N^0 d'ordre...../FT/UMBB/2024

PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

UNIVERSITY M'HAMED BOUGARA OF BOUMERDES

Faculty of Technology

Doctoral Thesis

Presented by:

Mrs Hadjer BOUREKOUCHE

Submitted for the Fulfillment of the Requirements of the **Doctorate-LMD** degree in

Field: Electronic

Specialty: Electronics of Embedded Systems

Contribution to chaotic encryption methods for digital data

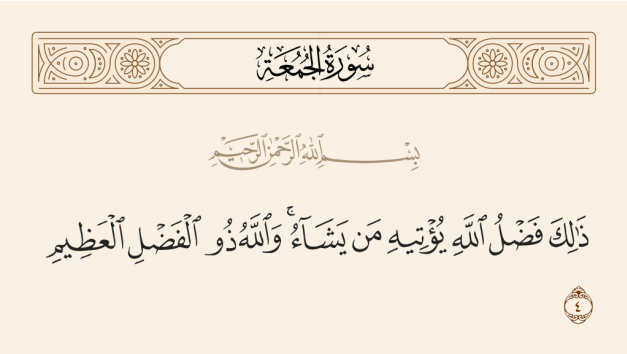
In front of a jury composed of:

Pr. M'Hamed	HAMADOUCHE	Professor	Univ. Boumerdes	President
Dr. Samia	BELKACEM	MCA	Univ. Boumerdes	Supervisor
Pr. Khaled	ROUABAH	Professor	Univ. M'Sila	Examiner
Pr. Yassine	MERAIHI	Professor	Univ. Boumerdes	Examiner
Dr. Abdenour	HACINE GHARBI	MCA	Univ. Bordj Bou Arreridj	Examiner
Dr. Yasmine	GUERBAI	MCA	Univ. Boumerdes	Examiner

Academic year:2023/2024

Acknowledgements

First and foremost, I would like to thank Allah for guiding me through all the difficulties and challenges. Without God's grace i would never reach this stage.



Second, I want to convey my profound gratitude to my supervisor, Dr. Samia BELKACEM, for her guidance, tolerance, motivation, leadership, and support throughout this entire work. Your advice and critical feedback have always encouraged and pushed me to pursue PhD truck, despite your multiple occupations. Thank you very much.

I would like to thank also all the members of the thesis jury particularly Professor HAMADOUCHE M'Hamed who served as the chairman, Professor ROUABAH Khaled, Professor MERAIHI Yassine, Dr HACINE GHARBI Abdenour, and Dr GUERBAI Yasmine. Your insightful comments, constructive feedback, and valuable suggestions have been instrumental in shaping the final outcome of this work.

I am deeply grateful to Dr. Noureddine MESSAOUDI, the head of Electrical Systems Engineering department and the Chairman of Doctoral contest for the academic year 2020/2021, for his encouragement, support, and for his invaluable help in completing the necessary administrative papers.

Finally, I must point out that the staff of my university have spared no effort to provide their help and assistance to allow me to progress normally in the search for quality in what I was undertaking and efficiency in my steps. May all these illustrious personalities find here the expression of my sincere gratitude and the testimony of my deep respect.

Dedication

To my father, who has always provided unending encouragement, motivation, and support. I am grateful for all your sacrifices that have helped me to become who I am today. I'm very happy to make you proud of me at that day.

To my mother, brother Abderahmane, sisters Amina & Hanane, for their enduring support throughout the years.

To my husband Chouaib, thank you for being with me during the hardest times. Thank you for your encouragement. Thank you for believing in me.

To my future son, my baby Nour Abdelhak, sorry for making you feeling all the pressure of this graduation period while you were still a fetus of five months in my womb. Be sure that you will be proud of your mother one day.

To all the family BOUREKOUCHE & ACHOUR
Thanks

ملخص

في هذه الأطروحة، نحقق في عملية تطوير خوارزميات تشفير الصور القائمة على الفوضى من وجهات نظر مختلفة، بما في ذلك التحدي الخطير المتمثل في إنشاء تسلسلات أرقام عشوائية آمنة لاستخدامها كمفاتيح تشفير ديناميكية. أولاً، بهدف تحسين الصفات العشوائية وغير الدورية لمولدات الأعداد العشوائية الزائفة الأساسية المستخدمة كمولدات للتيار الرئيسي، نستغل السمات الفريدة للخريطة اللوجستية، والنظام اللوجستي الجيب، وسجلات تحولات التغذية المرتدة الخطية، وسجل تحول التغذية المرتدة غير الخطية لتصميم مولدات جديدة للتيار الرئيسي. لذلك، تنجح مولداتنا في توليد تسلسلات غير محدودة وعشوائية وغير خطية من خلال اجتياز مجمل الاختبارات الإحصائية للمعهد الوطني للمعايير والتكنولوجيا، وأظهرت أماناً تشفيرياً قوياً، مما أدى إلى إنتروبيا عالية وحساسية عالية للمفتاح ومساحة رئيسية كبيرة تتجاوز 2^{100}.

يسلط الهدف الثاني الضوء على أهمية اختيار بنية مناسبة قائمة على الفوضى للارتباك والنشر. تختلف أبعاد بنية انتشار الارتباك القائمة على الفوضى اعتماداً على الخريطة الفوضوية المحددة المستخدمة. ومن ثم، فإننا نصمم ثلاث خوارزميات لنشر الارتباك على مستويات مختلفة لمناقشة وإثبات تأثير اختيار البعد المناسب للخريطة الفوضوية على ضعف نظام التشفير. لقد ثبت أن الخرائط الفوضوية عالية الأبعاد يمكن أن تعزز القدرة على مقاومة الهجمات الشاملة والإحصائية من خلال تحقيق القيم المرغوبة لمعدل تغيير عدد البكسل ومتوسط الشدة المتغيرة الموحدة، في حين أن هذه الخرائط غير قادرة على الحفاظ على سرعة التشفير.

الهدف الثالث من هذه الأطروحة هو تحسين جوهر النموذج الرياضي لأنظمة التشفير القائمة على الفوضى من خلال تعزيز التعقيد الفوضوي والنطاق الفوضوي للخرائط الفوضوية الأساسية أحادية البعد. حيث نقترح نظام فوضى غير خطي جديد قادر على إنتاج خرائط فوضوية منفصلة معززة ذات بعد واحد، من خلال تطبيق التحويلات غير الخطية المماسية على مخرجات الخرائط الفوضوية الحالية. تعمل هذه الاستراتيجية على تحسين أداء الخرائط الفوضوية الأساسية ذات البعد الواحد من خلال إظهار سلوك ديناميكي أفضل، وأس ليابونوف، وتشعب، وفترات فوضوية أكبر عبر ﴿٠-٤﴾.
كلمات مفتاحية: نظام التشفير، سلسلة مفتاحية، النظام الفوضوي، تشفير الصورة، التسلسل العشوائي، الاختبار الإحصائي.

Abstract

In this thesis, we investigate the development process of chaos-based image encryption algorithms from various perspectives, including the serious challenge of generating secure random number sequences for use as dynamic encryption keys. First, at the aim of improving the randomness and non-periodicity qualities of the basic pseudo-random number generators (PRNG) used as key-stream generators, we exploit the unique attributes of the logistic map (LM), logistic-sine system(LSS), linear feedback shift registers (LFSR), and nonlinear feedback shift register (NLFSR) to design new key-stream generators (namely: LSS-LFSR-PRNG, LM-NLFSR-PRNG, and LSS-NLFSR-PRNG). Therefore, our generators succeed in generating unlimited, random, and nonlinear sequences by passing the totality of the National Institute of Standard and Technology (NIST) statistical tests, and displayed strong cryptographic security, resulting in high entropy, high key sensitivity, and large key space exceeding 2^{100}.

The second goal highlights the importance of selecting an appropriate chaos-based architecture for confusion and diffusion. The dimensions of the chaos-based confusion-diffusion architecture vary depending on the specific chaotic map being used. Hence, we design three confusion-diffusion algorithms of various levels (1D LM-based cryptosystem, 1D LM-Chebyshev-based cryptosystem, and 3D intertwining logistic map-cosine (ILM) based cryptosystem), to discuss and demonstrate the impact of choosing the appropriate dimension of the chaotic map on the vulnerability of a cryptosystem. It has been proven that higher-dimensional chaotic maps, such as 3D-ILM, can enhance the ability to resist exhaustive and statistical attacks by achieving desirable values of the number of pixels change rate (NPCR) and unified average changing intensity (UACI), while these maps are unable to maintain encryption speed.

The third goal of this thesis is to improve the core of the mathematical model of chaos-based cryptosystems by boosting the chaotic complexity and chaotic range of basic one-dimensional chaotic maps. Where, we propose a new nonlinear chaotification system capable of producing 1D enhanced discrete chaotic maps (enhanced tangent-Logistic map T-LM, enhanced tangent-Sine map T-SM, and enhanced tangent-Chebyshev system T-CH), by applying tangent nonlinear transforms to the outputs of the existing chaotic maps. This strategy improves the performance of basic 1D chaotic maps by exhibiting better dynamical behavior, Lyapunov exponent, bifurcation, and larger chaotic intervals across [0-4].

Key words: *cryptosystem, key-stream, chaotic system, image encryption, random sequence, statistical test.*

Résumé

Dans cette thèse, nous étudions le processus de développement d'algorithmes de chiffrement d'images basés sur le chaos à partir de différentes perspectives, y compris le sérieux défi de générer des séquences de nombres aléatoires sécurisés pour une utilisation comme clés de chiffrement dynamiques. Dans un premier temps, dans le but d'améliorer les qualités d'aléatoire et de non-periodicité des générateurs de nombres pseudo-aléatoires de base (PRNG) utilisés comme générateurs de flux clés, nous exploitons les attributs uniques de la carte logistique (LM), du système logistique-sinusoïdal (LSS), des registres de décalage de rétroaction linéaire (LFSR), et registre de changement de rétroaction non linéaire (NLFSR) pour concevoir de nouveaux générateurs de flux clés (à savoir : LSS-LFSR-PRNG, LM-NLFSR-PRNG et LSS-NLFSR-PRNG). Par conséquent, nos générateurs réussissent à générer des séquences illimitées, aléatoires et non linéaires en passant la totalité des tests statistiques du National Institute of Standard and Technology (NIST), et ont affiché une sécurité cryptographique forte, entraînant une entropie élevée, sensibilité élevée des touches et grand espace pour les touches dépassant 2^{100}.

