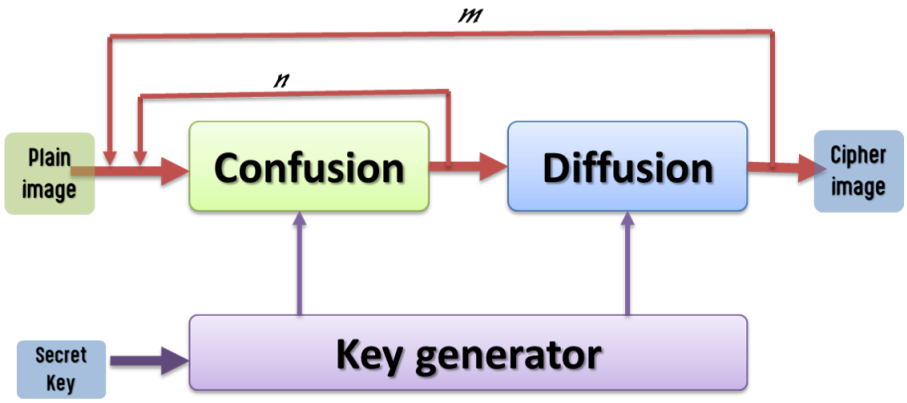


Figure 3.10: Typical architecture of chaos based image cryptosystems

3.4.1.1 The encryption key

In chaos-based image cryptosystems, the encryption key typically includes the set of parameters and initial conditions of the chosen chaotic map used to generate chaotic sequences. In our proposed algorithm, the key is The Extraparam generated from the original image and a key image, as we'll explain in the section.

- Parameters and initial conditions of the chosen chaotic maps are used to generate the Key in the encryption process.
- Conversely, in the decryption process, the Key is used to generate parameters and initial conditions of the chosen chaotic maps.

3.4.1.2 Chaotic Sequence Generation

First, we must calculate the parameters $(a, b, .., \alpha, \beta.)$ and initial conditions $(x_0, y_0...)$.

Then, The chaotic map $f(x)$ is iterated repeatedly, generating a sequence of pseudo-random numbers: $x_1, x_2, ..., x_n$. This sequence X is often used as the core for encryption operations.

3.4.1.3 pre-processing process

The encryption process is usually preceded by the image pre-processing process, it contains two stages (which are not mandatory):

- Step 1: Normalize the image: The image pixel values are converted to a normal range [0, 1].

  If we have an image I with pixel values p in the range $[0, 255]$, then the normalization is: $p = \frac{p}{255}$

- Step 2: Convert the image to a 1D array: Flatten a 2D image matrix I of size $m \times n$ into a 1D matrix I of length $N = m \times n$.

- Sometimes, The plaintext image is divided into smaller blocks or pixels to facilitate the encryption process.

3.4.1.4 Encryption Process

As mentioned previously, chaos-based image encryption processes rely on confusion and diffusion processes.

- Confusion (Pixel Permutation)

  Permutation using Chaotic Sequence: Generate a permutation index array P by sorting the chaotic sequence X and obtaining the indices of the sorted sequence.

  Let X_{sorted} be the sorted sequence and P be the permutation such that $X_{sorted} = X_P$.

  Permute the pixels of the 1D image array P according to :$I_P = Permute(I, P)$

- Diffusion (Pixel Value Transformation)

  Transforming Pixel Values: Use the chaotic sequence X to modify the pixel values of the permuted image I_P, A common operation is the XOR (exclusive OR) operation.

  eg. $I'' = X \otimes I'$

3.4.1.5 Decryption Process

The decryption process involves reversing the encryption steps using the same chaotic sequence and permutation indices generated during encryption. The parameters and initial conditions required for decryption are derived from the encryption key, ensuring that the chaotic sequences and permutations are accurately reproduced.

3.5 Image encryption evaluation metrics

We need reliable evaluation metrics to evaluate the effectiveness and security of image encryption methods. These metrics provide a comprehensive framework for assessing the impact and efficiency of the encryption technique. The figure 3.11 below summarizes the various metrics used to assess the quality and efficiency of an image encryption algorithm. A detailed discussion of these metrics can be found in the experiments section.

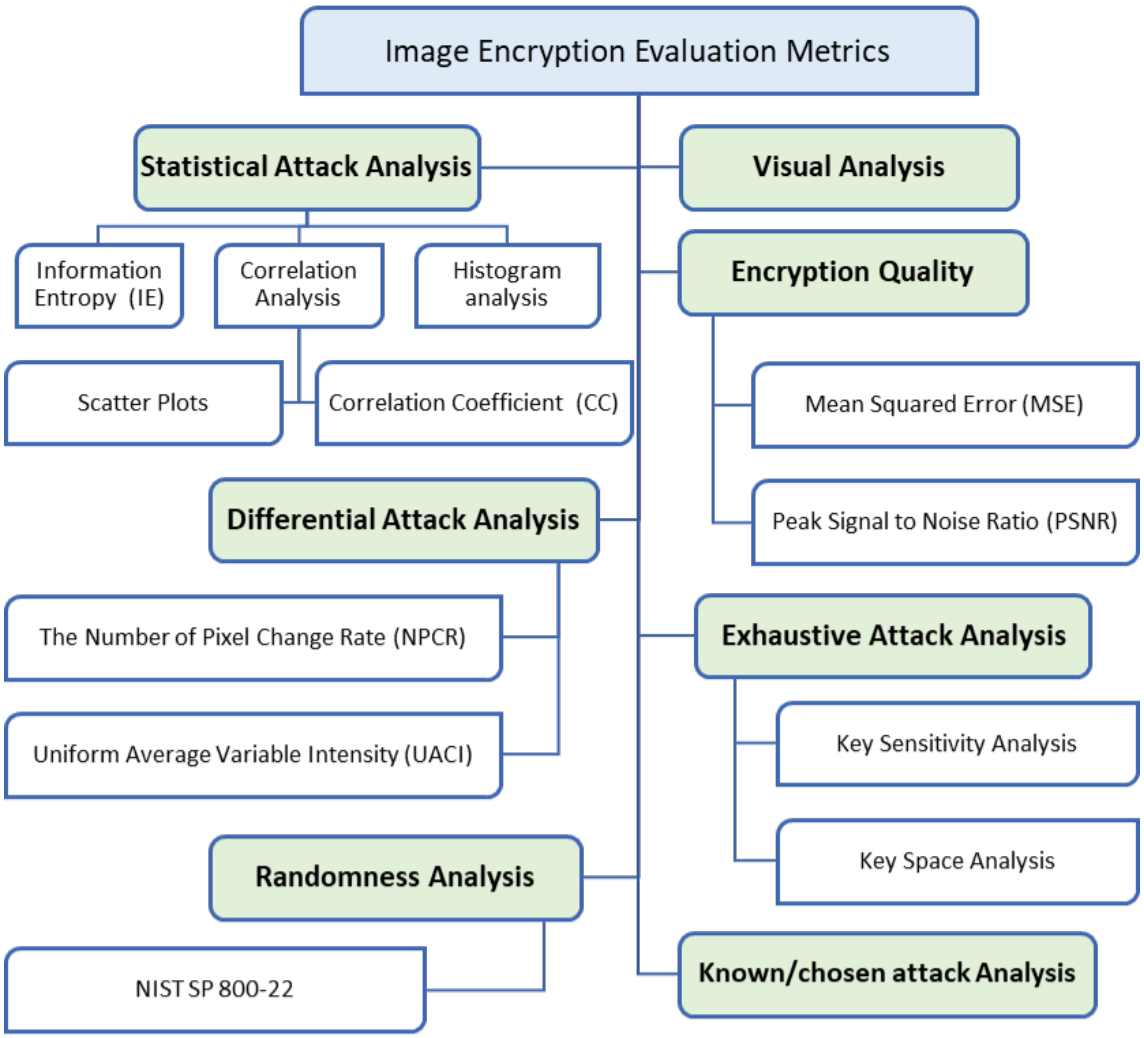


Figure 3.11: Image Encryption Evaluation Metrics

3.6 Related works

Due to their many advantages, chaotic maps are very suitable for encryption algorithms, which has led researchers to focus on this technique [63, 20, 87, 101, 18]. In

this section, we focus on presenting some recent works related to our research.

Chaos theory is a branch of mathematics and physics and is the mathematical embodiment of chaotic behaviours that can be observed in nature (population growth, weather and climate, ecology, economics, etc.) or in research and development laboratories, along with various systems such as electrical circuits, chemical reactions, mechanical systems, and so on [58, 52, 85, 45, 89]. Chaotic concept was summarized via Edward Lorenz, father of chaos theory as follows: "When the current determines the future, but the approximate current does not approximately determine the future" [52].

The chaotic map system is one of the most important nonlinear random systems known for its properties and extreme sensitivity to initial conditions and control parameters. They are also characterized by unpredictability, ergodicity, non-periodicity, pseudorandomness (random-like behavior) and reproduction [58, 8]. This makes chaotic maps very suitable for generating random number sequences. They can quickly and accurately generate a chaotic sequences that can therefore be used with encryption algorithm [102].

Mathematically, the chaotic maps are formulated in terms of difference equations as follows: $x_{n+1} = f(x_n)$, where $x \in R, n = 0, 1, 2, ...,$. It can be represented as a recurrence relation. Some researchers divide chaotic maps into two categories: one-dimensional (1D) chaotic maps and multidimensional (MD) chaotic maps [67]. Typically, chaotic 1D maps have simple structures with one variable and few parameters. They are easy to implement in image encryption systems and are also characterized by high execution speed. Among these systems, the Logistic [30], the Tent [49] and the Henon maps [31] are the most commonly used systems in image encryption algorithms. However, the above chaotic systems have the limitations of chaotic performance, which lead to some shortcomings in the encryption of information, such as small key space, insufficient complexity, and low security, making the corresponding encryption algorithms easy to decrypt.

In multidimensional maps, the structures are more complex because they are a set of equations with multiple variables and multiple parameters, which makes their chaotic behaviour better [47]. That makes their chaos trajectories more unpredictable [40]. However, the chaotic maps of MD have high computational costs and are difficult to implement, especially for real-time processing [67].

Due to the distinctive characteristics of chaotic maps, such as unpredictability, ergodicity, and sensitivity to the initial value, algorithms based on them have been in the focus of researchers for the last two decades [3]. Therefore, a variety of algorithms for chaos-based cryptographic systems have been proposed to increase

the efficiency of encryption methods [6, 85, 40]. This has been the case since the first research, in which the American researcher Friedrich proposed in 1997 [23] to use an algorithm based on chaos theory for image encryption. But, originally, the idea of using chaotic maps in cryptography first appeared in a 1989 article by Matthew [54].

To avoid the drawbacks of image encryption algorithms based on simple chaotic maps, many researchers resort to hybrid methods that use multiple maps [64, 90, 40, 85, 41] or with other technologies such as DNA computing [51, 6, 106], neural network [53] , cellular automata [46], compressive sensing [12, 35, 98], optical transformation [41], and asymmetric encryption [36, 100]. Moreover, other researchers have created new chaotic maps by improving or modifying existing ones [30, 66, 96, 65, 41]. Additionally, a characteristic structural design of the prevailing chaos-based image cryptosystems includes two main process, namely the confusion and the diffusion processes [85]. In the confusion phase, permutations of image pixels or bits or blocks are prepared, scrambling the positions over the entire image without varying their values. This phase is necessary to reduce the high correlation between the two adjacent pixels. In this way, the image becomes unrecognizable, but it is not very safe to use only the permutation stage. To increase safety, the diffusion stage aims to change the value of each pixel in the entire image. The pixel values are changed one by one by the sequence generated by the chaotic systems. The whole confusion-diffusion round is repeated several times to reach a satisfactory security level [72].

In [26], the authors propose a novel image encryption scheme that enhances the Advanced Encryption Standard (AES) by utilizing a variable S-box generated by a hyperchaotic system. The plain image is divided into 128-bit blocks, and each block is encrypted using AES. Unlike classical AES, which uses a single public S-box, this method generates a unique S-box for each block encryption. The hyperchaotic system produces a sequence of values that are sorted to create the S-box, with the final value of one sequence serving as the initial condition for the next. This dynamic S-box generation ensures that identical original blocks result in different encrypted blocks, significantly improving security. The experimental and analytical results demonstrate that this approach effectively enhances security, expands the key space, and offers strong resistance to various attacks.

The study in [17] proposes an asymmetric image encryption algorithm based on SHA-3 and compressive sensing to prevent unauthorized access to private images and ensure secure communication. The algorithm generates a random matrix, performs modular-addition operations, computes hash values, and generates three plaintext

keys using RSA. A new mathematical transformation model is used to transform keys into chaotic systems, compresses the plain image, and uses discrete wavelet transformation to generate components of high and low frequencies. The method resists known and chosen plaintext attacks.

The paper [63], introduces a new Logarithmic Chaotic Map (LCM), inspired by quadratic maps, with superior chaotic properties. The LCM's output passes all 15 NIST statistical tests, confirming its randomness. The paper also develops a new encryption algorithm based on LCM, ideal for encrypting numeric sensor data values in IoT applications. The algorithm is highly resilient to security attacks, providing robust protection for sensitive data in IoT environments.

In [10], the research presents a novel image cryptosystem that integrates DNA sequence operations, a Single Neuron Model (SNM), and a chaotic map. The system's initial conditions and parameters are derived from a 512-bit hash value, which is highly dependent on the plain image. The encryption algorithm follows a confusion-diffusion architecture. The 2D Logistic-adjusted-Sine map (2D-LASM) is utilized to simultaneously confuse the pixels of the color components, while the SNM generates the key stream. Additionally, the hash value of the plain image is injected into the diffusion process. Experimental results and security analysis show that this encryption scheme offers the highest security level due to its sensitivity and large key space.