Le deuxième objectif souligne l'importance de choisir une architecture appropriée basée sur le chaos pour la confusion et la diffusion. Les dimensions de l'architecture de confusion-diffusion basée sur le chaos varient en fonction de la carte chaotique spécifique utilisée. Par conséquent, nous concevons trois algorithmes de confusion-diffusion de différents niveaux (1D LM-based cryptosystem, 1D LM-Chebyshev-based cryptosystem, et 3D intertwining logistic map-cosine (ILM) based cryptosystem), pour discuter et démontrer l'impact du choix de la dimension appropriée de la carte chaotique sur la vulnérabilité d'un cryptosystème. Il a été prouvé que les cartes chaotiques de dimension supérieure, telles que 3D-ILM, peuvent améliorer la capacité de résister à des attaques exhaustives et statistiques en atteignant des valeurs souhaitables du taux de changement du nombre de pixels (NPCR) et de l'intensité de changement moyenne unifiée (UACI), alors que ces cartes sont incapables de maintenir la vitesse de cryptage.

Le troisième objectif de cette thèse est d'améliorer le cœur du modèle mathématique des cryptosystèmes basés sur le chaos en augmentant la complexité chaotique et la gamme chaotique des cartes chaotiques unidimensionnelles de base. Où, nous proposons un nouveau système de chaotification non linéaire capable de produire des cartes chaotiques discrètes 1D améliorées (carte tangente-logistique améliorée T-LM, carte tangente-sinus améliorée T-SM et système tangent-Chebyshev amélioré T-CH), en appliquant des transformations non linéaires tangentes aux sorties des cartes chaotiques existantes. Cette stratégie améliore les

performances des cartes chaotiques 1D de base en présentant un meilleur comportement dynamique, un exposant de Lyapunov, une bifurcation et des intervalles chaotiques plus grands sur [0-4].

Mots clés :*cryptosystème, key-stream, système chaotique, cryptage d'image, séquence aléatoire, test statistique.*

Publications List

Journal Papers:

1. BOUREKOUCHE, H., BELKACEM, S., & MESSAOUDİ, N. (2024). Efficient image encryption scheme using a nonlinear shift register and chaos. Journal of Information and Optimization Sciences, 45:1, 157–180, DOI: 10.47974/JIOS-1399

   https://www.tarupublications.com/doi/10.47974/JIOS-1399

2. BOUREKOUCHE, H., BELKACEM, S., & MESSAOUDİ, N. (2024). Lightweight Medical Image Encrypting and Decrypting Algorithm Based on the 3D Intertwining Logistic Map. International Journal of Informatics and Applied Mathematics, 6(2), 46-62. DOI: 10.53508/ijiam.1405959

   https://doi.org/10.53508/ijiam.1405959

3. BOUREKOUCHE, H., BELKACEM, S., & MESSAOUDİ, N. (2024).(in press) Design and analysis of a pseudo random number generator based on the logistic sine system and a nonlinear feedback shift register: Application to image encryption. Journal of discrete mathematical science and cryptography.

4. BOUREKOUCHE, H., BELKACEM, S., & MESSAOUDİ, N. (2024).(in press) New random number generator based on linear shift register and coupled chaotic map for image encryption. Int. J. of Image Mining.

   https://www.inderscience.com/info/ingeneral/forthcoming.php?jcode=ijim

International conferences

1. BOUREKOUCHE, H., BELKACEM, S., & MESSAOUDİ, N. (2024). Toward Ensuring Security for Teleradiology through a New Chaos-Based Medical Image Encryption Scheme. The 1st International Conference on Electrical Engineering & Renewable Energies Systems (ICEERES'23), Bechar.

   https://drive.google.com/file/d/1asS--24jZCaAzfLSwf2w0dzR0isMJeFi/view?usp=sharing

2. BOUREKOUCHE, H., BELKACEM, S., & MESSAOUDİ, N. (2024). Lightweight Medical Images Encrypting and Decrypting Algorithm Based on the Intertwining 3D Logistic map. The 6th international hybrid conference on informatics and applied mathematics IAM'23 December 6-7, 2023 Guelma, alegria. CEUR Workshop Proceedings.

   https://drive.google.com/file/d/1uM8Fx79toVUD4oJZs8eG4CsCyCYNPZUz/view?usp=sharing

3. BOUREKOUCHE, H., BELKACEM, S., & MESSAOUDİ, N. (2024). Impact of confusion-diffusion complexity on maintaining data security. 2nd INTERNATIONAL CONFERENCE ON ELECTRICAL ENGINEERING AND AUTOMATIC CONTROL, Mai 12-14, Setif.

National conferences:

1. BOUREKOUCHE, H., BELKACEM, S., & MESSAOUDİ, N. (2023). Comparative assessment of image cryptography technique based on coupled map and their seed maps. The 1st national conference on emergent technologies in electrical engineering (NCETEE'23), December 16-17,2023, Setif, Algeria.

2. BOUREKOUCHE, H., BELKACEM, S., & MESSAOUDİ, N. (2023). Randomness evaluation of coupled chaotic maps via NIST tests: A comparative study. The 1st National Workshop on Wireless Network, Cloud Computing and Cryptography (WWN3C'23), April 24,2023, Boumerdès.

   https://wwn3c2022.univ-boumerdes.dz/WWN3C23.pdf

Contents

List of Figures

List of Tables

List of Abbreviations

AE: Approximate Entropy
AES: Advanced Encryption Standard
BBS: Blum–Blum–Shub generator
BD: Bifurcation Diagram
BMP: Microsoft Windows Bitmap Format
CD: Correlation Dimension
CMOS: Complementary metal oxide semi-conductor
CT: Computerized Tomography
1D: One Dimension
DCT: Discrete cosine transform
DE: Differential evolution
DES: Data Encryption Standard
DNA: Deoxyribonucleic Acid
DRNG: Deterministic Random Number Generator
DRPE: Double random-phase encoding
DWSS: Discrete Wheel Switching System
DWT: Discrete wavelet transform
EA: Evolutionary algorithms
ECC: Elliptic Curve Cryptography
ECG: Electrocardiogram
ET: Execution Time
FMRI: Functional Magnetic Resonance Imaging
FCSR: Feedback with carry shift registers

FP: Fixed Points
FPGA: Field-Programmable Gate Array
FRT: Fractional Fourier transform
FST: Fresnel transform
FT: Fourier transform
GA: Genetic algorithm
GIF: Graphics Interchange Format
GSM: Global System for Mobile Communication
GT: Gyrator transform
HD: High Dimension
HDL: Hardware description languages
IBM: International Business Machines Corporation
ILM: Intertwining logistic maps
IOT: Internet of thing
JPEG: Joint Photographic Expert Group
KE: Kolmogorov-Sinai Entropy
LCG: Linear congruential generator
LE: Lyapunov dimension
LFSR: Linear Feedback Shift Register
LM: Logistic Map
LTS: Logistic-tent system
LSB: Least Significant Bit
LSS: Logistic-sine system
MLCA: Maximum length Cellular Automata
MLP: Modified logistic map
MRI: Magnetic Resonance Imaging
MSE: Mean Square Error
NCA: Nonlinear cellular automaton
NIST: National Institute of Standards and Technology
NLFSR: Nonlinear Feedback Shift Register

NPCR: Number of Pixels Change Rate
NPTRNG: Non-physical true random number generator
PCRNG: Pseudo chaotic random number generator
PET: positron emission tomography
PNG: Portable Network Graphics
PP: Phase Portrait
PRBG: Pseudo random bit generator
PRNG: Pseudo Random Number Generator
PSNR: Peak Signal-to-Noise Ratio
PTRNGs: Physical true random number generator
PWLM: Piecewise linear map
RAM: Random-access memory
RGB: Red, Green and Blue
SVG: Scalable Vector Graphics
TS: Time Series
TIFF: Tag Image File Format
TRNG: True Random Number Generator
UACI: Unified Average Changing Intensity

General introduction

Motivation

In recent decades, an increasing amount of digital data has been generated and transmitted in all types of networks [1]. The majority of these data are in the form of images; thus, their availability, confidentiality, and integrity must be protected during transmission via a cryptosystem, which involves converting plain images to cipher images at the sender's end using an encryption algorithm and a key.

One of the serious challenges in image encryption is to generate sequences of random numbers by mean of pseudorandom number generators (PRNGs), for use as dynamic encryption keys; hence, ensuring a high quality of randomness and a secure key becomes mandatory for those security modules aimed at providing and strengthening the security level of a system [2].

Traditional encryption methods, such as the Data Encryption Standard (DES), Advanced Encryption Standard (AES) [3], and Blowfish [4], cannot fulfill the requirements for image encryption because of their inefficiency and lack of security.

To address these challenges, researchers have proposed novel encryption algorithms that utilize techniques such as chaos [5], deoxyribonucleic acid (DNA) encoding, and compressive sensing [6]. These algorithms provide high information entropy, a large keyspace, and resistance against various types of attacks. By combining different encryption methods and techniques, these algorithms offer improved security and efficiency in image encryption.

Chaos theory is considered one of the most practical and outstanding PRNG methods and a source of randomness. Its simple structure can lead to extremely complex and unpredictable behavior. In addition, its sensitivity to the initial conditions, that is, the observed output, changes dramatically, even when the original parameters are slightly altered. Other significant properties of chaos include mixing, ergodicity, unpredictability, nonperiodicity, and pseudorandomness. These properties are linked to confusion and diffusion by separating

adjacent pixels and spreading random substitutions through all encrypted images.