In [64], the paper presents an image encryption method using chaotic maps and genetic operations. The Keccak algorithm is used to compute the hash values of a plain-image, allowing for pseudorandom sequences through an iterative logistic map. Genetic operations at the bit level are combined with the Hénon map and DNA coding technique to achieve pixel selection, crossover, mutation, and pixel diffusion and scrambling. The algorithm's diffusion and confusion features are strengthened by bidirectional exclusive OR operations with chaotic sequences. The theoretical analysis and simulation results show the algorithm is sensitive to keys and can effectively defend against statistical and differential attacks.

The paper [85] presents a novel color image encryption scheme using the Liu Lorenz-XOR Zigzag Arnold (LL-XZA) encryption algorithm. The scheme uses a hyperchaotic system for the 3D Liu system and a pseudorandom number sequence for the 4D Lorenz system. The LL-XZA algorithm significantly improves color image security performance, offering superior stability, visual security, and robustness.

The authors in [50] propose an improved chaos system for image encryption, which has been shown to have better dynamic characteristics, pass NIST tests, and good correlation properties. They also propose a novel image encryption algorithm

based on this system, which is more secure and efficient. The algorithm uses random selection of the least significant bit of a pixel, bit-plane shifts, and crossover-boxes for pixel swapping. It achieves higher mean Shannon entropy and lower mean Chi-square value, and can achieve more ideal NPCR and UACI with a single encryption round.

The article in [2] introduces a novel image encryption algorithm designed for Industrial Internet of Things (IIoT) environments with limited resources. The algorithm uses a chaotic model to generate a single data sequence and generate three matrices, which can be used to encrypt multiple images under the same session key. Experimental results show the algorithm's effectiveness on an IoT camera sensor and a separate device for decryption. Statistical tests confirm its robustness against various attacks and its superior security and efficiency, making it suitable for IIoT deployment.

The work in [100] provides an improved Lorenz system (ImproLS) and an asymmetric image encryption scheme based on blind signature and ImproLS. The sender uses the public key to produce cipher keys, using the RSA algorithm. The cipher image is obtained and sent to the receiver, where the signer encrypts the digital signature image using Arnold transform and DWT. The receiver checks the signature and recovers the cipher image. Experimental results show high information entropy and resistance to salt and pepper and clipping attacks.

3.7 Conclusion

This chapter commenced with a comprehensive overview of image encryption techniques, delving into their diverse types and classifications. We explored the various approaches employed to secure sensitive image data, providing a foundational understanding of the field. Subsequently, we focused on algorithms based on chaotic maps, which piqued our interest due to their demonstrated effectiveness in generating complex and unpredictable sequences, recognizing their potential to provide robust and secure encryption solutions. The complex and unpredictable nature of chaotic systems offers a promising foundation for developing algorithms that are resistant to traditional cryptanalytic attacks.To establish a solid foundation for our proposed image encryption algorithm, we conducted a thorough review of existing research in the field. This exploration of the state-of-the-art enabled us to identify the strengths, weaknesses, and limitations of previous approaches, guiding our development efforts towards a more innovative and secure solution. With this comprehensive understanding in place, we are now prepared to present our proposed

image encryption algorithm in the following chapter.

Chapter 4

The proposed Image encryption algorithm

Introduction

This chapter provides a comprehensive description of the novel image encryption algorithm we have developed. The algorithm utilizes the enhanced chaotic maps to ensure robust security and resistance against various attacks. We will delve into the details of the encryption process, including the key generation mechanism, the chaotic map initialization, the image confusion and diffusion operations, and the final decryption steps. The proposed algorithm aims to offer a secure and efficient solution for protecting sensitive image data[11].

4.1 Motivation

To increase the efficiency of our proposed scheme, the architecture of this algorithm depends on three main things:

- **Use of multiple chaotic 1D maps**: due to the strength and speed of chaotic maps in generating random number sequences and to avoid the shortcomings of one-dimensional maps and multidimensional maps, we selected three modified and improved 1D maps :

  1. improved logistic map ILM,
  2. logistic Mayan system LOMAS,
  3. improved sinusoidal map ISM.

  due to their robust performance, which was confirmed by Lyapunov and bifurcation analyses in the previous section.

- **Use the key image**: we used another image as a key image to generate parameters and initial conditions for the used maps; we also used this image

in the diffusion phase with the XOR operator as a mask. The size of this image does not matter as we will resize it to the appropriate size when we use it.

- **ExtraParam**: We proposed a novel variable, designated as "ExtraParam," to enhance the sensitivity of our encryption scheme to image modifications. This variable is derived from the original image through a carefully designed extraction process, ensuring that it is highly susceptible to even the most subtle changes in the image data. Consequently, any alteration, regardless of its magnitude, even if this change is only one bit, will result in a significant variation in the value of ExtraParam. This variable is subsequently employed to dynamically generate parameters and initial conditions for the chaotic maps utilized in our encryption algorithm, thereby strengthening its resilience against various attacks.

4.2 The used chaotic maps

In this section, we present the related preliminaries of chaotic maps for the proposed image encryption algorithm.

The chaotic 1D maps (such as the logistic, May, and sine maps) have some drawbacks, such as simple behavior and small chaotic intervals, which are evident in the bifurcation diagram and the Lyapunov exponent. These shortcomings can have a negative impact on some chaos-based applications, especially in encryption methods [102]. Moreover, these limitations are in fact security holes that cryptanalysis can exploit to crack the cryptosystem due to the weakness in generating the encryption keys [102, 75].

To avoid the aforementioned 1D disadvantages and to overcome the computational complexity and implementation difficulties related to multidimensional maps, we have opted to use improved and modified one-dimensional (1D) chaos maps. These 1D maps retain the advantages of simpler structure, easier implementation, and faster execution speeds compared to their multidimensional counterparts.

Specifically, we have selected three maps for our analysis: the Improved Logistic Map (ILM), the Logistic-May System (LOMAS), and the Improved Sine Map (ISM). According to our experiments and diagrams, these selected maps exhibit excellent chaotic behavior.

Our choice of these particular maps is based on several key factors. Firstly, they are all 1D maps that have been modified and improved from their original forms. Secondly, they demonstrate highly chaotic characteristics, as evidenced by

their bifurcation diagrams and positive Lyapunov exponents. Thirdly, their parameters have a wide range of possible values, allowing for greater flexibility in their application.

By leveraging these improved 1D chaos maps, we aim to avoid the computational and implementation challenges associated with multidimensional maps while still preserving the advantages of chaotic systems for our research or application.

4.2.1 Improved Logistic map (ILM)

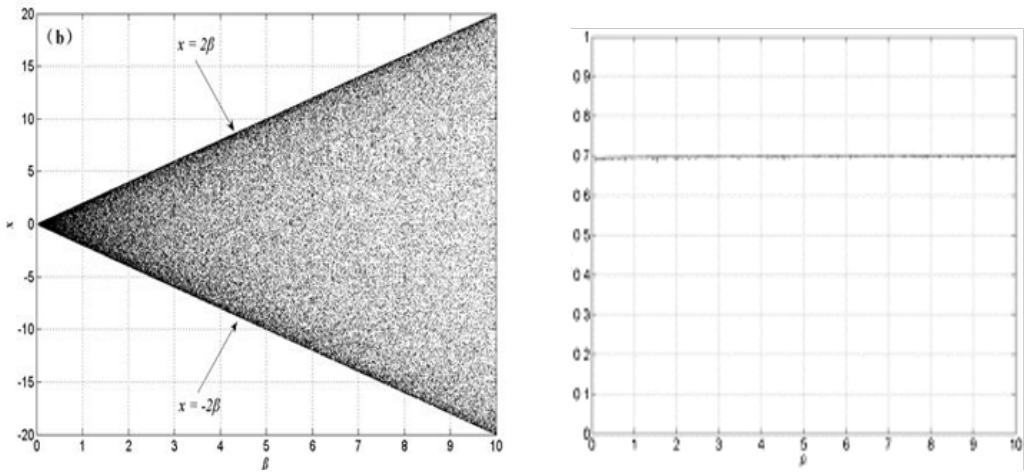
In [30], the author presents a modified Logistic map Based on ordinary Logistic map, with the following expression.

$$x_{n+1} = 2\beta - x_n^2/\beta \tag{4.1}$$

Where :

- The parameter $\beta \in]0, \infty[$ (Instead of $]0, 4]$ in the seed map) and
- x_n is in $[-2\beta, 2\beta]$ (Instead of $[0, 1]$ in the seed map),

This map has parameters, outputs, and initial conditions with much larger ranges than the original maps, which increases the key space, Lyapunov coefficients for ILM are always positive (showed in figures 4.1a and 4.1b This shows that ILM has chaotic substantiality and better mixing property.



(a) The bifurcation diagrams. (b) The Lyapunov exponent.

Figure 4.1: The bifurcation and The Lyapunov exponent of Improved Logistic map [30].

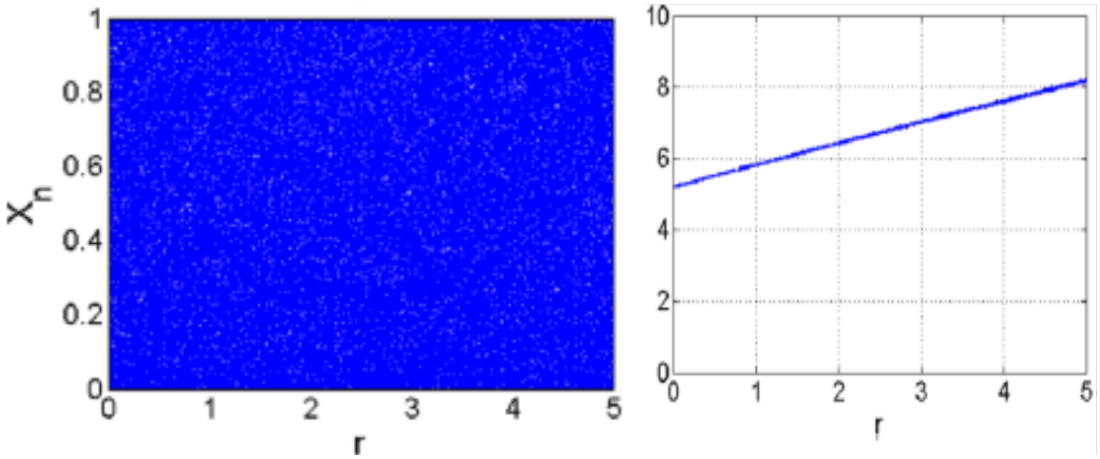
4.2.2 The Logistic-May System (LOMAS)

It is one of the maps created in [65], is made of the Logistic and May maps and is called (LOMAS). Its equation is written as:

$$x_{n+1} = (x_n \exp((r+9)(1-x_n)) - (r+5)x_n(1-x_n)) \bmod 1 \tag{4.2}$$

Where $x_n \in [0,1]$ and $r \in [0,5]$.

The bifurcation diagram is shown in figure 4.2a, there are no free white areas indicating no isolated values, and the entire area is almost completely covered. Thus, we can see that the chaotic properties within $[0,5]$ are excellent. Lyapunov coefficients are always positive with a maximum value equal to 8.3 as shown in Figure 4.2b.



(a) The bifurcation diagrams. (b) The Lyapunov exponent.

Figure 4.2: The bifurcation and the lyapunov exponent of logistic-may system map[65].

4.2.3 Improved Sine Map (ISM)

In [66], the authors modified the sinusoidal map to overcome the weaknesses of the sinusoidal map by adding a parameter to the map equation and creating a new equation as :

$$x_{n+1} = \lambda \sin(\pi . x_n) + p \tag{4.3}$$

Where x_n values are restricted to the interval of $[1/\alpha, 1-(1/\alpha)]$ with $2 < \alpha < \infty$. With

$$\lambda = \frac{\alpha - 2}{\alpha[1 - \sin\frac{\pi}{\alpha}]} \tag{4.4}$$

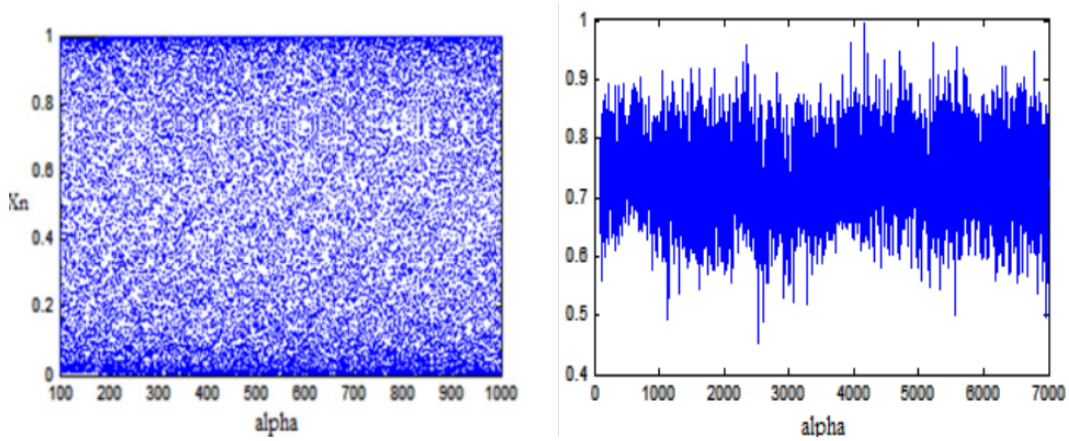
and

$$p = \frac{\alpha - 1}{\alpha} + \frac{2 - \alpha}{\alpha[1 - \sin\frac{\pi}{\alpha}]} \tag{4.5}$$

Substituting these values (in (4.4) and (4.5)) to the (4.3), we obtain the following final equation :

$$x_{n+1} = \frac{\alpha - 2}{\alpha[1 - \sin\frac{\pi}{\alpha}]}[\sin(\pi.x_n) - 1] + \frac{\alpha - 1}{\alpha} \tag{4.6}$$

The Lyapunov coefficients for ISM are always positive (Figure 4.3b). This figure shows that ISM has chaotic substantiality and a better mixing property. The bifurcation diagram (Figure 4.3a) shows no free white areas, indicating that there are no isolated values and almost the entire range is covered; thus, we can see that the chaotic properties are excellent. More importantly, different values of α can be used to produce different outputs that increases the key space of the cryptosystem.