However, most one-dimensional (1D) chaotic maps are unsuitable for image encryption owing to their small chaotic parameter range, uneven distribution of chaotic behavior, lack of strong uncertainty, and insufficient key space [7]. However, new solutions have to be developed to address these issues and enhance the overall security of image encryption systems.

The first approach is to integrate chaotic generators with other random number generator (RNG) techniques, such as cellular automata (CA), linear congruential generators (LCGs), block ciphers [8], hash functions, elliptic curve methods, and feedback shift registers (FSRs), to enhance their randomness and security properties [9]. This approach may boost the effectiveness of these methods and yield promising results in terms of randomness, security, and applicability in various fields.

The second approach aims to solve the problems of uneven distributions of conventional chaotic behavior, insufficient key space, and small amounts of key change in a single chaotic system [10]; this approach involves the use of modified forms of chaotic maps. Thus, new techniques for creating new chaotic systems with remarkable qualities, such as cascading [11], switching, perturbing maps, time parameter control of chaotic systems, and nonlinear combination of chaotic systems [12], have been proposed. These improvements effectively compensate for the defects of simple 1D chaotic systems.

The confusion-diffusion architecture of a cryptosystem is also another challenging task in ensuring the security and robustness of the encryption process. The security of the cryptosystem depends strongly on the confusion and diffusion architecture. An insufficient level of confusion and diffusion may cause inefficiency [13]. Apart from security, high speed, which is the other main requirement of a secure cryptosystem, may be affected by choosing an appropriate confusion-diffusion architecture.

Contributions

This thesis is based on three major contributions. In the first contribution, the integration of chaotic maps with two types of feedback shift registers(FSR) was explored based on random properties and the ease of implementation of feedback shift registers to design a new secure key-stream generators: LSS-LFSR-PRNG, LM-NLFSR-PRNG, and LSS-NLFSR-PRNG, these systems are effective at overcoming the limitations of simple one-dimensional chaotic maps and eliminating the linearity and periodicity of the FSR output, resulting in a good random number generator.

In the second contribution, we discuss the impact of choosing the appropriate remote chaotic map for confusion-diffusion in affecting factors such as time complexity, space com-

plexity and resistance to several attacks, during the encryption and decryption processes. Where, we compare and demonstrate the effectiveness of three proposed confusion-diffusion algorithms, 1D LM-based cryptosystem, 1D LM-Chebyshev based cryptosystem, and 1D intertwining LM- based cryptosystem. These systems have different confusion-diffusion architectures that vary from simple to complex architecture.

In the last contribution, we propose a tangent nonlinear transformation that has the ability to generate completely different new chaotic systems (enhanced tangent-Logistic map T-LM, enhanced tangent-Sine map T-SM, and enhanced tangent-Chebyshev system T-CH), which typically possess more parameters and more intricate chaotic behaviors and produce more random and unpredictable output sequences. This approach can significantly expand the chaotic ranges of an established chaotic map and offer users great flexibility in selecting seed maps to produce numerous new chaotic maps.

Thesis organization

The thesis is organized as follows:

Chapter 1 provides the necessary details and background of the thesis subject. It covers basic digital image concepts, image encryption approaches, and a theoretical foundation for developing safe PRNG systems.

Chapter 2 introduces the basis of nonlinear dynamics of chaotic systems, and provides a comparative analysis based on the randomness of two coupled maps (a logistic-sine system (LSS) and a logistic-tent system (LTS)). The evaluation measures include the National Institute Statistical Test (NIST), and the performances of the chaotic maps are analyzed on specific parameters to assess their ability to produce random sequences valid for use as secret keys.

Chapter 3 investigates a new coupling method for PRNG designs based on chaotic maps and FSR, where different coupling combinations are proposed: LSS-LFSR-PRNG, LM-NLFSR-PRNG, and LSS-NLFSR-PRNG.

Chapter 4 covers the proposed cryptosystems based on different confusion-diffusion architectures (1D LM-based cryptosystem, 1D LM-Chebyshev based cryptosystem, and 1D intertwining LM- based cryptosystem), with the most important security analysis and results compared to those of previous works.

Finally, Chapter 5 presents a new nonlinear chaotification system capable of producing new 1-D discrete-chaotic maps that exhibit improved dynamical behavior. We also propose a new encryption algorithm in which the optimal sequences generated by the designed systems T-LM and T-CH are used as secret keys for the confusion-diffusion process.

Chapter 1

Theoretical background of digital data encryption

1.1 Introduction

The significance of data encryption in current digital environments cannot be emphasized. Regardless of their rampant usage for end-to-end secret information sharing across networks in several fields, hence, an attacker may have a substantial chance to steal sensitive information. The primary goal of protecting these data, which are mostly in the form of images, is to use an algorithm that makes it impossible for an adversary to access any information. This chapter outlines the essential concepts of digital images, cryptography, and image cryptography. We review and summarize the development of the current image encryption algorithms, analyzing their suitability for different scenarios and expounding their advantages and disadvantages. Evaluation metrics to measure security and performance of encryption algorithms is also highlighted. At last, theoretical concepts of generating random numbers are reviewed.

1.2 Digital image concepts

A digital image is a spatial representation of objects, scenes, or other images in a digital format[14]. It is composed of x, y elements having an exact location and value and often referred as pixels[15]. It can be written as a mathematical function $f(x, y)$ that is divided into x rows and y columns. The coordinate ranges are $X = 0, 1, ..., m - 1$ and $Y = 0, 1, 2, ..., n - 1$

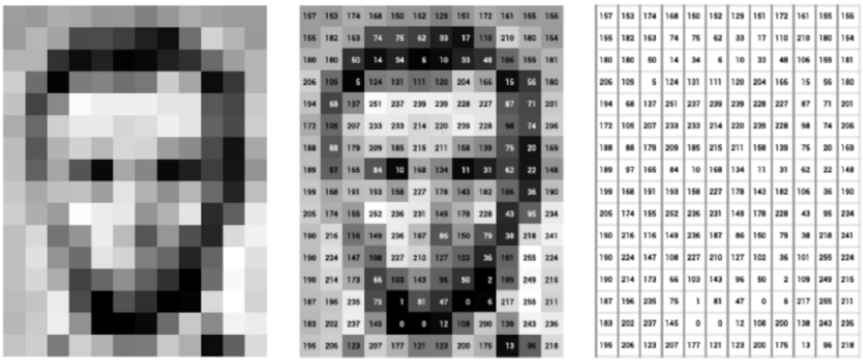


Figure 1.1: Digital image as a matrix of numerical values represents pixel's brightness [16]

as given below by Equation 1.1:

$$f(x,y) = \begin{pmatrix} f(0,0) & f(0,1) & f(0,2) & \dots & f(0,y-1) \\ f(1,0) & f(1,1) & f(1,2) & \dots & f(1,y-1) \\ & & & \dots & \\ f(m-1,0) & f(m-1,1) & f(m-1,2) & \dots & f(m-1,n-1) \end{pmatrix} \tag{1.1}$$

A sample digital image and its equivalent matrix are shown in Figure 1.1. The matrix's elements are shown as either 0 or 255, which represents the value of the function $f(x,y)$, and each element is known as a pixel. They represent the discrete data of any digital image, and serve as the actual building blocks of digital images. The value of the function $f(x,y)$ at every point indexed by a row and a column is a number and has no units, and it is known as the gray value of intensity of the image at that point [17].

1.2.1 Types of digital images

Classification of images can be performed on the basis of various criteria like attributes, color, dimension, and data types.

1. **Classification of images on the basis of attributes**
   Based on the attributes of any image, it is classified as raster or vector graphic images.

   (a) Vector graphics: vector graphics uses graphic primitives to describe an image. Hence, the notion of resolution is practically not present in graphics. SVG (Scalable Vector Graphics) is a common vector image format that enables to describe two-dimensional graphics using graphical primitives like lines, circles, and polygons [18].

(b) Raster graphics: raster images, also called bitmap images, are pixel-based which mean a collection of pixels arranged on a rectangular grid, hence, their quality is dependent on the number of pixels[19].

2. **Classification of images on the basis of color**

   On the basis of color, the images can be classified into the following categories:

   (a) Monochrome images: are the images where the color component is absent, they are further classified as gray-scale and binary images.

   (b) Gray-scale images: the spectrum of shades between white and black, or vice versa, is referred to as grayscale, such images have many shades of gray, and eight bits ($2^8 = 256$) are enough to represent the gray-scale [20].

   (c) Binary images: the binary images are just special case of grayscale images, where the process of thresholding is applied, they are actually Bi-level images where the pixels assume the values of 0 or 1 [21] [22].

   (d) True color (or full color) images: are the images where the pixel has a color that is obtained by mixing the primary colors red, green, and blue[20].

   (e) Pseudo color images: are in fact false color images, their color component is manipulated artificially.

3. **Classification of images on the basis of dimensions**

   Images can be classified on the basis of dimensions also. Normally, digital images are a $2D$ rectangular array of pixels. If another dimension, of depth or any other characteristic, is considered [20], which may be necessary to use, then a higher-order stack of images like 3D images are produced. A good example of a $3D$ image is a volume image, where pixels are called voxels [20]. By $3D$ images, it is meant that the dimension of the target in the imaging system (may be a scene or an object) is three dimensional (x,y, depth).

4. **Classification of images on the basis of data types**

   Images may be classified based on their data type. For example, image processing operations may produce images with decimal fraction, or negative number, and maybe complex number [20].

1.2.2 Image characteristics

A digital image can be described with respect to several characteristics or fundamental parameters[23], these ones help to assess its quality at the aim to improve the quality of an image including intensity, contrast, brightness, noise, resolution, and the bit depth.

1. **Intensity**

   The term intensity refers to the amount of light or the numerical value of a pixel, it is the measure of energy of a wave, which is directly proportional to square of amplitude of the signal. From the point of view of image processing it is a numerical value which represents a pixel.