(a) The bifurcation diagrams. (b) The Lyapunov exponent.

Figure 4.3: The bifurcation and The Lyapunov exponent of Improved Sine Map [66].

4.3 Initialization

To generate the initial values containing the conditions and parameters, we apply the Algorithm 1.

Algorithm 1 *generate_parameters*$(img_k, ExtraP_P)$

Input : img_k : Key image ; $ExtraP_P$: ExtraParam of Plain image;
Output: $r, X1_0$:LOMAS parameters; $\beta, X2_0$:ILM parameters; $\alpha, X3_0$:ISM parameters;
$[m, n, d] = size(img_k)$;
$hash = SHA256(img_k)$;
$mx_h = max(hash)$;
$mn_h = min(hash)$;
$moy_h = mean(hash)$;
$im2 = imresize(flip(img_k), [mn])$;
$cc = corr2(img_k, im2)$;
$mse = immse(img_k, im2)$;
$p = psnr(img_k, im2)$;
$ExtraP_K = \left(mse \times cc \times p \times N_{Big} + mx_h \times {moy_h}^{mn_h \bmod 10}\right) \bmod A_{number}$;
$r = ((mse + ExtraP_P^3 + ExtraP_K^2) \times ExtraP_P) \bmod 5$;
$X1_0 = ((mx_h \times ExtraP_P + ExtraP_K^2)/(ExtraP_P^2 + moy_h) + mse) \bmod 1$;
$\beta = moy_h/mx_h \times ExtraP_P + (ExtraP_K^2 + abs(cc) * ExtraP_P^2/(n * ExtraP_P + ExtraP_K + d)$;
$X2_0 = (ExtraP_K^2 + p \times ExtraP_P/(mn_h + 1)) \bmod \beta$;
$\alpha = 2 + (ExtraP_P^3 + ExtraP_K^2)/(ExtraP_P + Extra \bmod PK - cc \times ExtraP_P)$;
$X3_0 = 1 - (((p \times ExtraP_P^2 \times ExtraP_K^2)/(mn_h + 1)) \bmod 1) + 0.05/\alpha)$;

Where $ExtraP_P$ is the $ExtraParam$ extracted from the plain image, $ExtraP_K$ is the $ExtraParam$ extracted from the key image, and img_k is the key image.

The following formula is used to extract ExtraParam number:

$$ExtraParam = \left(M_{2img} \times C_{2img} \times P_{2img} \times N_{Big} + mx_h \times {moy_h}^{(mn_h \bmod 10)}\right) \bmod A_{number} \tag{4.7}$$

Where

$$hash = SHA256(img1);$$
$$mx_h = max(hash);$$
$$mn_h = min(hash);$$
$$moy_h = mean(hash);$$

And

$$M_{2img} = immse(img1, img2);$$
$$C_{2img} = corr2(img1, img2);$$
$$P_{2img} = psnr(img1, img2);$$
$$N_{Big} = \ a\ Very\ big\ number;$$

Here img1 is the original image, img2 is the flip(img1); in our experiments, we gave N_{Big} the value 10^{11}, and A_{number} the value 999, One of the most important characteristics of chaotic maps is their extreme sensitivity to initial conditions and parameters; hence, the importance of $Extraparam$ becomes clear. To see how $Extraparam$ is affected by a bit change in the image, we compute the parameter values for multiple copies of Lena's image with a bit change in each copy, with alternating arguments N_{Big} and A_{number}.If the value of $ExtraParam$ is NaN or zero, we change the value of the first bit in the image and recalculate the parameter, this happens in the case of one-color image (such as a black or white image) or in images that exactly match their rotation. To increase the key sensitivity, we calculate the $ExtraParam$ from the key image and use it in preparing the used chaotic maps. The Table 4.1 shows the obtained results. The difference between each value is very clear, which shows the high sensitivity of the parameter $Extraparam$ to very small changes.

N_{big}	10^{10}	10^{10}	10^{11}	10^{11}	10^5	10^5
A_{number}	**999**	**9999**	**999**	**9999**	**999**	**9999**
lena (original)	812	2702	133	6469	996	2607
lena (1,1,1)	319	4702	188	7577	430	1555
lena (1,1,8)	260	8873	105	3462	211	4333
lena (15,245)	68	7205	178	7333	740	8606
lena (111,111)	98	5138	990	279	716	6368
lena (133,31)	651	1983	526	8725	838	487
lena (190,231)	593	7190	930	2460	352	514
lena (204,211)	672	6531	731	4763	967	1453
lena (255,45)	722	2414	217	5248	572	7259

Table 4.1: ExtraParam values with a bit change of lena image and in terms of N_{big} and A_{number}

Table 4.2 shows the change in the values of the initial conditions for the maps used according to the value of $ExtraParam$, it is very clear the large difference in the values, although the difference in ExtraParam is very small; This is another proof of the power and importance of $ExtraParam$ (133 is the ExtraParam extracted from Lena 256 X 256 , peppers image as Key image).

4.4 Image encryption process

The overall architecture of our encryption scheme is shown in Figure 4.4. The encryption algorithm has been divided into three phases:

- **Phase 1**: This is the preparation stage, where the variables are prepared (the most important of which is $Extraparam$, which goes into setting all the

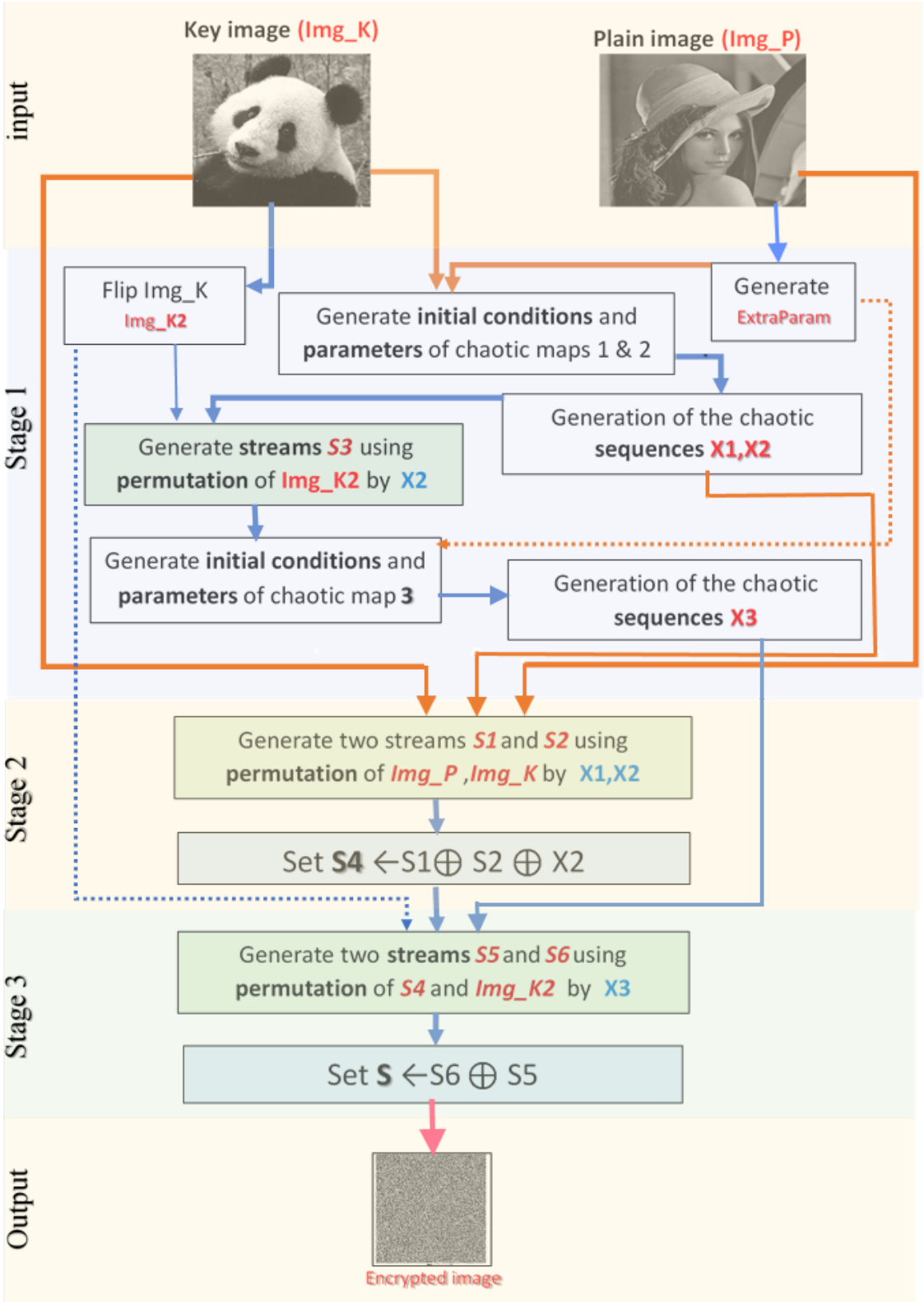


Figure 4.4: The flowchart of the proposed encryption process

$ExtraParam$	ILM	LOMAS	ISM
132	4.8012	0.1442	0.5488
133	11.9246	0.6676	0.8865
134	45.0632	0.1999	0.0041
150	52.7136	0.7501	0.9525
151	19.9309	0.4029	0.3285
152	10.9271	0.0614	0.4844

Table 4.2: Values of initial conditions (x_0) for the maps used according to the value of $ExtraParam$

parameters) and the initial values for the chaotic maps used are generated. At this stage, we create two chaotic sequences X1, X2 by the first and second maps (Map1 and Map2), respectively, and use Img_K (the key image) as parameters. We also create an S3 stream using the permutation of Img_K2 by X2, where Img_K2 is the result of rotating Img_K by 180 degrees, and then resize Img_K2 to match the size of the plain image. At the end of this phase, we generate the chaotic sequence X3 by the third map (Map3) using S3 to define the initial conditions and parameters.

- **Phase 2**: (Confusion and Diffusion Phase Level1) In this phase, we create two streams S1 and S2 using the permutation of Img_P (the plaintext image) and Img_K (the key image) by X1and X2, respectively. Then, stream S4 is created by an XOR operation between S1 and S2 and X2.

- **Phase 3**: (Confusion and Diffusion Phase Level2) In this phase, we generate two streams S5 and S6 by permuting S4 and Img_K2 by X3. In a final step, S streams are generated by an XOR operation between S5 and S6. The resulting encrypted image is the final stream S.

  To determine which map is map 1, map 2 or map 3, we tried all possible combinations; the test results showed that the best combination is the following: Map 1 is LOMAS, Map2 is ILM and Map 3 is ISM.

The proposed encryption algorithm consists of 9 steps, as shown in Algorithm 2.

4.5 Conclusion

In this chapter, we have thoroughly explained the proposed image encryption algorithm. We began by exploring the enhanced chaotic maps that serve as the core of our approach, ensuring the algorithm's security and randomness. Subsequently, we

Algorithm 2 $encryption(Img_P, img_k)$

Input : img_k : Key image ; img_p : Plain image;
Output: img_c : cipher image;
$getImageInfo(img_p)$; // Step 1
$ExtraParam \longleftarrow generate\_ExtraParam(img_p)$;
$Sz \longleftarrow MxN$;
$[r, x\_1_0] \longleftarrow generate\_parametrs\_LOMAS(img_k, ExtraParam)$; // Step 2
$[\beta, x\_2_0] \longleftarrow generate\_parametrs\_ILM(img_k, ExtraParam)$;
$X1 \longleftarrow LOMAS(r, x\_1_0, Sz)$; // Step 3
$X2 \longleftarrow Logistic\_Map\_M(\beta, x\_2_0, Sz)$;
$img_k2 \longleftarrow image\_rotate(img_k, 180deg)$; // Step 4
$S1 \longleftarrow genrate\_stream_1(X2, img_k2, img_p)$;
$[\alpha, x3_0] \longleftarrow generate\_parametrs\_ISM(S1, ExtraParam)$; // Step 5
$X3 \longleftarrow Sine\_Map\_ISM(\alpha, x3_0, S)$;
$img_kR \longleftarrow Resize\_key\_image(img_k, img_p)$ // Step 6
$S2 \longleftarrow genrate\_stream_2(X2, img_p, img_kR)$;
$S3 \longleftarrow genrate\_stream_3(X1, img_p)$;
$S4 \longleftarrow bitxor(S2, S3, X2)$; // Step 7
$S5 \longleftarrow genrate\_stream_5(S4, X3)$; // Step 8
$img_k2_R \longleftarrow Resize\_key\_image(img_k2, img_p)$;
$S6 \longleftarrow genrate\_stream_6(img_k2_R, X3)$;
$img_c \longleftarrow bitxor(S5, S6)$; // Step 9
Return img_c

delved into the concept of the key image, a vital component that introduces additional complexity and enhances the algorithm's resistance against attacks. Furthermore, we discussed the Extraparam parameter, a crucial element allowing flexible customization of the algorithm's behavior. By carefully selecting the Extraparam values, we can fine-tune the initial conditions and parameters, tailoring the algorithm to specific security requirements. Throughout the chapter, we have provided a detailed explanation of how these components interact to ensure the algorithm's robustness and effectiveness.The next chapter will focus on a comprehensive security analysis to validate the strength and security of our proposed algorithm.

Chapter 5

Security and performance analysis

Introduction

A robust and secure image encryption method should be efficient against various possible attacks, This is what we will verify in this chapter, which has been devoted to experimenting with various evaluation metrics, including: statistical analysis (histogram, correlation coefficient, information entropy, peak signal-to-noise error, and mean square error), differential attack analysis (number of pixel change rates, and average uniform change intensity), key analysis (key space and the effect of key image), and known/chosen attacks are examined. In addition, we will compare our proposed algorithm with other current algorithms.