2. **Contrast**

   The term contrast of an image [23] relates to recording of the differences in the magnitude of the intensity at the surface of an object. It can be described as a product of the sensor signal contrast and detector contrast. But, a common measure of contrast (C) as given by Equation 1.2 involves intensity of foreground (I_{object}) and background ($I_{background}$) objects, i.e.

   $$C = \frac{I_{object} - I_{background}}{I_{object} + I_{background}} \tag{1.2}$$

   Where, I_{object} is the average pixel intensity of the object pixels and $I_{background}$ is the averaged pixel intensity of the background. Another useful contrast measure using the same parameters I_{object} and $I_{background}$ is given by Equation 1.3.

   $$C = log_{10}\frac{I_{object}}{I_{background}} \tag{1.3}$$

3. **Brightness**

   Brightness resolutions expresses brightness quantization accuracy. Brightness resolution is also a kind of relative resolution because it refers to the average pixel intensity of the image.

4. **Noise**

   Noise is an unwanted disturbance that causes fluctuations in the pixel value. It is a random or stochastic process, and hence its true value cannot be predicted accurately [23].

5. **Resolution**

   Resolution refers to the quantity of pixels present in an image[23].

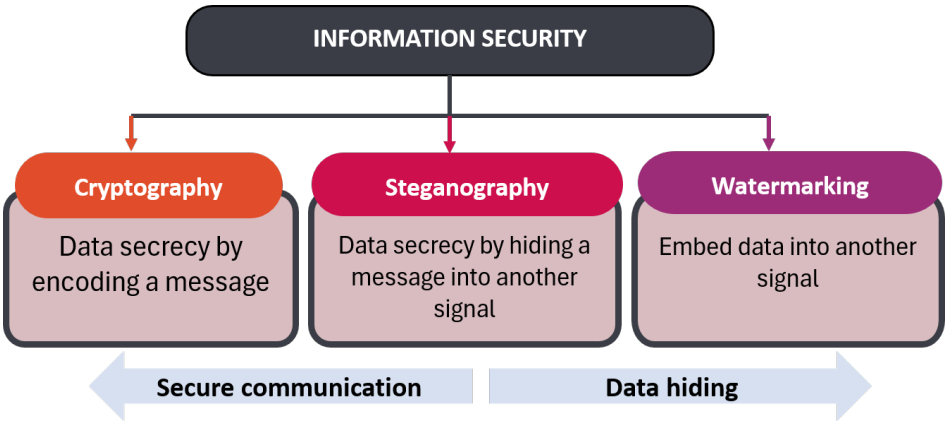


Figure 1.2: Main classification of information security methods

6. **Bit-depth**

   Bit-depth refers to the shades of gray used to characterize each pixel. This is often quantified in terms of the number of bits.

1.3 Information security and cryptography essentials

Security and privacy touch on various applications, ranging from safe commerce and payments to private conversations and preserving health care information. One key feature of safe communication is cryptography. In this section, we cover the fascinating area of data security. We clarify some of the terms and principles underlying fundamental cryptographic approaches and provide a tool to compare the numerous cryptographic algorithms currently in use.

1.3.1 Information security

In today's world, humans are more dependent on computer networks to communicate with one another. Communication over the Internet is efficient, but intruders can steal private information. Vulnerabilities and security breaches occur due to poor layout, misuse by the user, or defects of the system. Protecting this shared information in potentially hostile environments is crucial for the growth of information-based processes in industry, business, and administration. Hence, many authors have proposed various approaches to enhance the security of confidential data transmission. From this perspective, the three techniques, cryptography, steganography, and watermarking with their main methods (see Figure 1.2) are widely used to hide the original transmitted data for secure communication.

Steganography, regarded as the art of clandestine communication [24], is defined as the

art or practice of hiding information, concealing a file, message, image, or video.

Watermarking may be described as the art of injecting the concealed stream of bits (the watermark) into a component (a file)[25], or digitally sent data that can be detected by a computer program to confirm the authenticity and integrity of the component or the transmitted data. The file may be audio, text, pictures, or video[26]. Digital watermarking is now used for a variety of purposes, including owner identification, broadcast monitoring, copy control, content authentication, transaction tracking, owner identification, and proof of ownership [27].

Cryptography is thus used to achieve three main security objectives: confidentiality, integrity, and authenticity [28]. It is described as the ability to create secret code to protect the data sent between two communicating parties from an outside attack [29].

1.3.2 Image encryption terminology

Image encryption uses a mathematical algorithm to convert the original image into a form that is hard to interpret [30]; it is used to safeguard sensitive image data in several sectors. Recent applications of image encryption are depicted in Figure 1.3. These applications include internet and communication multimedia systems [31], telemedicine [32], and military communication [33]. As instance, in telemedicine applications, image encryption aims to ensure the confidentiality and integrity of medical images during storage and transmission, and it plays a crucial role in military communication by ensuring the secure transmission and storage of military maps and sensitive information.

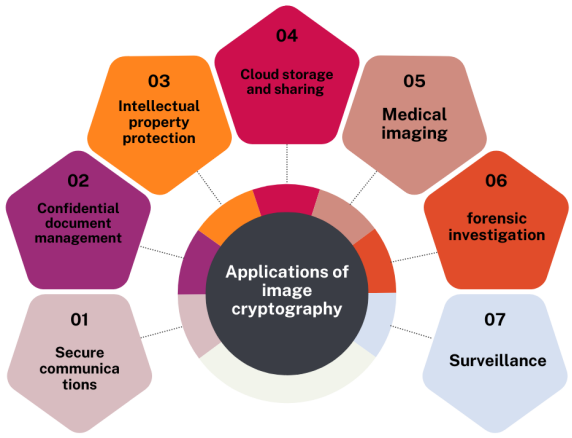


Figure 1.3: Applications and use cases of image encryption

The fundamental components of every encryption cipher is defined by Shannon's principle.

The process involves converting a plain image into an unreadable encrypted image using an encryption algorithm controlled by an encryption key[34], according to the two phases "confusion" and "diffusion" (see Figure 1.4).

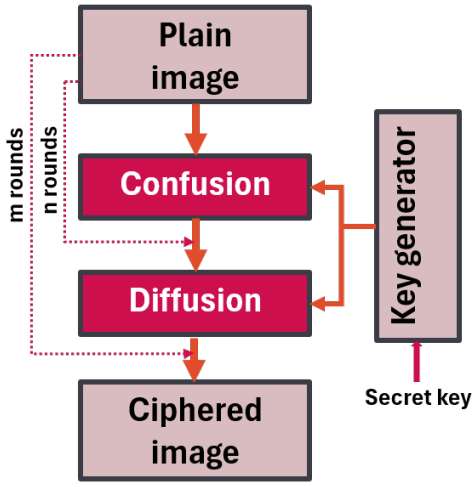


Figure 1.4: Confusion-diffusion architecture [35]

Here, we define terms that are a subset of image cryptology.

Cryptology: is a mathematical science, that support cryptography and cryptanalysis.

Cryptography: is the discipline incorporates ideas, tools, and procedures for the alteration of data in order to conceal their semantic content, prohibit their illicit use, or avoid undetected changes [36].

Cryptanalysis: is the examination of cryptographic systems to identify vulnerabilities or information disclosure[37].

Cryptosystem: is the use of cryptography methods and a supporting framework to provide information security services. A cipher system is another term used for a cryptosystem. A simple cryptosystem comprises the following parts: plaintext, encryption-key, encryption/decryption algorithm, cipher-text, and decryption key.

Plain-Text: is the original message that to be transmitted or stored, it feeds into the algorithm as input [36].

Cipher-Text: is the scrambled or encrypted message produced as output [38].

Encryption: is a process that makes the information in a message or file unreadable to unauthorized individuals [39].

Decryption: is the reverse process of encryption in which the intendent recipient can reveal the encrypted message [40], it requires two things; a decryption algorithm and a key.

Key: it is a string of characters that appears random. Depending on the specific key the

algorithm will produce a different output [41].

Confusion: is also called substitution, or permutation. Through permutation, the position of the pixels can be modified, making the statistical link between the ciphertext and secret key as complex as feasible [42].

Diffusion: is also called transposition, it modifies the value of a pixel, spreading the influence of a single byte/bit to subsequent bytes/bits [43]. Diffusion hides plaintext's statistical properties. For a robust cryptographic scheme, both confusion and diffusion are essential. If a cipher is constructed with confusion or diffusion, it will not be able to withstand a cryptanalytic assault [44].

Attack: an attack is any effort to penetrate a computer system, steal data, or use a hacked computer system to launch another attack. A multitude of tactics, including malware, phishing, ransomware, and man-in-the-middle attacks [45], are used by cybercriminals to initiate attacks.

1.3.3 Image encryption approaches

Several image encryption approaches have been introduced so far including spatial, transform, optical, and compressive sensing schemes. A comparison of the discussed techniques is resumed in Table 1.1.

1. **Spatial domain techniques**
   The spatial domain encryption technique operates directly on the image pixels. Famous approaches are addressed as follows:

   (a) **Chaos based image encryption techniques**
   Chaos-based cryptographic models have been utilized to construct innovative and efficient image encryption systems, demonstrating excellent qualities such as speed, cost, computational power, computational overhead, complexity, and vulnerability [30]. The two stages of a cryptosystem's architecture are typically the confusion and the diffusion phases. Therefore, the sequence produced by chaotic systems sequentially modifies the values of all pixels in the image when the diffusion phase is carried out with the aid of a chaotic map [46].

   (b) **DNA based image encryption techniques**
   The cryptographic approach based on deoxyribonucleic acid (DNA) converts each letter of the alphabet into a unique combination of the four bases Adenine (A), Cytosine (C), Guanine (G), and Thymine (T) [47].

Researchers are interested in DNA based image encryption owing to its low power consumption, high information density, and parallel processing capabilities[30]. The lack of random DNA rule implementation at each encryption level is a major drawback of current DNA cryptography systems. This puts data encryption and key generation security at risk. Because the approaches have static beginning settings, they are readily exploited [48].

(c) **Cellular automata based image encryption techniques**

Cellular automata-based encryption algorithms have been applied to various image encryption scenarios [49], due to its features of homogeneity, parallelism, unpredictability, and ease of implementation in both hardware and software systems [30]. These complex models offer a high degree of efficiency and resilience. Cellular automata produce random sequences to spread pixel values in an image. However, it is crucial to emphasize that the design and analysis of cellular automata-based encryption methods need significant thought. Factors such as cellular automata rule selection, key management, and security analysis are critical to assuring the encryption scheme's efficacy and resilience.

(d) **Meta-heuristics based image encryption techniques**

Metaheuristic approaches are mainly utilized in a situation where we need optimized results and to provide near to optimal solutions. Recently, the use of such approaches has been increased in the image encryption. Metaheuristic techniques may be used for image encryption in two ways: generating numerous cipher images and selecting the best one, and optimizing chaotic map initially parameters for efficient key generation. Researchers used metaheuristic techniques for image encryption, taking into account many features [50]. There are many metaheuristic approaches in the literature that yield optimum solutions, including GA, Multi-swarm Intelligence Algorithm, Differential Evolution (DE), and the Bat algorithm. GA belongs to the larger family of evolutionary algorithms (EA).

(e) **Elliptic curve based image encryption techniques**

Elliptic curve cryptography (ECC) has shown to be an efficient encryption system [51], because it works on the least amount of memory with the small key size. In general, elliptic curves are non-singular cubic equations having an elliptic form over a finite field. The symmetry of these structures around the $x-$axis is crucial for their functioning.

$$y^2 = x^3 + ax + b \mod p \tag{1.4}$$

where a and b are integers which satisfy $4a^3 + 27b^2 \neq 0(\mod p)$ and p is a large prime number.

Kumar et al.[**?**] suggested an improved asymmetric cryptosystem technique for image encryption, combining Elliptic Curve and Fourier transforms. To construct affine coordinates on the elliptic curve, pixel values were transferred to elliptic curve coordinates and point multiplication was conducted using the 'G' generator. When decrypting the cipher image, a matching pixel value is created. The bigger number cannot be more than the prime 'P', one of the parameters in the finite field equation of the elliptic curve, and we focused on scrambling at the time of encryption and descrambling at the time of decryption in our algorithm.

Transform domain techniques

Transform-based techniques are commonly used for image encryption, where the data are transformed from spatial to frequency using the appropriate following transforms.

1. **Fractional Fourier transform based image encryption techniques**

   The Fractional Fourier Transform (FRT) is applied in various image encryption techniques to enhance security and protect information. FRT, in an analogous way, can be seen as a linear transformation that rotates the signal through any arbitrary angle into a mixed frequency space domain. Equation 1.5 defines the FRT, which is an extension of the ordinary Fourier transform (FT), with the addition of a fractional order to indicate the domain of the signal being transformed [52].

$$F^{\alpha}(f(x))(u) = \int_{-\infty}^{+\infty} K_{\alpha}(x,u) f(x) dx \tag{1.5}$$

   where $K_{\alpha}(x,u)$ is expressed as [51]:

$$K_{\alpha} \begin{cases} A \exp[i\pi(x^2 \cot\varphi - 2xucsc\varphi + u^2 \cot\varphi)], \alpha \neq n\pi \\ \delta(x-u), \alpha = 2n\pi \\ \delta(x+u), \alpha = 2(n+1)\pi \end{cases} \tag{1.6}$$

$$A = \frac{\exp(-i(\pi\frac{\varphi}{4} - \frac{\varphi}{2}))}{\sqrt{\sin(\alpha)}) (1.7)}$$

   Where $\varphi = \frac{\alpha\pi}{2}$ is the angle analogous to the transform order α along the $x-$axis.

2. **Fresnel, wavelet and cosine transform based image encryption techniques**

   Fresnel, wavelet, and cosine transform-based image encryption techniques have been

proposed in several papers. Fresnel transform (FST) represents free-space propagation, which is a basic optical process. Every FST may be connected to an FRT of some order, followed by a suitable magnification and further quadratic phase multiplication [52]. These techniques offer several advantages in the field of image encryption, and provide high levels of security in the frequency domain, making them resistant to attacks and statistically robust.

3. **Gyrator transform based image encryption techniques**
   Rodrigo introduced the gyrator transform, a standard transform in image processing that employs three lenses separated by a set distance [53]. The 2-D function $f(x, y)$ for GT can be mathematically expressed by Equation 1.8 as follows:

$$GT(u, v) = GT^{\alpha} f(x_i, y_i(u, v) = \oint_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x_i, y_i) K_{\alpha}(x_i, y_i, u, v)\, dx_i\, dy_i \tag{1.8}$$

Optical image encryption techniques

Optical techniques are often used in cryptography owing to their fast calculation and parallel processing capabilities. A double random-phase encoding (DRPE) method is used to transform a simple image into stationary white noise. Two random-phase masks were used in the input and Fourier planes. These random phase masks serve as keys to the DRPE [50]. This technology has been extensively investigated and many optical encryption schemes have been developed.

Compressive sensing based image encryption techniques

Researchers have proposed numerous compressive-sensing-based image encryption techniques, since it can simultaneously perform compression and encryption [54]. It employs a measuring matrix and a reconstruction method. A measurement matrix was used for the compression. The measurement matrix may be used as a secret key between the sender and recipient, creating a cryptosystem.

Table 1.1: Advantages and disadvantages of most common image cryptographic algorithms

Techniques	Advantages	Disadvantages
1D chaotic system	Robustness against commonly known attacks. Simple implementation. High encryption efficiency.	Small key space.
HD chaotic system	High security. Large keyspace. High speed. Strong anti-attack ability.	Complicated and lengthy algorithm.
Cellular automata	The dynamic structure of keyspace.	The algorithm is too lengthy.
DNA	Very low correlation coefficient. High entropy. High security. Strong anti-attack ability.	Not a cost-effective process.
Elliptic curve	Less use of storage. Low consumption of CPU. Strong short keys.	Using pre-computed tables.
Deep learning	Nonlinear characteristics. Large key space. High security. The use of multiple layers in neural networks. Improves the algorithm's performance. Enhanced security without compromising image quality.	The training time of the model is long /computationally expensive/complex in terms of implementation and execution.
Fuzzy logic	Increased security. Imperceptibility. Resistance to certain attacks.	Reduced visual quality/complexity/resource-intensive/vulnerable to brute force attacks.
Metaheuristic	Optimization of initial conditions improving the performance of encryption algorithms.	Increased complexity and operation time/ pixel expansion and image quality degradation.
Compressive sensing	Low-cost compression coding. Reduces the bandwidth and storage demand. Better computational speed.	Poor reconstructed image quality/ low compression ratios.
Optical	Faster and suited to real-time applications.	Frequency and spatial information are not retained.

Neural network-based encryption algorithm

Neural network-based encryption algorithms have gained popularity owing to their ability to recognize complex patterns and correlations in data [49]. These techniques use artificial neural networks to encrypt images. Deep neural networks perform in computer vision and

pattern recognition, among other applications.

1.3.4 Typical tests for image cryptography

The performance evaluation parameters used in majority of image cryptography methods include four main groups of tests as schematized by Figure1.5, including visual, diffusion and quality, valance effect, and computational processing analyzes [55].

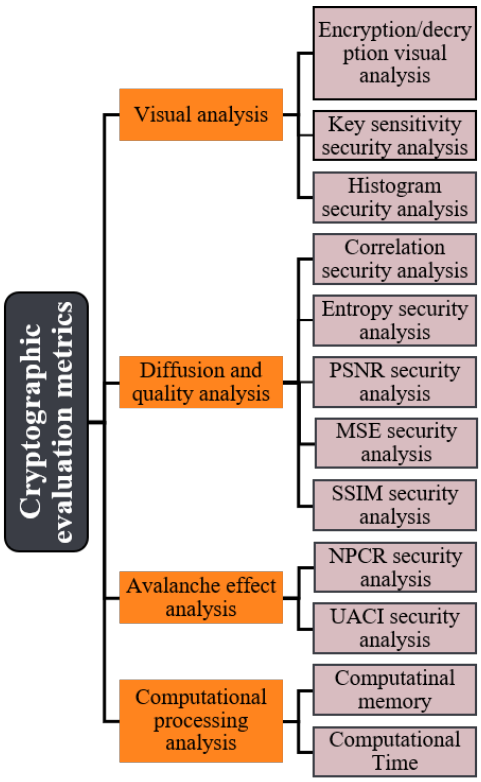


Figure 1.5: Cryptographic evaluation metrics used for test

1. **Visual analysis**

   The arrangement, color, and orientation of the pixels make up a large amount of information in a plain image. All of these details about the plain image must be hidden by the cipher image once an encryption technique has been applied. Several evaluation criteria are included in visual analysis, including the key sensitivity, the histogram, and an analysis of the encryption and decryption procedures.

   (a) **Encryption/decryption visual analysis**

   One of the most important measures used to evaluate the durability and effectiveness of the cryptography method is the visual encryption/decryption analysis.

Where the primary information included inside the plain image should be able to be concealed and disappeared by an excellent encryption technique. In parallel, it should successfully recover and decipher image with high efficiency [56].

(b) **Histogram security analysis**

The histogram plot is used to demonstrate the pixel strength distribution and rates of an image [57]. An effective image encryption technique should remove the image's features and provide a uniform histogram for the encrypted image [58].

(c) **Key sensitivity analysis**

Confusion is an essential feature of cryptography. This obscures the link between the cipher image and the key. A slight modification of the key should produce a completely different cipher image [59]. According to a key sensitivity study, changing a single bit in the key causes a considerable change in the cipher image [58].