To verify the proposed encryption algorithm's performance, the simulation experiment is performed on MATLAB R2017a on a computer with a Windows 10 (64-bit) operating system, Intel(R) Xeon(R) CPU E5-2609 0 @ 2.40GHz, and 8GB RAM. The test images are located in the USC SIPI image database (http://sipi.usc.edu/database). Some images were resized by resizing other images from the previous database using the paint.net v4 application.

5.1 Visual analysis

Visual Analysis is a subjective method used to compare an original image with an encrypted one. It involves visually inspecting the encrypted image to determine its perceptual similarity to the original image. The encrypted image should not reveal any details about the original image. If the encrypted image seems random and lacks distinguishable features, it indicates effective encryption. This metric is useful for evaluating the encryption algorithm's ability to preserve the original image's properties and details.

The figure 5.1 shows a group of original images, and below each one is the encrypted image. It is clear that there is a vast difference between the original image and the encrypted one; the encrypted image seems random, and it is impossible to infer anything about the original based on it. This proves that the proposed method

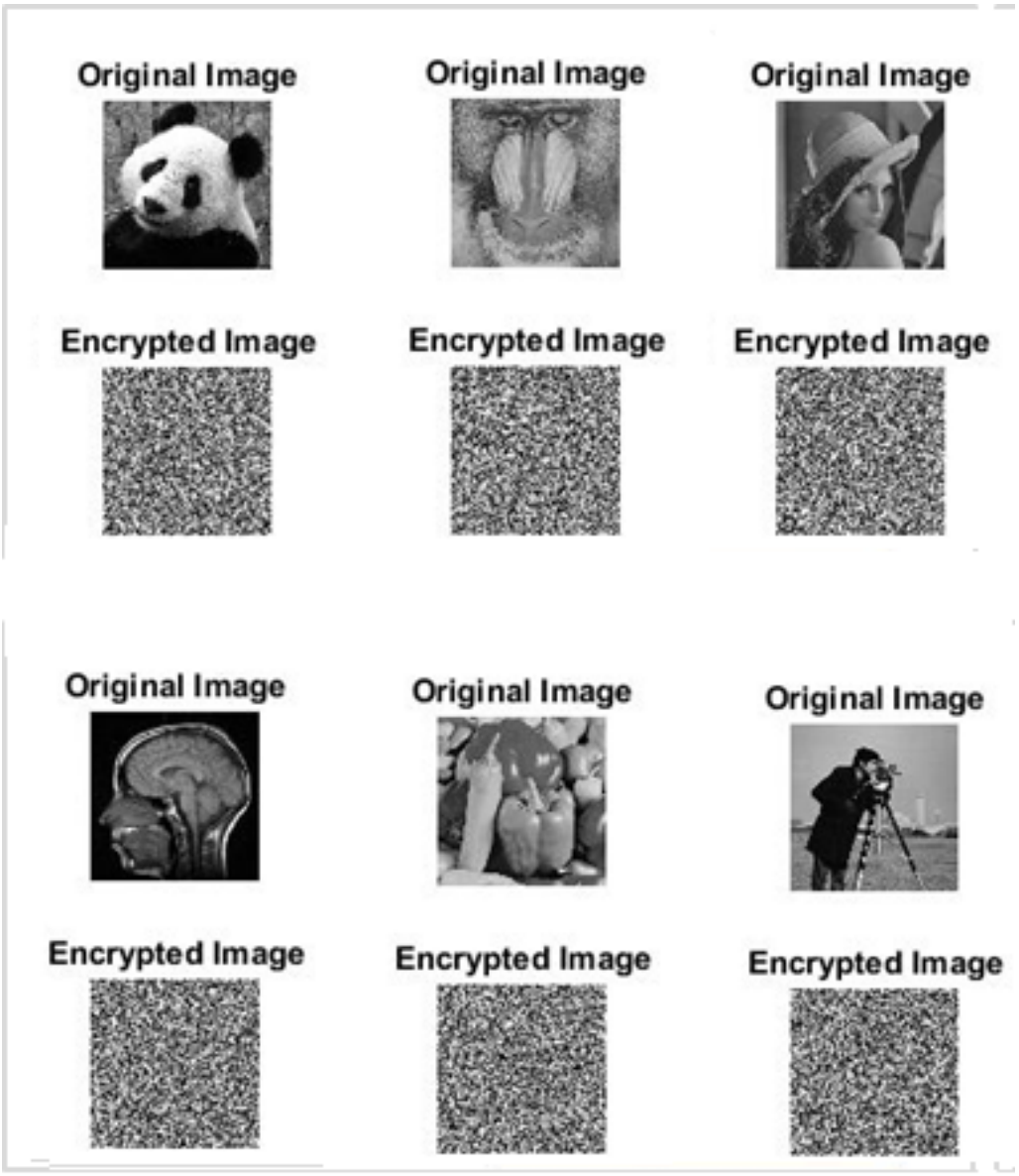


Figure 5.1: Visual analysis

preserves the perceptual properties of the original image well.

5.2 Statistical Attack Analysis

5.2.1 Histogram analysis

Histogram analysis is a crucial tool for evaluating the pixel intensity distribution of an encrypted image. It helps to identify patterns, and outliers, and assess the encryption algorithm's ability to preserve the original image's statistical properties. It also provides statistical information about the image, and the cryptanalyst can

get very useful information by analyzing it.

A histogram is a graph showing the distribution of pixel intensities in an image. It consists of adjacent rectangles whose area is proportional to the frequency of a variable, with no gaps between the bars. The x-axis represents the scale of values, and the y-axis represents the frequency. This metric is used to analyse patterns in data, find outliers, and check if a process has changed over time.

A significant deviation in the histograms indicates a noticeable alteration in the image content.

A good encryption algorithm should produce an encrypted image with a flat histogram, indicating a uniform and random distribution of pixel values.

To ensure the security of the image, the resulting histogram should exhibit a uniform distribution and be entirely distinct from the original [66].

Figure 5.2 shows the original image histogram, the encrypted image histogram, and the decrypted image histogram using the proposed algorithm (lena [256 x 256] as the normal image and panda [160 x 160] as the key image). In Figure 5.2, it is evident that obtaining any information from the encrypted image histogram is impossible. Additionally, the scrambled image differs significantly from the original image and is evenly distributed over all possible intensity values. Therefore, it can be concluded that the proposed algorithm effectively conceals image information. As a result, the proposed system is resistant to statistical attacks and demonstrates good confusion properties.

Figure 5.3 displays the histograms of all test and encrypted images with Panda [160 x 160] as the key image, exhibiting the same effect as previously observed.

5.2.2 Correlation analysis

The correlation between adjacent pixels in an image is crucial for image encryption. In a plain image, neighboring pixels often exhibit a strong correlation, leading to patterns and redundancies in the image data. In contrast, an encrypted image aims to reduce the correlation between adjacent pixels by scrambling pixel values. Ideally, the encrypted image should resemble random noise statistically, with minimal to no predictable relationship in terms of intensity values. This makes it computationally difficult to exploit these correlations and derive information about the original image from the encrypted version.

To analyze the correlation between neighboring pixels, you can use The correlation coefficient (cc) and Scatter Plots :

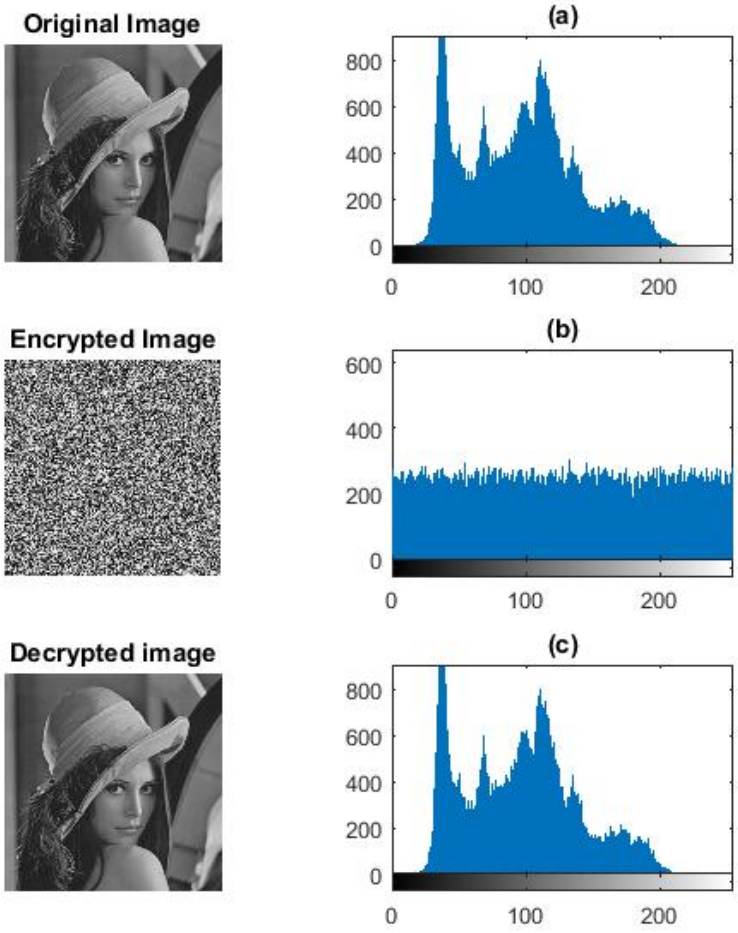


Figure 5.2: The Histograms of : (a) original image , (b) Encrypted image , (c) Decrypted image

5.2.2.1 Correlation coefficient

The correlation coefficient (CC) is a quantitative evaluation metric that measures the similarity between the original image and the encrypted image. it measures the correlation between adjacent pixels in an image. To resist statistical attacks, a strong encryption algorithm significantly reduces the correlation between neighboring pixels in an image. This randomness makes it much harder to guess the original image's content

In an encrypted image, the correlation between adjacent pixels should be close to zero, indicating that the pixel values are randomly distributed and statistically independent. But, A correlation coefficient close to 1 indicates a high similarity between the original and encrypted images.

We used the following equation to calculate the correlation coefficients:

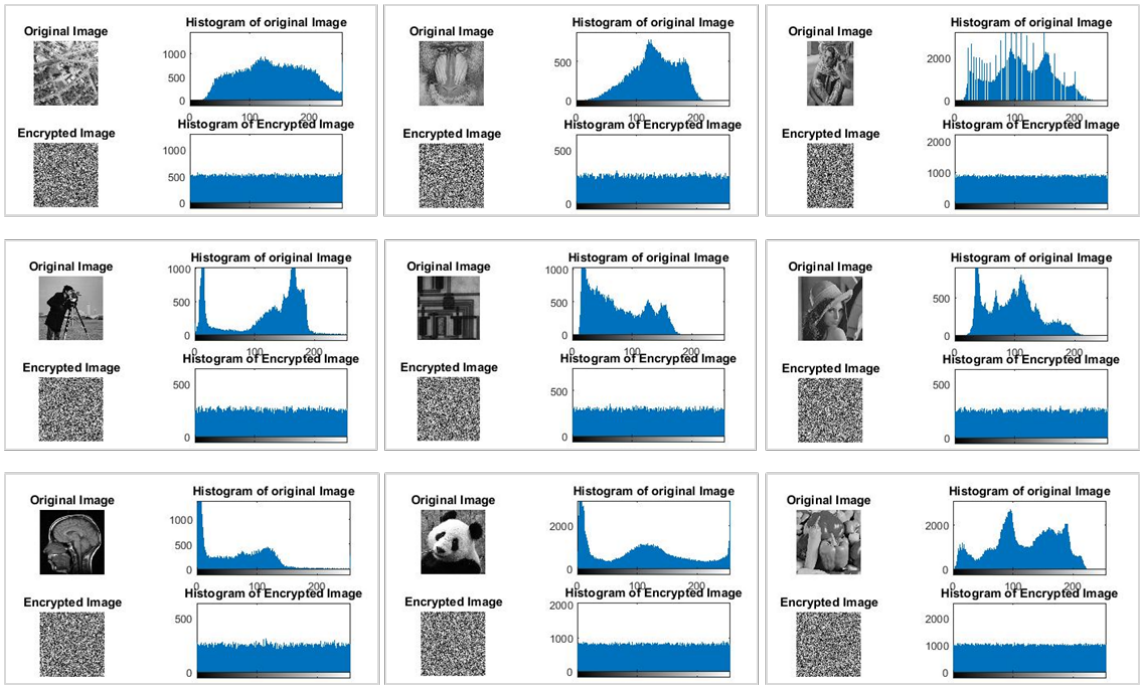


Figure 5.3: The Histograms of different test images (key image : panda[160 x 160])

$$r = \frac{Cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{5.1}$$

Where

$$Cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \tag{5.2}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2 \tag{5.3}$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N}x_i \tag{5.4}$$

Where x and y are two adjacent pixels in horizontal, vertical and diagonal directions, N is the total number of pixels selected from the image to calculate the correlation.

Table 5.1 shows the strength of correlation between the adjacent pixels of the original image in three directions. The results show that our proposed method scheme can reduce the correlation of adjacent pixels. From these results, it can be concluded that the proposed encryption algorithm cannot be broken with a statistical attack by correlation analysis between neighbouring pixels.