2. **Diffusion and quality analysis**

The following four metrics are used to evaluate the quality and diffusion traits of an image cryptography algorithms:

(a) **Information entropy**

The entropy $H(S)$ of a source S defined by Equation 1.9 [60] is calculated to determine the randomness of the pixel values in the image [61]. Consequently, the entropy of an $n-bit$ image is n when the pixel distribution is perfectly uniform, for an 8 bit gray-scale image, it should belongs to $[0, 8]$ [62].

$$H(S) = -\sum_{i=0}^{2^{n-1}} P(S_i) \log_2 P(S_i) \tag{1.9}$$

n is the number of bits needed to express a symbol [63]. S_i and $P(S_i)$ represents the probability of symbol S_i [64].

(b) **Correlation security analysis**

The correlation coefficient is used to assess how closely an important pixel in an encrypted image resembles its original state. The vertical, diagonal, and horizontal orientations had a significant impact on the rate of neighboring pixels in the original data. The strong image-ciphering technique minimizes the associations between pixels in the image [62]. According to the estimates, the correlation coefficient value is calculated by Equation 1.10 as follows:

$$\begin{cases} r_{x,y} = \frac{Cov[x,y]}{\sqrt{D(x)D(y)}} \\ D(x) = \frac{1}{N}\sum_{i=1}^{N}[(x_i - E(x))]^2 \\ E(x) = \frac{1}{N}\sum_{i=1}^{N}[x_i] \\ Cov[x,y] = \frac{1}{N}\sum_{i=1}^{N}[x_i - E(x)][y_i - E(y)] \end{cases} \tag{1.10}$$

Where the gray-scale values of two neighboring image pixels are defined by x and y. The total number of duplets (x, y) identified from the image is N [65].

(c) **Peak signal-to-noise ratio (PSNR)**

The PSNR [66] metric formulated by Equation 1.11 is used to measure the superiority of the encryption/decryption procedures [57]. It stands on the mean square error (MSE) represented by Equation 1.12, where i, j, and k are the pixel positions, and I and K are the pixel values of the original and encrypted images [67].

$$PSNR = 20\log_{10}\frac{255}{\sqrt{MSE}} \tag{1.11}$$

$$MSE = \frac{1}{W \times H \times L}\sum_{i=1}^{W}\sum_{j=1}^{H}\sum_{k=1}^{L}[I(i,j,k) - K(i,j,k)]^2 \tag{1.12}$$

3. **Avalanche effect analysis**

By carefully examining the connections between plain and ciphered image, a differential attack is a form of attack that may be used to recover the input image from the encrypted image without the secret key. This is measured using:

(a) **Unified averaging changing intensity (UACI)**

UACI is employed to analyze the differential attacks [68]. It calculates the average intensity of divergence between encrypted and relevant plain images that vary by one pixel as follows [62]:

$$UACI = \sum_{i=1}^{M}\sum_{j=1}^{N}\left(\frac{|I1(i,j) - I2(i,j)|}{W \times H \times L}\right) \times 100 \tag{1.13}$$

where W and H are the width and the height of the image, respectively, and L is the maximum pixel value possible in an image [69]. Q is the number of bits corresponding to the red, green, and blue channels [70].

(b) **Number of pixels change rate (NPCR)**

The value of NPCR is estimated as:

$$NPCR = \sum_{i=1}^{M}\sum_{j=1}^{N}\frac{A(i,j)}{W \times H} \times 100 \tag{1.14}$$

where $A(i,j) = \begin{cases} 0 & if \quad I1(i,j) = I2(i,j) \\ 1 & otherwise \end{cases}$

W and H are the width and the height of the image, respectively, and L is the maximum pixel value possible in an image [69].

4. **Computational processing analysis**

   Owing to the vast amount of image data that must be processed, computational processing analysis is crucial for image encryption and decryption techniques. It assists in identifying potential areas for improvement and optimization, such as reducing useless tasks or memory utilization.

   (a) **Run-time analysis**

   The execution time is measured as the time required before the encryption process begins [62]. It is the sum of run and compile times. Minimum run-time , which is measured in minutes, milliseconds, or seconds, influences the efficacy of the approach [62].

   (b) **Memory analysis**

   Memory analysis is used to estimate the memory required by the algorithm [32].

1.4 Random number generators

The term "randomness" is often used in a variety of contexts, including computer programming, simulations, numerical analysis, decision-making, sampling, and cryptography. Most of the time, the overall notion underlying this generic phrase relates to distributions, sequences, or homogeneous outputs produced by a particular entropy source. Given past history or any other knowledge, a process is considered "random" if the known unconditional probability and the known conditional probability of the subsequent occurrence are the same [71].

Usually, randomness is expressed in terms of complexity or probability. A random bit sequence might be viewed as the result of unbiased "fair" coin flips with sides that are labeled “0” and “1” [72], with each flip that are independent of each other and having a probability of exactly 1/2 of producing a “0” or “1” [73].

Many cryptosystem algorithms rely on the process of generation random numbers [74], by which one or more digits are arbitrarily produced. In most cases, a random number generator algorithm can be given by Definition 1.4.1 (see also Figure 1.6).

Definition 1.4.1 *A random number generator algorithm is defined by a tuple* (S, f, g, U, x^0)*, in which* S *is the state space of the generator,* U *is the random output space,* $f : S \to S$ *is the transition mapping function,* $g : S \to U$ *is the output extractor function from a given state, and* x^0 *is the seed [74].*

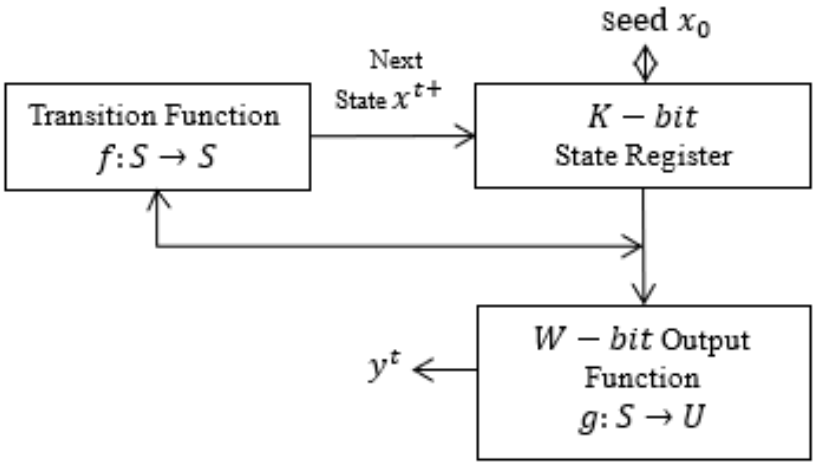


Figure 1.6: General architecture of a random number generator [75]

1.4.1 Classification of RNG

A well-known classification of the different types of random numbers generators [76], establishes three fundamental groups as illustrated by Figure1.7: the so-called true-RNG, pseudo-RNG, and quasi-RNG.

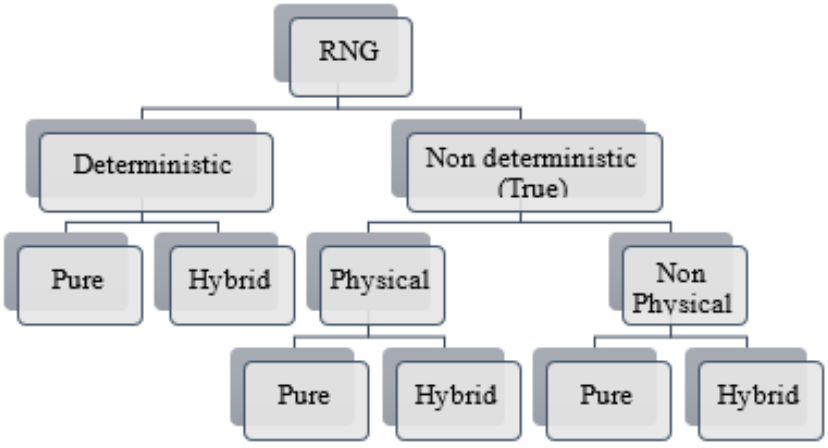


Figure 1.7: Fundamental classification of random numbers generators [77]

1. **Pseudo-random number generators**
   The first class includes deterministic random number generators (DRNGs) [76], sometimes known as pseudorandom number generators (PRNGs). The word "pseudo-random" is often used to characterize random numbers that are generated algorithmically on computers by starting with a seed (see Definition 1.4.2). When the PRNGs can rapidly generate a large number of numbers, they are considered efficient. For the most part, modern PRNGs may be disregarded because of their lengthy duration [78].

   Definition 1.4.2 (PRNG) *A pseudo-random number generator G is a structure (S, μ, f, U, g), where S is a finite set of states, μ is the probability distribution on S for the initial state called seed, $f : S \to S$ is the transition function, U is the output space and $g : S \to U$ is the output function.*

   As previously stated, pseudo-random numbers are distinguished by the fact that they are created using an algorithms and are not truly random, even if they seem to be such if the algorithm is unknown.

2. **True random number generators** True random numbers (TRNGs), which make up the second class, are created with an unexpected input. Using the entropy of a random phenomena, such photoelectric fluctuations or computer clocks. They are particularly helpful in encryption systems since they are unpredictable. Physical TRNGs (PTRNGs) and non-physical TRNGs (NPTRNGs) [79] are the two subclasses of TRNG. Physical TRNGs use non-deterministic effects of electronic circuits. NPTRNGs exploit non-deterministic events [72]. The generation of such number sequences is influenced by an unpredictable resource in the sense of having high entropy [76].

3. **Quasi random numbers generator**
   Quasi-random numbers, named also as low-discrepancy points [76]. These numbers are not meant to be a random sequence, although they do have statistical characteristics in common with random sequences. They are produced by a numerical algorithm that attempt to evenly fill an n-dimensional space with points, without clustering or grouping of points, but are not intended to look random; rather, they are dispersed as evenly as possible to lower errors in Monte Carlo integration. They are employed in the numerical evaluation of integrals, yielding, instead of numbers, a series of points in a chosen dimension that meet an equidistributional requirement [76].

1.4.2 Techniques for generating pseudo random numbers

There are currently several basic techniques in common use for generating uniform random numbers, in this section, most common PRNGs are covered.

1. **Linear congruential generators**

   Linear congruential generator is a source of random numbers. Where, each single number determines its successor. The form of the generator is defined by Equation 1.15.

   $$x_{i+1} = (ax_i + c) \mod m \tag{1.15}$$

   a is called the "multiplier"; c is called the "increment"; and m is called the "modulus" of the generator [71]. A linear congruential PRNG enters a readily observable cycle, the duration of which is termed a period, with a maximum period of m. The period m can be achieved by certain values of the pair (a, c) [80]. The most frequent LCGs have m values of $m \leq 2^{64}$; that is why they doesn't exhibits good statistical properties. As a result, their applications in simulations (such as Monte Carlo) are restricted. This problem may be handled in situations where m can be configured to be several hundred or even thousands of bits in size. However, even though increased precision integer arithmetic can now be implemented in hardware due to significant advancements in modern microprocessor technology, for such values of m, arbitrary precision integer arithmetic must be implemented in software libraries, which may be prohibitively expensive for practical purposes.

2. **Ranrot generators**

   Initially, the Monte Carlo methods were developed using the Ranrot generator class. They are based on the Fibonacci sequence, with the addition of a bit shifting operation [80]. There are more types of Ranrot generators defined by Equation 1.16-1.19 [81].

   Type A:

   $$x_n = ((x_{n-j} + x_{n-k}) \mod 2^b) \gg r \tag{1.16}$$

   Type B:

   $$x_n = ((x_{n-j} \gg r_1) + (x_{n-k} \gg r_2))(\mod 2^b) \tag{1.17}$$

   Type B3:

   $$x_n = ((x_{n-i} \gg r_1) + (x_{n-j} \gg r_2) + (x_{n-k} \gg r_3))(\mod 2^b) \tag{1.18}$$

Type W:

$$\begin{cases} z_n = ((y_{n-j} \gg r_3) + (y_{n-k} \gg r_1)(\mod 2^{\frac{b}{2}}) \\ y_n = ((z_{n-j} \gg r_4) + (z_{n-k} \gg r_2)(mod 2^{\frac{b}{2}}) \\ x_n = y_n + z_n 2^{\frac{b}{2}} \end{cases} \tag{1.19}$$

3. **Feedback shift register generators**

   Linear Feedback Shift Register generators (LFSR) or Tausworthe are linear recurrent generators [74]. The diagram of the general shift register with feedback is given in Figure 1.8, each of the square labeled $x_i, x_{i+1}, ..., x_n$ is a binary storage element like Flip-flop, position on a delay line, or other memory device.

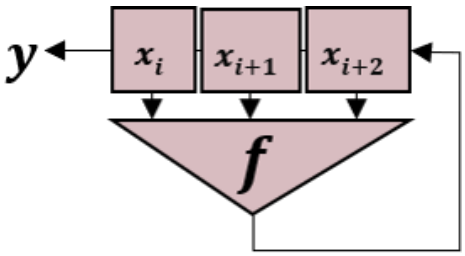


Figure 1.8: General diagram of the feedback shift register

   At periodic intervals determined by a master clock, the content of x_2 is transferred into x_{i+1}. However, to obtain a new value for location x_1, we compute some function $f(x_2, x_2, ..., x_n)$ of all the present terms in the shift register and use this in x_1.

4. **Blum–Blum–Shub generators**

   Blum–Blum–Shub generator (BBS) is a nonlinear and cryptographically secure PRNG [75]. It is based on the quadratic residue problem $x^2 = q \mod w$, where q is the "quadratic residue".

   It works as follows: consider $n = p \times q$, where p and q are prime numbers that are congruent to 3 mod 4. Let x^0 be an integer lower than w, which operates as a seed of the BBS generators [75]. Consider now the recurrent sequence $x^{t+1} = (x^t)^2 \mod w$, and $j = \left\lfloor \log_2(\log_2(w)) \right\rfloor$, where $\lfloor x \rfloor$ is the integral part of x. Then, at iteration t, the BBS generator outputs the j least significant bits of x^t [74].

5. **Chaotic generators**

   Chaotic generators, named also as chaotic pseudo random number generators (CPRNGs), are non-linear generators of the form $x^0 \in R : x^{t+1} = f(x^t)$, where f is a chaotic map [74].

John von Neumann [82] initially presented the idea of a pseudo-random number generator based on chaos in 1946. Although it produces a poor pseudo-random sequence, it is nonetheless regarded as a significant turning point in the development of pseudo-random generators. A generalized scheme for producing pseudo-random numbers using an iterated function f and a control parameter k is shown in Figure 1.9. Chaotic series modification may be necessary when a chaotic map exhibits periodic or predictable behavior.

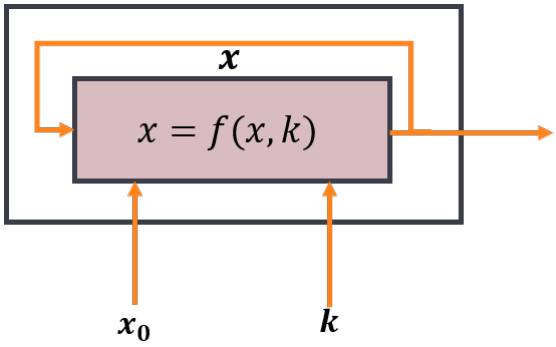


Figure 1.9: Pseudo-chaos generator based on an iterated function f with control parameter k

1.4.3 Statistical tests of random number generators

The security of a cryptographic system can be compromised by feeble or predictable random numbers; therefore, the quality of the random number generator is critical [83]. It is mathematically challenging to establish significant aspects of a generator's quality [83]. Therefore, the ultimate evaluation of PRNG quality is empirical [83].

There are numerous methods for assessing the quality of a random number generator, such as hardware evaluation and statistical tests. Statistical tests comprise the examination of the RNG's output to ascertain whether it demonstrates specific statistical properties that are intrinsic to numbers that are genuinely random. Approximations of unpredictability, independence, and uniformity can be incorporated into these tests. Hardware evaluations comprise a physical inspection of the hardware utilized in the generation of random numbers to ascertain its proper operation, and the absence of any defects or vulnerabilities that could compromise the security of the RNG [84]. Famous statistical tests are: the Diehard Battery of tests, PractRand, ENT test, RaBiGeTe, and the NIST test suite.

NIST test is a statistical package that tests the randomness of (arbitrarily long) binary sequences [85]. It consists of 15 tests, each of which evaluates the necessary condition for

randomness in probabilistic terms. The test suite calculates P_{value}, the probability that a perfect random number generator would have produced a sequence less random than the sequence that was tested. A significance level (α) can be chosen for the tests. If $P_{value} \geq \alpha$, then the null hypothesis is accepted, otherwise null hypothesis is rejected [86]. In Figure 1.10, a flow diagram for the randomness assessment of RNGs when using the NIST is shown for the case in which $\alpha = 0.01$ [84].

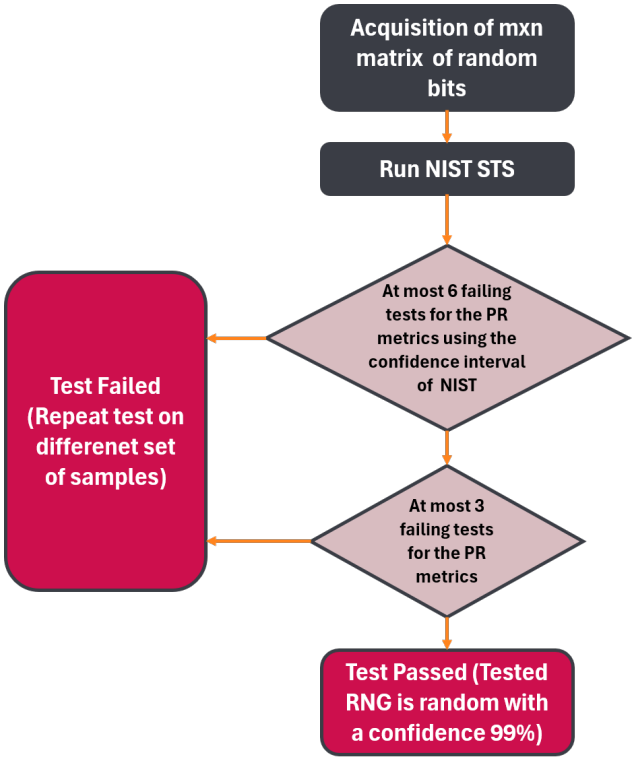


Figure 1.10: Flow diagram for randomness assessment using NIST test suite for a significance level of $\alpha = 0.01$ [84]

1.5 Conclusion

The need for image security stems from protecting the data contained in the image, including preventing accidental loss and damage, preventing unauthorized access or deliberate alteration of data, and encryption is generally considered to be the core method of protecting images from passive and active attacks. In this chapter, we have reviewed a number of contemporary methods of image encryption. We also discussed methods for generating random numbers.

In the next chapter, the nonlinear dynamics of basic chaotic systems are reviewed, and we

explore the unique attributes of some chaotic maps, such as sensitivity to initial conditions and pseudorandomness, for improving RNG quality.

Chapter 2

Chaos-Based Random Number Generator

2.1 Introduction

Random numbers are needed in a wide variety of commercial and scientific calculations, such as Monte Carlo computation and simulation, industrial testing and labeling, hazard games, gambling, statistical research, randomized algorithms, lotteries, and cryptography. Generators based on chaos have shown promising results in terms of security and unpredictability, these techniques have exhibited superior performance compared to traditional methods. The generation of pseudorandom numbers based on chaos theory is the subject of this chapter. A rich and current list of chaotic maps is presented at first, along with complete technical information on their definitions and classifications. A comparison of the statistical proprieties of two coupled chaotic maps via the usual NIST batteries of tests is then outlined in order to select the appropriate and extremely nonlinear pseudorandom number generator for use as secret key in the encryption process .