24emImage	Plain image			Encrypted image		
	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Lena260	0.9494	0.9667	0.9366	0.00283	-0.0017	0.00215
Aerial	0.9083	0.8891	0.8502	-0.0004	0.00257	-0.0024
Baboon	0.8677	0.8198	0.7800	-0.005	-0.0014	0.00155
Barbara	0.8271	0.9501	0.8310	0.0006	-0.0009	0.00125
Cameraman	0.9329	0.9566	0.9117	0.00933	-0.0021	0.00233
Circuit	0.9766	0.9775	0.9678	-0.0056	-0.0022	0.00218
Lena512	0.9691	0.9841	0.9639	0.00305	1.1E-06	-0.0022
MRI	0.9540	0.9598	0.9237	-0.0039	-0.0101	0.00366
Panda	0.9832	0.9794	0.9748	7E-05	-0.0041	0.00127
boat	0.8236	0.8329	0.7837	0.00117	0.00017	0.00033
Peppers	0.9733	0.9763	0.9650	-0.0006	0.00208	0.00204

Table 5.1: Correlation coefficient of images before and after encryption (key image : panda[160 x 160]).

3[6]***Scheme**	**Correlation coffcients**			
	Image File	**Horizontal**	**Vertical**	**Diagonal**
	Ideal value	$\simeq 0$	$\simeq 0$	$\simeq 0$
5[2]*Our scheme	Lena	-0.0036	-0.0045	-0.0041
	Barbara	0.0022	-0.0024	-0.0004
	Baboon	-0.0036	-0.0014	-0.0065
	Cameraman	0.0001	-0.0058	-0.0001
	Pepper	0.0004	0.0013	-0.0007
3[2]*Ref [64]	Lena	0.0055	0.0305	0.0042
	Baboon	-0.0138	-0.0267	-0.0187
	Peppers	0.0190	0.0029	-0.0072
3[2]*Ref [5]	Lena	0.0119	0.0092	0.0013
	Barbara	0.0136	0.0084	0.0146
	Peppers	0.0115	0.0109	-0.0101
3[2]*Ref [60]	Lena	0.0054	0.0019	0.0067
	Cameraman	-0.0037	-0.0054	-0.0021
	Peppers	0.0076	-0.0098	0.0087

Table 5.2: Comparison of the Correlation values between our proposed approach and the other methods

5.2.2.2 Scatter plots

Scatter Plots: Correlation between pixels can be visualized using scatter plots. Plotting the intensity values of adjacent pixel pairs can visually reveal the degree of correlation. A strong positive correlation will result in a cluster of points along a diagonal line, while a weak or no correlation will result in a scattered distribution of points.

In a plain image, the scatter plot for adjacent pixels would show a clustered pattern around a diagonal line.

In a well-encrypted image, the scatter plot would be more dispersed, with no clear relationship between neighboring pixel values.

Figure 5.4 shows the correlation of adjacent pixels in the horizontal, vertical, and

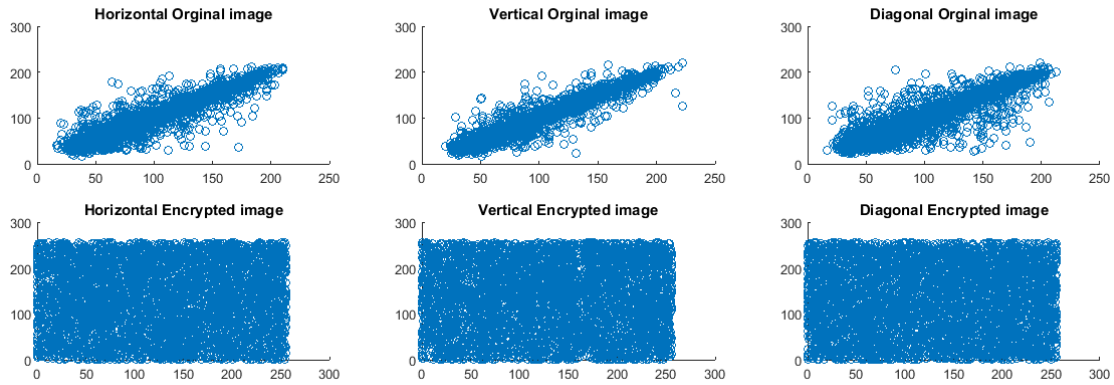


Figure 5.4: The Correlation of adjacent pixels in horizontal, vertical and diagonal direction for original Lena image and Encrypted image

diagonal directions of the plain and the encrypted image. In the plain image, it is clear that the pixels are grouped together diagonally, indicating a strong correlation between neighbouring pixels. In the encrypted image, the pixels are spread over the entire area, indicating a very weak correlation between them.

5.2.3 Information entropy analysis

Information entropy is a fundamental concept in information theory. It indicates the amount of information or uncertainty in a random variable. It is widely used in image encryption evaluation, as it is a quantitative measure that evaluates the randomness or unpredictability of an encrypted image.

Entropy measures how difficult it would be to predict the value of any given pixel in an encrypted image. A perfectly random image would have a maximum entropy value of log2(N), where N is the number of possible pixel values.

A well-designed encryption algorithm should result in an encrypted image with an entropy value close to the maximum possible, A higher entropy value indicates that the pixel values are effectively randomized and unpredictable, a desirable property for a secure encryption algorithm.

The effectiveness of an encryption algorithm is typically assessed by its information entropy, which measures the randomness of the information it contains [96]. Information entropy is used to analyze the distribution of pixel values in an image. This analysis demonstrates that the more uniform distributed the pixel values are, the higher the information entropy.

The calculation for information entropy is as follows:

$$H(s) = -\sum_{i=0}^{2^n-1} p(s_i) \log_2 p(s_i) \tag{5.5}$$

Here s_i denotes the grey level and $p(s_i)$ denotes the probability of occurrence of s_i and n denotes the number of bits in each pixel. The ideal information entropy of a $n-bit$ random image is n, where the image would not contain any useful information for attackers.

Table 5.3 shows the entropy values for the different experimental images and their keyed images: Panda (160 x 160). The results show that the entropy of all encrypted images is above 7.99. Therefore, we can say that these values are very close to the optimal value (8), so that entropy attack is not possible.

Key image	2[4]*Original Image	**panda160**	**MRI**
File		Encrypted Image	
Aerial	7.7357	7.9989	7.9984
Baboon	7.2316	7.9971	7.9976
Barbara	7.1674	7.9990	7.9992
Cameraman	7.1048	7.9971	7.9972
Circuit	7.2069	7.9974	7.9976
Lena	7.2283	7.9974	**7.9977**
MRI	6.7863	7.9972	7.9974
Panda	7.5085	7.9991	7.9992
Lena512	7.4451	7.9993	**7.9993**
Boat	7.1479	7.9972	7.9971
Peppers	7.5925	7.9993	7.9992

Table 5.3: Information entropy test results for standard images (Key images:panda160 ; MRI)

5.3 Encryption quality analysis

The more the resulting image differs from the original image, the better the encryption algorithm. We can use the MSE and PSNR values to calculate the difference between the two images. PSNR is the ratio of the peak power of the signal to the noise power. It is measured for image quality. For a good scrambled image, the PSNR value must be low. The lower the PSNR value, the greater the difference between the original image and the encrypted image.

5.3.1 Mean Squared Error (MSE)

Mean Squared Error (MSE) is a widely used metric in image processing, specifically in the field of image encryption, used to evaluate the difference between the

2[4]***Scheme**	**Image File**	**Entropy**
	Ideal value	$\simeq 8$
5[2]*Our scheme	Lena	7.9977
	Baboon	7.9976
	Barbara	7.9992
	Cameraman	7.9972
	Pepper	7.9992
4[2]*Ref [41]	Lena	7.9938
	Baboon	7.9852
	Cameraman	7.9846
	Pepper	7.9956
3[2]*Ref [51]	Lena	7.9973
	Barbara	7.9974
	Pepper	7.9974
3[2]*Ref [5]	Lena	7.9970
	Barbara	7.9970
	Peppers	7.9969

Table 5.4: Comparison of the entropy value between proposed scheme and other methods

original image and the encrypted one. The metric calculates the average squared difference between the two images, which quantifies the overall distortion caused by the encryption process. A lower mean squared error (MSE) value indicates that the encrypted image is closer to the original image. The calculation involves summing the squared differences between the two values and then dividing by the number of observations. It is performed using the following formula :

$$MSE = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(O\left(i,j\right) - D\left(i,j\right)\right)^2}{M \times N} \tag{5.6}$$

Here, N represents the total number of pixels in the image. The parameters O and D refer to the original image and the decrypted image, respectively. In addition, (i, j) stands for the position of the pixels.

5.3.2 Peak Signal to Noise Ratio (PSNR)

Peak Signal to Noise Ratio (PSNR) is a quantitative measure of the difference between original and encrypted images. it is a crucial metric for evaluating the quality of an encrypted image compared to the original image. It is defined as the ratio of the maximum possible power of a signal to the power of corrupting noise. For image encryption analysis, the original input image is the signal, and the noise is the error introduced by encryption.

PSNR measures the similarity between the original and encrypted images based

on the mean squared error (MSE). A higher PSNR value indicates a higher similarity and potentially weaker encryption. The more the resulting image differs from the original image, the better the encryption algorithm. The value of PSNR is usually expressed in decibels (dB) and calculated using the formula:

$$PSNR = 10\log_{10}\left[\frac{MAX^2}{MSE}\right] \tag{5.7}$$

where MAX is the maximum possible pixel value (e.g., 255 for an 8-bit image) and MSE is the mean squared error.

It's important to note that MSE and PSNR don't directly measure encryption strength. Instead, they are used to evaluate the similarity or difference between the original image and the encrypted image. A lower mean squared error (MSE) or a higher peak signal-to-noise ratio (PSNR) value indicates a stronger similarity between the two images, which might not be ideal in an encryption context.

In the Table 5.5 we find the PSNR and MSE measurements of the different test images with Panda as the key image. The obtained results show high values for the MSE and low values for the PSNR measurements (< 10), which proves that the proposed algorithm is very good.

File	NPCR	UACI	MSE	PSNR	CC
Aerial	99.6082	33.3858	17565.9427	8.7651	-0.0014
Baboon	99.5667	33.5354	6920.1784	9.7296	**-0.0001**
Barbara	99.5921	33.4642	28007.6751	9.0637	-0.0034
Boat	**99.6080**	**33.4643**	13372.4535	7.0034	0.0013
Cameraman	**99.6201**	**33.4591**	9389.1378	8.4045	0.0033
Circuit	99.6022	33.4982	11923.4389	8.0193	-0.0017
Lena 256x256	**99.6185**	**33.4671**	**8333.3851**	**8.9226**	**0.0005**
Lena 512x512	99.6223	33.3823	31078.8827	9.2267	-0.0020
Panda	99.6222	33.5107	40433.6016	7.1348	0.0004
Peppers	99.6403	33.5468	33667.8582	8.8792	0.0009

Table 5.5: NPCR, UACI, MSE, PSNR and CC measurements of the different test images

5.4 Differential Attack Analysis

When designing an encryption algorithm, it is important to ensure that it is resistant to differential attacks. To test this, we conduct a differential attack analysis to confirm that even a small change in the unencrypted image, such as a modification in the value of one or more pixels, leads to a significant difference in the encrypted image. To assess an algorithm's resistance to this type of attack, we use two metrics: the Number of Pixel Change Rate (NPCR), which measures the percentage

of different pixels between two images, and the Average Uniform Change Intensity (UACI), which calculates the average change in intensity between the original and encrypted images.

5.4.1 Number of Pixel Change Rate (NPCR)

The Number of Pixel Change Rate (NPCR) is a metric used in image encryption to evaluate the randomness of encrypted images and the algorithm's resistance to differential attacks. It calculates the ratio of different pixels between the original and encrypted images, indicating the sensitivity of the encryption to changes in the plain image. A higher NPCR score suggests a stronger resistance to differential attacks, where a small change in the plain image should result in significant changes in the encrypted image. that is, the encryption algorithm will be more sensitive to changes in the input image, resulting in a larger number of pixel changes. Thus, encrypted images are more secure, because they exhibit a greater degree of randomness and are less susceptible to statistical attacks.

NPCR is often used in conjunction with other metrics, such as Unified Average Changing Intensity (UACI), to provide a more comprehensive evaluation of the encryption algorithm's resistance to differential attacks. it is calculated as follows :

$$NPCR = \frac{1}{W \times H}\left[\sum_{i=0}^{H}\sum_{j=0}^{W} D(i,j) \times 100\%\right] \tag{5.8}$$

Where

$$D(i,j) = \begin{cases} 0; & C_1(i,j) = C_2(i,j) \\ 1; & C_1(i,j) \neq C_2(i,j) \end{cases} \tag{5.9}$$

In general, NPCR values greater than 99% are acceptable for many applications; the ideal value is around 99.6094.

5.4.2 Uniform Average Variable Intensity (UACI)

The Uniform Average Variable Intensity (UACI) is a measure used to quantify the average change in intensity between the original and encrypted images. UACI calculates the average intensity change between two encrypted images (ciphertext images) in situations where there is minimal alteration between the corresponding original images (plaintext images). UACI is frequently used in conjunction with NPCR to determine the sensitivity of the encrypted image to changes in the original image and encryption key. The optimal value for UACI is 33.46%.

Key Image	Aerial (364 x 366)		MRI (512 x 512)		Panda (256 x 256)	
Modified image	NPCR	UACI	NPCR	UACI	NPCR	UACI
Lena (1,1,1)	99.6277	33.6381	99.6078	33.4785	99.5987	33.5194
Lena (1,1,8)	99.5987	33.4355	99.6140	33.4405	99.6078	33.6276
Lena (15,245)	**99.6094**	**33.5135**	**99.6460**	**33.4601**	**99.6063**	**33.4676**
Lena (111,111)	99.6155	33.5406	99.6078	33.4789	99.5972	33.5478
Lena (133,31)	99.6094	33.5014	99.6231	33.3387	99.5865	33.5266
Lena (190,231)	99.5880	33.4394	99.6155	33.3777	99.6368	33.5980
Lena (204,211)	99.6140	33.5523	99.6063	33.2911	99.6307	33.5693
Lena (255,45)	99.6490	33.5010	99.5804	33.5775	99.6094	33.4569

Table 5.6: NPCR and UACI tests results for cipher Lena image

The formula to calculate UACI is :

$$UACI = \frac{1}{W \times H}\left[\sum_{i=0}^{H}\sum_{j=0}^{W}\frac{C_1(i,j) - C_2(i,j)}{255} \times 100\%\right] \tag{5.10}$$

Where C_1 is the encrypted image with the original secret key,and C_2 is the encrypted images with mismatched key.

To compute these values, the selected image is first encrypted using the proposed encryption algorithm. Then, the same image is encrypted after changing only one bit.For this purpose, modified images were prepared by changing only one bit (the most significant bit (MSB) of randomly selected pixels at position (x, y)). For the image Lena seven modified images were prepared, these modified images are Lena (1,1), Lena (15,245), Lena (111,111), Lena (133,31), Lena (190,231), Lena (204,211), Lena (255,45), As for the rest of the test images the bit was changed from the first pixel. Table 5.6 shows the NCPR and UACI measurements of the Lena image and the prepared images. The obtained results are greater than 99.6094% for NCPR and 33.4635% for UACI, which are considered ideal values in the literature. These values also differ significantly between the modified images, which is further evidence that the proposed algorithm is affected by a very small change in the image and is very sensitive to any change. Therefore, the proposed algorithm is resistant to differential attacks and nullifies these attacks.

To test key sensitivity. We encrypt the same plain image (lena) using four different key images (panda and three copies of it), the difference between each of them is only one bit. Then we try to decrypt the encrypted images with the wrong key image (Among the previous). The results (shown in Figure 5.5) confirm that the decryption of the image is not possible without the exact key image that have been encrypted with it.

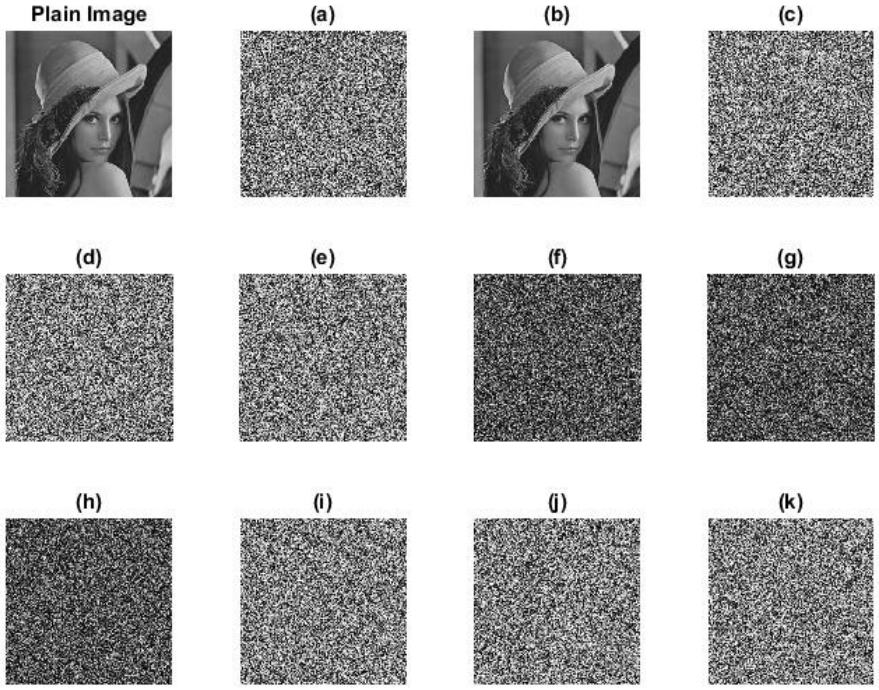


Figure 5.5: Key sensitivity results. (a) The encrypted image using original Key image.(b) The decrypted image (a) using the correct key . (c - e) Encrypted images using modified Key images. (f) The image difference $|a-c|$ (g) The image difference $|a-d|$ (h) The image difference $|a-e|$ (i - k) The decrypted image (a) using wrong key images.

5.5 The Exhaustive Attack Analysis

5.5.1 Key space analysis

In the field of cryptography, The term "Key Space" refers to the total number of possible keys that can be used for an algorithm. It has a direct influence on the difficulty of brute-force attacks, making it an essential component of strong encryption. The difficulty of brute-force attacks, in which the attackers attempt to guess the correct key by trying all possible combinations, significantly increases when the key space is greater. For a robust image encryption algorithm to be able to resist brute force attacks, it is required that the key space is more than 2^{100} [17].

The chaotic maps used in our method have six parameters and initial conditions ($[x1_0, \beta]$ in the improved logistic map; $[x2_0, r]$ in the Logistic-May system; and $[x3_0, \alpha]$ in the Improved Sine Map), all 6 key parameters are floating point numbers. According to the IEEE floating point standard, the calculation accuracy of the 64-

2[1]***Scheme**	**Image File**	**NPCR (%)**	**UACI (%)**
	Ideal value	$\simeq 99.6094$	$\simeq 33.4635$
5[2]*Our scheme	Lena	99.6059	33.4889
	Baboon	99.5819	33.4659
	Barbara	99.6106	33.4515
	Cameraman	99.6063	33.5454
	Pepper	99.5972	33.4844
4[2]*Ref [41]	Lena	99.6006	34.6379
	Baboon	99.6048	32.9759
	Cameraman	99.6521	34.5351
	Pepper	99.6078	32.9312
3[2]*Ref [46]	Lena	99.6048	33.3737
	Boat	99.6159	33.3825
	Cameraman	99.6033	33.3902
3[2]*Ref [38]	Lena	99.707	34.7772
	Baboon	99.7849	36.1264
	Peppers	99.7101	34.4092
4[2]*Ref [60]	Lena	99.6032	33.5986
	Barbara	99.6118	33.4142
	Cameraman	99.6112	33.5076
	Peppers	99.6561	33.4312
2[2]*Ref [51]	Lena	99.6152	28.618
	Pepper	99.6105	33.4581

Table 5.7: Comparison of the NPCR and UACI values between our proposed approach and the other methods

Scheme	Our scheme	[64]	[96]	[17]	[41]	[40]
Key space size	10^{704}	10^{70}	2^{279}	2^{262}	10^{102}	10^{112}

Table 5.8: Key space size comparison.

bit number with double precision is about 10^{15} [37]. In addition to maps, we also use the key image in the key generation process. If we consider that the size of the smallest image that can be used is (16×16). Thus, the total number of possible secret keys is greater than:

$$Ks \geq (10^{15})^6 \times (2^{8\times16\times16}) = 10^{90} \times (2^{2048}) = 2^{300+2048} = 2^{2348} = 10^{704}$$

Our proposed work has a key space large enough to resist all forms of brute force attacks. Table 5.8 compares the size of our key space with that of other works. Our proposed image encryption method has a larger key space compared to some other works.

5.5.2 Key sensitivity

Key sensitivity evaluates the strength and robustness of the encryption algorithm. Whereas, a very sensitive algorithm produces completely different ciphertexts for minor changes to the keys. Key Sensitivity Analysis measures the encryption image's sensitivity to small changes in the encryption key. it directly affects the security of the encryption process, which is why a strong encryption algorithm must produce completely different encrypted images, making the original image unrecoverable without exactly the right key.

5.5.3 The effect of the key image on the proposed algorithm

To find out the effect of selecting a key image on the results of the proposed algorithm, we performed encryption operations on the experimental images using a different key image each time. Table 5.9 shows the results of the test on Lena image,it is clear that there is a difference between the results from one key image to another. This is evidence of the impact of choosing the key image on the results of the proposed algorithm.

Key image	Dimensions	Entropy	NPCR	UACI	MSE	PSNR	CC
Aerial	364 x 366	7.9970	99.6094	33.5135	8385.4127	8.8956	-0.0017
Baboon	256 x 256	7.9975	99.6185	33.4360	8321.1509	8.9290	-0.0007
Barbara	402 x 560	7.9968	99.5865	33.3774	8332.7596	8.9229	-0.0015
Boat	256 x 256	7.9970	99.6429	33.4643	8353.5777	8.9121	0.0030
Cameraman	256 x 256	7.9969	**99.6094**	**33.4704**	8349.8765	8.9140	**-0.0005**
Circuit	256 x 256	7.9973	**99.6338**	**33.4874**	8343.5332	8.9173	0.0061
Lena	256 x 256	7.9969	99.6231	33.4127	8378.1769	8.8993	0.0035
Lena	512 x 512	7.9971	99.6033	33.4110	8330.1288	8.9243	0.0005
Lena	220 x 220	7.9973	99.6277	33.3727	8330.8691	8.9239	**-0.0001**
MRI	256 x 256	**7.9977**	99.6567	33.4431	8298.6701	8.9407	0.0058
MRI	512 x 512	7.9973	**99.6460**	**33.4601**	8411.5670	8.8820	-0.0077
Panda	459 x 459	7.9970	99.6048	33.4281	8333.3851	8.9226	0.0005
Panda	160 x 160	7.9974	99.6170	33.4375	8364.5892	8.9064	0.0014
Panda	256 x 256	7.9970	**99.6063**	**33.4676**	8366.1967	8.9055	0.0029
Onion	198 x 135	7.9965	99.6429	33.5234	8395.8145	8.8902	-0.0019
Football	320 x 256	7.9973	99.6201	33.4464	8406.4602	8.8847	-0.0084
Peppers	512 x 512	7.9969	99.6201	33.5174	8357.9740	8.9098	-0.0002

Table 5.9: The effect of the key image

5.6 Randomness tests

To check for possible vulnerabilities in the design of the proposed encryption algorithm, we use the National Institute of Standards and Technology (NIST) statistical

test suite (SP 800-22) [75]. The NIST suite test requires a binary sequence of at least 10^6 bits and the results of the test provide P values that must be greater than 0.01 for the test to be considered valid [75].

In our experiments, we use the default values provided by the NIST tests. We passed the encrypted image to verify that the encrypted images have random properties and no attacker can distinguish them from a random source. The results are listed shown in Table 5.10. Based on the results, we can confidently say that the encrypted images have excellent random statistical properties and pass all NIST tests.

Statical test	**P-Values**	**Results**
Frequency Test (Monobit)	0.9098	Passed
Frequency Test (Block)	0.9330	Passed
Run Test	0.1233	Passed
Longest Run of Ones in a Block	0.4692	Passed
Binary Matrix Rank Test	0.2881	Passed
Discrete Fourier Transform (Spectral) Test	0.7533	Passed
Non-Overlapping Template Matching Test	0.7238	Passed
Overlapping Template Matching Test	0.2771	Passed
Universal Statistical test	0.3391	Passed
Linear Complexity Test	0.1614	Passed
Serial test	0.8398,0.4244	Passed
Approximate Entropy Test	0.4375	Passed
Cummulative Sums (Forward) Test	0.7697	Passed
Cummulative Sums (Reverse) Test	0.8694	Passed
Random Excursions Test (x = -4)	0.6908	Passed
Random Excursions Variant Test (x = -3)	0.7476	Passed

Table 5.10: NIST randomness test.

5.7 The Analysis of Known/chosen attack

In the proposed scheme, the difference in one bit in the original image or in the key image results in an encrypted image that is completely different from the other. This is due to the use of $ExtraParam$, which is affected by the very small difference that leads to different initial conditions and different control coefficients for all the maps used, resulting in completely different PRNS (proven by the high Lyapunov exponents of the maps used). An intruder will not be able to decrypt an image encrypted with keys generated from other keys or encrypted images. This makes the encryption system secure against an attack with chosen/known plaintext [41].

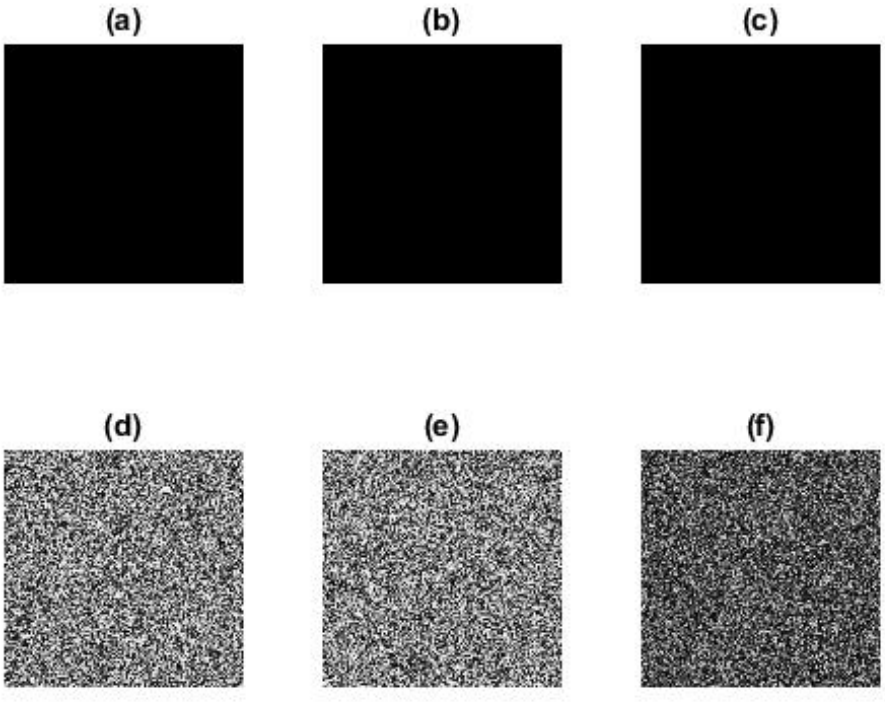


Figure 5.6: Plain-image sensitivity results. (a) The plain image I.(b) The modified image J. (c) The image difference $|I - j|$. (d) The encrypted image CI. (e) The encrypted image CJ. (f) The image difference $|CI - Cj|$

The results of the experiment shown in Figure 5.6 prove that the proposed algorithm is resistant to these types of attacks.

5.8 Overall comparison with encryption algorithms

We aim in this subsection to compare our work with with some recent chaotic-based image encryption algorithms. The 5.2, 5.4, 5.7, 5.8, and 5.11 tables list the performance metrics (keyspace size, information entropy, NPCR, UACI, and correlation coefficients). The comparison analysis indicate that most of the metrics of the proposed algorithm are the best values. This proves that this algorithm has high key sensitivity and can withstand known attacks such as statistical attacks and differential attacks. And it is strongly nominated to take its place in the literature.

2[4]*scheme	2[4]*Entropy	2[4]*NPCR	2[4]*UACI	Correlation coefficients		
				Horizontal	Vertical	Diagonal
Ideal value	$\simeq 8$	> 99.6094	$\simeq 33.4635$	$\simeq 0$	$\simeq 0$	$\simeq 0$
Our scheme	7.9977	99.6567	33.4431	-0.0001	−0.0049	0.0008
Ref [6]	7.9972	99.6800	33.4700	0.0038	0.0036	0.0022
Ref [51]	7.9973	99.6152	28.6180	−0.0002	−0.0015	−0.0008
Ref [46]	7.9972	99.6048	33.3737	−0.0034	0.0011	−0.0012
Ref [41]	7.9938	99.6006	34.6379	−0.0006	−0.0057	0.0009

Table 5.11: Security performance comparisons with other schemes

5.9 Conclusion

The results of all our experiments on various performance metrics, including histogram and entropy analysis, the number of pixel change rate (NPCR), the unified average changing intensity (UACI), the mean square error (MSE), the peak signal-to-noise ratio (PSNR), and key space evaluation, demonstrated that the proposed algorithm is not only powerful but also remarkably robust and highly effective in its application. These metrics provide a comprehensive assessment of the algorithm's performance, revealing its ability to maintain high levels of security while ensuring efficient processing.

In particular, the histogram and entropy analysis indicated that the algorithm produces outputs with a uniform distribution, which is a key characteristic of secure cryptographic systems. The NPCR and UACI metrics further confirmed that even minor changes in the input lead to significant variations in the output, highlighting the algorithm's sensitivity and unpredictability—essential traits for effective encryption.

Additionally, the mean square error (MSE) and peak signal-to-noise ratio (PSNR) assessments illustrated the algorithm's capacity to preserve the quality of the original data while implementing cryptographic transformations. The analysis of key space revealed a vast range of potential keys, contributing to the overall strength of the algorithm against brute-force attacks.

Moreover, the algorithm exhibits excellent cryptographic properties, including pseudo-random behavior, which enhances its security by making patterns difficult to detect. Its resistance to various types of attacks—such as differential and linear cryptanalysis—further underscores its reliability in safeguarding sensitive information.

This chapter focuses specifically on these critical aspects of the proposed algorithm, delving into the methodologies used to evaluate its performance and the implications of the findings for future applications in secure communication and data protection.

Conclusion and perspectives

In conclusion, This dissertation introduces a groundbreaking image encryption algorithm leveraging the robust security properties of hyperchaotic systems. The proposed method departs from traditional approaches by incorporating a key image for a dual purpose: initializing the pseudorandom generator and acting as a mask during the diffusion phase. This innovative integration enhances the algorithm's resistance to cryptanalysis. Furthermore, it strategically utilizes a parameter extracted from the original image, termed "ExtraParam," due to its inherent sensitivity to even minor bit changes during the initialization of the chaotic maps. This sensitivity translates to a significant amplification effect on any modifications within the image data, further bolstering the encryption strength.

The algorithm's core functionality hinges on a two-phase process: confusion and diffusion. The confusion phase meticulously scrambles the pixel positions within the image, effectively disrupting the original spatial relationship between pixels. The subsequent diffusion phase modifies the actual pixel values themselves, further obfuscating the image content. This meticulously designed two-pronged approach demonstrably transforms the original image (plaintext) into a securely encrypted form (ciphertext).

Rigorous testing methodologies were employed to evaluate the algorithm's robustness against a comprehensive suite of known attacks. The vast keyspace generated by the algorithm effectively renders brute-force attacks computationally infeasible. The successful completion of these cryptanalysis validations, coupled with the demonstrably high level of security achieved, positions this method as a highly secure solution for real-world applications across various industries. Its potential applications range from securing sensitive medical imagery in telemedicine to safeguarding confidential financial data during electronic transactions.

In this dissertation, several avenues for future research and development are identified. These perspectives aim to build upon the current work and address potential improvements and expansions in the field of image encryption. Below are the key perspectives:

- Develop a method for color image Encryption: Extend the current encryption techniques to effectively handle color images, ensuring that the method can maintain high security and efficiency while accommodating the additional

complexity introduced by color information.

- Enhance the calculation method of ExtraParam: Refine the algorithm used to compute ExtraParam, aiming for greater accuracy and efficiency. This enhancement could involve optimizing the mathematical processes or leveraging advanced computational techniques to reduce computation time and resource usage.

- Optimize the proposed method in terms of execution time and file size: Focus on improving the encryption method's performance by reducing the execution time and minimizing the file size of the encrypted images. This could involve algorithmic optimizations, parallel processing, or more efficient data structures.

- Explore applications in Real-time systems: Investigate the applicability of the encryption method in real-time systems where quick and secure image processing is crucial, such as video streaming, live surveillance, and telemedicine.

- Implement robustness against various attacks: Strengthen the encryption method to resist different types of cyber-attacks, including brute force, statistical, and differential attacks. This might involve integrating additional layers of security or developing adaptive techniques that can detect and respond to potential threats.

- Evaluate performance on diverse image datasets: Extend the evaluation of the encryption method across a wider range of image datasets, including those with varying resolutions, formats, and content types. This will help in understanding the method's generalizability and robustness.

- Incorporate machine learning techniques: Explore integrating machine learning approaches to enhance the encryption process. For example, using machine learning models to predict and optimize certain parameters could lead to more efficient and secure encryption methods.

- Develop a user-friendly encryption software: Create a software tool that implements the proposed encryption method, making it accessible and easy to use for practitioners and researchers. This tool could include a graphical user interface (GUI) and support for various image formats and encryption settings.

These perspectives provide a roadmap for future work that can enhance this dissertation's contributions and push the boundaries of image encryption research.

Bibliography

[1] Mohammed Abutaha, Islam Amar, and Salman AlQahtani, "Parallel and Practical Approach of Efficient Image Chaotic Encryption Based on Message Passing Interface (MPI)", *in*: *Entropy* 24.*4* (2022), p. 566.

[2] Moatsum Alawida, "A Novel Image Encryption Algorithm Based on Cyclic Chaotic Map in Industrial IoT Environments", *in*: *IEEE Transactions on Industrial Informatics* (2024).

[3] Moatsum Alawida, Azman Samsudin, and Wafa'Hamdan Alshoura, "Enhancing one-dimensional chaotic map based on bitstream dividing model", *in*: *Proceedings of the 2019 8th International Conference on Software and Computer Applications*, 2019, pp. 130–134.

[4] Khawaja Muhammad Ali and Majid Khan, "Application based construction and optimization of substitution boxes over 2D mixed chaotic maps", *in*: *International Journal of Theoretical Physics* 58 (2019), pp. 3091–3117.

[5] Souyah Amina and Faraoun Kamel Mohamed, "An efficient and secure chaotic cipher algorithm for image content preservation", *in*: *Communications in Nonlinear Science and Numerical Simulation* 60 (2018), pp. 12–32.

[6] Rengarajan Amirtharajan et al., "A robust medical image encryption in dual domain: chaos-DNA-IWT combined approach", *in*: *Medical & biological engineering & computing* 58.*7* (2020), pp. 1445–1458.

[7] Alireza Arab, Mohammad Javad Rostami, and Behnam Ghavami, "An image encryption method based on chaos system and AES algorithm", *in*: *The Journal of Supercomputing* 75.*10* (2019), pp. 6663–6682.

[8] Minal Govind Avasare and Vishakha Vivek Kelkar, "Image encryption using chaos theory", *in*: *2015 International Conference on Communication, Information & Computing Technology (ICCICT)*, IEEE, 2015, pp. 1–6.

[9] O El-Basha, A Fayed El-Shahat, and H Fayed, "Chaos Theory and Lorenz Attractors", *in*: *Sohag Journal of Sciences,\textbf {(1)(1)}* (2016).

[10] Nabil Ben Slimane et al., "A novel chaotic image cryptosystem based on DNA sequence operations and single neuron model", *in*: *Multimedia Tools and Applications* 77 (2018), pp. 30993–31019.

[11] Sellami Benaissi, Noureddine Chikouche, and Rafik Hamza, "A novel image encryption algorithm based on hybrid chaotic maps using a key image", *in*: *Optik* 272 (2023), p. 170316.

[12] A Hadj Brahim, A Ali Pacha, and N Hadj Said, "Image encryption based on compressive sensing and chaos systems", *in*: *Optics & Laser Technology* 132 (2020), p. 106489.

[13] Wilhelm Burger and Mark J Burge, *Digital image processing: An algorithmic introduction*, Springer Nature, 2022.

[14] A Buscarino, L Fortuna, and M Frasca, "Experimental robust synchronization of hyperchaotic circuits", *in*: *Physica D: Nonlinear Phenomena* 238.*18* (2009), pp. 1917–1922.

[15] Guanrong Chen, Yaobin Mao, and Charles K Chui, "A symmetric image encryption scheme based on 3D chaotic cat maps", *in*: *Chaos, Solitons & Fractals* 21.*3* (2004), pp. 749–761.

[16] Hang Chen et al., "Color image encryption based on the affine transform and gyrator transform", *in*: *Optics and Lasers in Engineering* 51.*6* (2013), pp. 768–775.

[17] Zhuozhao Chen and Guodong Ye, "An asymmetric image encryption scheme based on hash SHA-3, RSA and compressive sensing", *in*: *Optik* (2022), p. 169676.

[18] Zahraa M Dawood, Mouner Aboud, and Fadhil S Hasan, "Speech encryption using finite precision chaotic maps based stream ciphers", *in*: *Proceedings of the International Conference on Information and Communication Technology*, 2019, pp. 127–133.

[19] Robert L. Devaney, *An introduction to chaotic dynamical systems.* English, 2nd ed., Redwood City, CA etc.: Addison-Wesley Publishing Company, Inc., 1989, ISBN: 0-201-13046-7.

[20] Lina Ding et al., "A new lightweight stream cipher based on chaos", *in*: *Symmetry* 11.*7* (2019), p. 853.

[21] Michael Doebeli and Graeme D Ruxton, "Controlling spatial chaos in metapopulations with long-range dispersal", *in*: *Bulletin of Mathematical Biology* 59.*3* (1997), pp. 497–515.

[22] Behrouz Fathi-Vajargah, Mohadeseh Kanafchian, and Vassil Alexandrov, "Image encryption based on permutation and substitution using Clifford Chaotic System and logistic map", *in*: *Journal of Computers* 13.*3* (2018), pp. 309–326.

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

[24] Borko Furht, Esad Akar, and Whitney Angelica Andrews, *Digital Image Processing: Practical Approach*, Springer, 2018.

[25] Étienne Ghys, "The butterfly effect", *in*: *The Proceedings of the 12th International Congress on Mathematical Education: Intellectual and attitudinal challenges*, Springer International Publishing, 2015, pp. 19–39.

[26] A Hadj Brahim, A Ali Pacha, and N Hadj Said, "An image encryption scheme based on a modified AES algorithm by using a variable S-box", *in*: *Journal of Optics* 53.*2* (2024), pp. 1170–1185.

[27] Basavaraj P Halagali and Veena V Desai, "Implementation of chaos based cryptography in kasumi block cipher", *in*: *2018 International Conference on Communication and Signal Processing (ICCSP)*, IEEE, 2018, pp. 0165–0169.

[28] Rafik Hamza and Faiza Titouna, "A novel sensitive image encryption algorithm based on the Zaslavsky chaotic map", *in*: *Information Security Journal: A Global Perspective* 25.*4-6* (2016), pp. 162–179.

[29] Rafik Hamza et al., "A privacy-preserving cryptosystem for IoT E-healthcare", *in*: *Information Sciences* 527 (2020), pp. 493–510.

[30] Chunyan Han, "An image encryption algorithm based on modified logistic chaotic map", *in*: *Optik* 181 (2019), pp. 779–785.

[31] Michel Hénon, "A two-dimensional mapping with a strange attractor", *in*: *The theory of chaotic attractors*, Springer, 1976, pp. 94–102.

[32] Jeonghwan Heo and Jechang Jeong, "Deceptive techniques to hide a compressed video stream for information security", *in*: *Sensors* 21.*21* (2021), p. 7200.

[33] Lee Mariel Heucheun Yepdia, Alain Tiedeu, and Guillaume Kom, "A Robust and Fast Image Encryption Scheme Based on a Mixing Technique", *in*: *Security and Communication Networks* 2021 (2021).

[34] Guiqiang Hu et al., "Cryptanalysis of a chaotic image cipher using Latin square-based confusion and diffusion", *in*: *Nonlinear Dynamics* 88.*2* (2017), pp. 1305–1316.

[35] Xiaoling Huang et al., "Meaningful image encryption algorithm based on compressive sensing and integer wavelet transform", *in*: *Frontiers of Computer Science* 17.*3* (2023), pp. 1–15.

[36] Saleh Ibrahim and Ayman Alharbi, "Efficient image encryption scheme using Henon map, dynamic S-boxes and elliptic curve cryptography", *in*: *IEEE Access* 8 (2020), pp. 194289–194302.

[37] IEEE, "IEEE Standard for Floating-Point Arithmetic, IEEE Std 754-2019 (Revision of IEEE 754-2008)", *in*: Institute of Electrical and Electronics Engineers New York, 2019.

[38] Aiman Jan, Shabir A Parah, and Bilal A Malik, "IEFHAC: Image encryption framework based on hessenberg transform and chaotic theory for smart health", *in*: *Multimedia Tools and Applications* 81.*13* (2022), pp. 18829–18853.

[39] Sara T Kamal, Mohamed M Darwish, and Khalid M Hosny, "Chaotic Maps for Image Encryption: An Assessment Study", *in*: *Multimedia Security Using Chaotic Maps: Principles and Methodologies*, Springer, 2020, pp. 27–51.

[40] Ahmad Pourjabbar Kari et al., "A new image encryption scheme based on hybrid chaotic maps", *in*: *Multimedia Tools and Applications* 80.*2* (2021), pp. 2753–2772.

[41] Gurpreet Kaur, Rekha Agarwal, and Vinod Patidar, "Chaos based multiple order optical transform for 2D image encryption", *in*: *Engineering Science and Technology, an International Journal* 23.*5* (2020), pp. 998–1014.

[42] M Kaur and VJEL Kumar, "Efficient image encryption method based on improved Lorenz chaotic system", *in*: *Electronics Letters* 54.*9* (2018), pp. 562–564.

[43] Rajandeep Kaur and Pooja Choudhary, "A review of image compression techniques", *in*: *Int. J. Comput. Appl* 142.*1* (2016), pp. 8–11.

[44] Noura Khalil, Amany Sarhan, and Mahmoud AM Alshewimy, "An efficient color/grayscale image encryption scheme based on hybrid chaotic maps", *in*: *Optics & Laser Technology* 143 (2021), p. 107326.

[45] Taqseer Khan and Harindri Chaudhary, "Controlling chaos generated in predator-prey interactions using adaptive hybrid combination synchronization", *in*: *Proceedings of 3rd International Conference on Computing Informatics and Networks*, Springer, 2021, pp. 449–459.

[46] Y Khedmati, R Parvaz, and Y Behroo, "2D Hybrid chaos map for image security transform based on framelet and cellular automata", *in*: *Information Sciences* 512 (2020), pp. 855–879.

[47] Rushi Lan et al., "Integrated chaotic systems for image encryption", *in*: *Signal Processing* 147 (2018), pp. 133–145.

[48] Laurent Larger and Jean-Pierre Goedgebuer, "Encryption using chaotic dynamics for optical telecommunications", *in*: *Comptes Rendus Physique* 5.*6* (2004), pp. 609–611.

[49] Chunhu Li et al., "An image encryption scheme based on chaotic tent map", *in*: *Nonlinear Dynamics* 87.*1* (2017), pp. 127–133.

[50] Lizong Li, "A novel chaotic map application in image encryption algorithm", *in*: *Expert Systems with Applications* (2024), p. 124316.

[51] Xiaofeng Liao, Muntazim Abbas Hahsmi, Rizwan Haider, et al., "An efficient mixed inter-intra pixels substitution at 2bits-level for image encryption technique using DNA and chaos", *in*: *Optik-International Journal for Light and Electron Optics* 153 (2018), pp. 117–134.

[52] Edward N Lorenz, "Deterministic nonperiodic flow", *in*: *Journal of atmospheric sciences* 20.*2* (1963), pp. 130–141.

[53] Gururaj Maddodi et al., "A new image encryption algorithm based on heterogeneous chaotic neural network generator and dna encoding", *in*: *Multimedia Tools and Applications* 77.*19* (2018), pp. 24701–24725.

[54] Robert Matthews, "On the derivation of a "chaotic" encryption algorithm", *in*: *Cryptologia* 13.*1* (1989), pp. 29–42.

[55] Robert M May, "Simple mathematical models with very complicated dynamics", *in*: *The Theory of Chaotic Attractors* (2004), pp. 85–93.

[56] Robert M. May, "Simple mathematical models with very complicated dynamics", English, *in*: *Nature, London* 261.*5560* (1976), pp. 459–467, ISSN: 0028-0836, DOI: 10.1038/261459a0.

[57] Fanhao Meng et al., "A multi-connection encryption algorithm applied in secure channel service system", *in*: *EAI Endorsed Transactions on Security and Safety* 5.*15* (2018), e1–e1.

[58] Robert A Meyers et al., *Encyclopedia of complexity and systems science*, vol. 9, Citeseer, 2009.

[59] Kareem Mostafa and Tarek Hegazy, "Review of image-based analysis and applications in construction", *in*: *Automation in Construction* 122 (2021), p. 103516.

[60] Farhan Musanna and Sanjeev Kumar, "Image encryption using quantum 3-D Baker map and generalized gray code coupled with fractional Chen's chaotic system", *in*: *Quantum Information Processing* 19.*8* (2020), pp. 1–31.

[61] Rasika B Naik and Udayprakash Singh, "A review on applications of chaotic maps in pseudo-random number generators and encryption", *in*: *Annals of Data Science* 11.*1* (2024), pp. 25–50.

[62] Prabir Kumar Naskar et al., "A robust image encryption scheme using chaotic tent map and cellular automata", *in*: *Nonlinear Dynamics* 100.*3* (2020), pp. 2877–2898.

[63] Nashreen Nesa, Tania Ghosh, and Indrajit Banerjee, "Design of a chaos-based encryption scheme for sensor data using a novel logarithmic chaotic map", *in*: *Journal of Information Security and Applications* 47 (2019), pp. 320–328.

[64] Ying Niu, Zheng Zhou, and Xuncai Zhang, "An image encryption approach based on chaotic maps and genetic operations", *in*: *Multimedia Tools and Applications* 79.*35* (2020), pp. 25613–25633.

[65] Yannick Pascal Kamdeu Nkandeu and Alain Tiedeu, "An image encryption algorithm based on substitution technique and chaos mixing", *in*: *Multimedia Tools and Applications* 78.*8* (2019), pp. 10013–10034.

[66] H Oğraş and M Türk, "A secure chaos-based image cryptosystem with an improved sine key generator", *in*: *American Journal of Signal Processing* 6.*3* (2016), pp. 67–76.

[67] Chanil Pak and Lilian Huang, "A new color image encryption using combination of the 1D chaotic map", *in*: *Signal Processing* 138 (2017), pp. 129–137.

[68] Jing Pan, Qun Ding, and Na Qi, "The Research of Chaos-based SMS Encryption in Mobile Phone", *in*: *2012 Second International Conference on Instrumentation, Measurement, Computer, Communication and Control*, IEEE, 2012, pp. 501–504.

[69] Shouquan Pang and Yongjian Liu, "A new hyperchaotic system from the Lü system and its control", *in*: *Journal of Computational and Applied Mathematics* 235.*8* (2011), pp. 2775–2789.

[70] Narendra K Pareek, Vinod Patidar, and Krishan K Sud, "Image encryption using chaotic logistic map", *in*: *Image and vision computing* 24.*9* (2006), pp. 926–934.

[71] Priyansi Parida et al., "Image Encryption and Authentication With Elliptic Curve Cryptography and Multidimensional Chaotic Maps", *in*: *IEEE Access* 9 (2021), pp. 76191–76204.

[72] Ping Ping et al., "A chaos based image encryption scheme using digit-level permutation and block diffusion", *in*: *IEEE Access* 6 (2018), pp. 67581–67593.

[73] Ahmad Pourjabbar Kari et al., "A new image encryption scheme based on hybrid chaotic maps", *in*: *Multimedia Tools and Applications* 80.*2* (2021), pp. 2753–2772.

[74] Arezoo Rajabi et al., "On the (im) practicality of adversarial perturbation for image privacy", *in*: *Proceedings on Privacy Enhancing Technologies* (2021), pp. 85–106.

[75] Andrew L Rukhin et al., "SP 800-22 Rev. 1a. A statistical test suite for random and pseudorandom number generators for cryptographic applications", *in*: *Applied Physics Letters* 22.*7* (2010), pp. 1645–2179.

[76] Banshidhar Sahoo and Swarup Poria, "The chaos and control of a food chain model supplying additional food to top-predator", *in*: *Chaos, Solitons & Fractals* 58 (2014), pp. 52–64.

[77] Nedhal AM Al-Saiyd, "Hybrid Medical Colored Image LSB Steganography Based on Primitive Root Numbers", *in*: *International Journal of Computer Science and Network Security (IJCSNS)* 17.*2* (2017), p. 206.

[78] N Sasikaladevi et al., "H3-hybrid multilayered hyper chaotic hyper elliptic curve based image encryption system", *in*: *Optics & Laser Technology* 127 (2020), p. 106173.

[79] SJ Sheela, KV Suresh, and Deepaknath Tandur, "Image encryption based on modified Henon map using hybrid chaotic shift transform", *in*: *Multimedia Tools and Applications* 77.*19* (2018), pp. 25223–25251.

[80] Zhiguo Shi, Shaohua Hong, and Kangsheng Chen, "Experimental study on tracking the state of analog Chua's circuit with particle filter for chaos synchronization", *in*: *Physics Letters A* 372.*34* (2008), pp. 5575–5580.

[81] Christos H Skiadas and Charilaos Skiadas, *Chaotic modelling and simulation: analysis of chaotic models, attractors and forms*, Chapman and Hall/CRC, 2008.

[82] Ali Soleymani and Md Jan Nordin, "Selective Image Encryption Based On Chaotic Maps And Elliptic Curve Cryptography", *in*: (2021).

[83] Douglas Robert Stinson and Maura Paterson, *Cryptography: theory and practice*, CRC press, 2018.

[84] Mohamed Zakariya Talhaoui and Xingyuan Wang, "A new fractional one dimensional chaotic map and its application in high-speed image encryption", *in*: *Information Sciences* 550 (2021), pp. 13–26.

[85] Meng Tang et al., "A hyperchaotic image encryption scheme based on the triple dislocation of the Liu and Lorenz system", *in*: *Optik* 261 (2022), p. 169133.

[86] Yongli Tang, Mingjie Zhao, and Lixiang Li, "Secure and Efficient Image Compression-Encryption Scheme Using New Chaotic Structure and Compressive Sensing", *in*: *Security and Communication Networks* 2020.*1* (2020), p. 6665702.

[87] Je Sen Teh and Azman Samsudin, "A chaos-based authenticated cipher with associated data", *in*: *Security and Communication Networks* 2017 (2017).

[88] Roy Tenny et al., "Using distributed nonlinear dynamics for public key encryption", *in*: *Physical review letters* 90.*4* (2003), p. 047903.

[89] Sundarapandian Vaidyanathan, "Adaptive backstepping control of enzymes-substrates system with ferroelectric behaviour in brain waves", *in*: *International Journal of PharmTech Research* 8.*2* (2015), pp. 256–261.

[90] Milad Yousefi Valandar, Milad Jafari Barani, and Peyman Ayubi, "A fast color image encryption technique based on three dimensional chaotic map", *in*: *Optik* 193 (2019), p. 162921.

[91] Gerard Vidal, Murilo S Baptista, and Hector Mancini, "A fast and light stream cipher for smartphones", *in*: *The European Physical Journal Special Topics* 223.*8* (2014), pp. 1601–1610.

[92] Manuel Villegas et al., "Application of the polynomial chaos expansion to the simulation of chemical reactors with uncertainties", *in*: *Mathematics and Computers in Simulation* 82.*5* (2012), pp. 805–817.

[93] Yong Wang et al., "A chaos-based image encryption algorithm with variable control parameters", *in*: *Chaos, Solitons & Fractals* 41.*4* (2009), pp. 1773–1783.

[94] Wenying Wen et al., "Differential attack on a hyper-chaos-based image cryptosystem with a classic bi-modular architecture", *in*: *Nonlinear Dynamics* 87.*1* (2017), pp. 383–390.

[95] Lu Xu et al., "A novel bit-level image encryption algorithm based on chaotic maps", *in*: *Optics and Lasers in Engineering* 78 (2016), pp. 17–25.

[96] Amina Yahi et al., "A color image encryption scheme based on 1D cubic map", *in*: *Optik* 249 (2022), p. 168290.

[97] Guodong Ye, "Image scrambling encryption algorithm of pixel bit based on chaos map", *in*: *Pattern Recognition Letters* 31.*5* (2010), pp. 347–354.

[98] Guodong Ye et al., "A novel multi-image visually meaningful encryption algorithm based on compressive sensing and Schur decomposition", *in*: *Transactions on Emerging Telecommunications Technologies* 32.*2* (2021), e4071.

[99] Guodong Ye et al., "An image encryption scheme based on public key cryptosystem and quantum logistic map", *in*: *Scientific Reports* 10.*1* (2020), pp. 1–19.

[100] Guodong Ye et al., "Image encryption scheme based on blind signature and an improved Lorenz system", *in*: *Expert Systems with Applications* (2022), p. 117709.

[101] Wei-Zhu Yeoh, Je Sen Teh, and Huey Rong Chern, "A parallelizable chaos-based true random number generator based on mobile device cameras for the Android platform", *in*: *Multimedia Tools and Applications* 78.*12* (2019), pp. 15929–15949.

[102] Fei Yu et al., "A survey on true random number generators based on chaos", *in*: *Discrete Dynamics in Nature and Society* 2019 (2019).

[103] Guohui Yuan, Xin Zhang, and Zhuoran Wang, "Generation and synchronization of feedback-induced chaos in semiconductor ring lasers by injection-locking", *in*: *Optik* 125.*8* (2014), pp. 1950–1953.

[104] Leo Yu Zhang et al., "On the security of a class of diffusion mechanisms for image encryption", *in*: *IEEE transactions on cybernetics* 48.*4* (2017), pp. 1163–1175.

[105] Ying-Qian Zhang and Xing-Yuan Wang, "A symmetric image encryption algorithm based on mixed linear–nonlinear coupled map lattice", *in*: *Information Sciences* 273 (2014), pp. 329–351.

[106] Yong Zhang, "Cryptanalysis of a novel image fusion encryption algorithm based on DNA sequence operation and hyper-chaotic system", *in*: *Optik* 126.*2* (2015), pp. 223–229.

[107] Lena C Zuchowski, "Disentangling complexity from randomness and chaos", *in*: *Entropy* 14.*2* (2012), pp. 177–212.