2.2 Fundamentals of nonlinear dynamics and chaotic systems

The chaotic behavior of nonlinear systems was one of the most productive contributions to physics and applied mathematics in the latter half of the twentieth century. It has matured into a mature field that keeps researchers engaged in many areas of science. In this section, we focus on the fundamentals of chaotic systems.

2.2.1 Dynamical system

The term "dynamical system" can describe a wide range of processes and can be applied to seemingly all areas of science [87]. Dynamics is defined as the study of the time-evolutionary process, and a dynamical system is a set of variables, functions (rules, equations), or quantities, whose values change with time according to some predefined rules [88].

Dynamical systems can be classified according to the nature of time into continuous and discrete-time processes.

- **Continuous time system**
  Let $x = x(t) \in \mathbb{R}^n$, $t \in I \subseteq \mathbb{R}^n$ be the vector representing the dynamics of a continuous time system. Continuous systems are defined by a set of coupled ordinary differential equations (ODEs) [88], defined by Equation 2.1.

$$\dot{x} \equiv \frac{dx}{dt} = f(x, t) \tag{2.1}$$

  The variable t is usually interpreted as time, and $f(x, t)$ is a sufficiently smooth function that takes the current state and returns the rate of change of the state.

- **Discrete time system**
  The discrete-time systems are usually given by iterated maps as in Equation 2.2.

$$x_{(n+1)} = f(x_n) \tag{2.2}$$

  f is the dynamic rule or equation of motion specifying the temporal evolution of the system. Here time is a discrete quantity, like steps, iterations, generations or other similar concepts. The state x_n is plugged into f to yield the state at the next step x_{n+1}, e.g., $x_1 = f(x_0)$, $x_2 = f(x_1)$ [87].

2.2.2 Nonlinear systems

Linear systems can be characterized by a set of ordinary differential equations or difference equations, and closed-form formulations for their solutions can be constructed [89]. A system is linear if it displays the traits of superposition and homogeneity (occasionally, superposition is understood to include homogeneity) as defined by Equation 2.3 as follows:

$$f(\alpha x + \beta y) = \alpha f(x) + \beta f(y) \qquad \alpha, \beta \in \mathbb{R} \tag{2.3}$$

Therefore, any function that does not fulfill superposition and homogeneity is nonlinear [89].

Nonlinear dynamics pertain to systems whose dynamics are regulated by nonlinear algebraic or nonlinear differential equations [90]. A nonlinear system may display chaotic dynamics if its dynamics are on average exponentially sensitive to changes in its starting state [90].

2.2.3 Fundamentals of chaotic dynamics

Chaos theory is centered on the development of encryption sequences as opposed to algorithms, and it generates extremely random sequences depending on the appropriate choice of a chaotic system. Compared to traditional encryption techniques, chaotic image encryption is more effective and safer [91]. Chaos theory is a mathematical topic of study that asserts that seemingly random nonlinear dynamical systems are predictable based on simpler equations.

Chaos is derived from the Greek word "Khaos", which means "gaping void" [92]. In other words, chaos refers to a condition of complete bewilderment or predictability in the behavior of a complex natural system.

Chaotic systems are physical system that exhibit high sensitivity to initial conditions [93], they are unstable because they do not resist external disruptions, and instead react in important ways [94]. In general, and according to the Devaney, chaotic systems have the following superior qualities resumed in Table 2.1 [95].

Table 2.1: Properties of chaotic dynamics

Chaotic properties	**Meaning**
Randomness	Chaotic systems produce chaotic sequences that resemble random.
Initial condition sensitivity	Their behavior may alter noticeably even with the smallest disturbance or change. A high sensitivity to the initial value is one of the key characteristics of these systems.
Ergodicity	All states in the phase space are evenly represented by each state variable.
Nonlinearity	There is no linear link between the input and output.
Deterministic	Mathematical equations may be used to represent and regulate the process quantitatively, and they can also be used to approach it quantitatively to some degree [96].

These systems are deterministic and can be mathematically modeled.

2.2.4 Classification of chaotic maps

Despite advancements in cryptography, the encryption of sensitive data using chaotic maps (the architecture of which is described in Figure 2.1) remains an efficient and popular process. Discrete and continuous chaotic systems are two basic categories of chaotic processes. The core of a chaos-based cryptosystem is labeled as chaotic maps [97]. The dimensions of the chaotic maps, are further classified as one-dimensional or multidimensional.

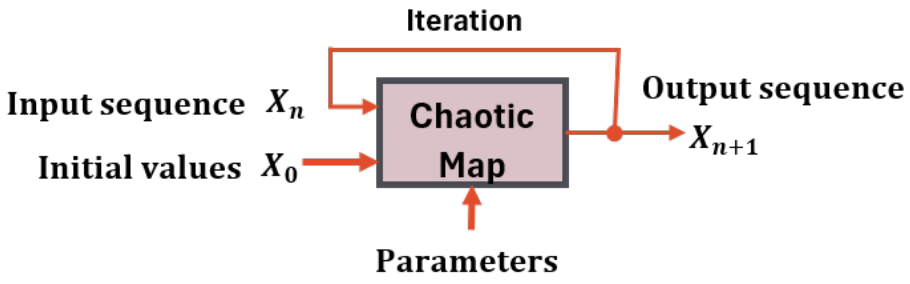


Figure 2.1: Chaotic map structure diagram

Its sensitivity to the initial circumstances, that is, the observed output, changes drastically even when the original settings are barely tweaked, making it an ideal choice for many cryptography applications. Other important characteristics of chaos include mixing, ergodicity, and nonperiodicity as well as unpredictability and pseudorandomness. Thus, several types of chaotic systems with varied dimensions have been proposed [98].

1. **One dimensional chaotic maps**

   One-dimensional chaotic systems are advantageous for practical applications because they have a simple chaotic structure, are easy to implement using hardware and software, and have good data characteristics [99]. However, they also have the drawbacks of limited vulnerability ranges of chaotic behaviors, nonuniform data distributions of output data sequences [100], and vulnerability to attacks.

   The tent mapping method, logistic mapping, and the Chebyshev algorithm are typical examples of low-dimensional chaotic encryption algorithms.

   (a) **Logistic map:** logistic map came under the category of the easiest and popular maps [101], developed by Robert May [102] in 1976, and defined by Equation 2.4

[103]. Where $X_n \in [0,1]$ for $n = 0, 1, 2, ...$ and $r \in [3.5699456, 4]$ is a control parameter to present the chaotic behavior. Parameters of logistic map (r, X_n) represents the initial conditions.

$$X_{n+1} = rX_n(1 - X_n) \tag{2.4}$$

(b) **Tent map:** one of the most well-known and utilized chaotic maps for generating pseudorandom numbers is the tent map [104] defined by Equation 2.5 [103]. Where μ is in the range of $(0, 4]$.

$$X_{n+1} = \begin{cases} \mu\frac{X_n}{2} & \text{for} \quad X_n < 0.5 \\ \mu\frac{1-X_n}{2} & \text{for} \quad X_n \geq 0.5 \end{cases} \tag{2.5}$$

(c) **Sine map:** the sine map defined by Equation 2.6 is based on the sine iteration function [105]. Where $0 \leq a \leq 1$, $X_n \in [0, 1]$. The produced series has a tendency to be chaotic when $a = 1$, but it may or may not be chaotic for other values of a [104].

$$X_{n+1} = a \sin(\pi X_n) \tag{2.6}$$

(d) **Piecewise linear map:** Piecewise linear chaotic map described by Equation 2.7 may provide a large random sequence that is suitable for information encryption, because it has extremely good ergodicity, confusion, and determinacy, as well as a uniform invariant distribution.

$$X_{n+1} = \begin{cases} \frac{X_n}{p} & \text{for} \quad X_n \in [0, p] \\ \frac{X_n - p}{0.5 - p} & \text{for} \quad X_n \in [p, 0.5] \\ F(1 - X_n, p) & \text{for} \quad X_n \in [0.5, 1] \end{cases} \tag{2.7}$$

Where the initial value of $X \in [0, 1]$ and $p \in [0, 0.5]$. The values of p and X, which act as secret keys, can be specified by the users [106].

(e) **Chebyshev map:** Chebyshev map is a symmetrical region map that generates chaotic sequence [107] defined by:

$$X_{n+1} = \cos \rho \arccos X_n \tag{2.8}$$

where $\rho > 0$, and $X_n \in [-1, 1]$.

2. **Multidimensional chaotic maps**

High-dimensional systems with complex dynamical properties, such the Chen system,

Lorenz system, and 3D LM, have been presented by researchers to get beyond the drawback of low-dimensional schemes. Phase-space reconstruction and other decryption attacks can be successfully countered by expanding the dimensions of chaos systems, but these systems are vulnerable to plaintext assaults.

(a) **Arnold's cat map:** the 2D Arnold's cat represented by Equation 2.9, is a well-known discrete chaotic map that uses P repeated shear mapping on the input image to scramble image pixels in order to recover the original image. The periodicity of the transformation map is specified as the parameter P. N is the dimensional value of the image, and a and b are two positive integers that act as control parameters.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & a \\ b & ab+1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \mod N \tag{2.9}$$

(b) **Henon map:** Henon mapping is the simplest two dimensional nonlinear mapping that has chaotic attractors [108], its mathematical expressions are defined by Equation 2.10, where α and β are parameters, when $\alpha \in [1.07, 1.4]$, $\beta = 0.3$.

$$\begin{cases} X_{n+1} = 1 + \beta Y_n - \alpha X_n^2 \\ Y_{n+1} = X_n \end{cases} \tag{2.10}$$

(c) **Lorentz:** Lorentz system is traced by Equation 2.11 [107], where, r and b are control parameters to bring chaotic behavior with $\sigma = 10$.

$$\begin{cases} \frac{dx}{dt} = \sigma(y - x) \\ \frac{dy}{dt} = -zx + rx - y \\ \frac{dz}{dt} = xy - bz \end{cases} \tag{2.11}$$

(d) **3-D logistic map:** A 3D logistic map defined by Equation 2.12 with superior chaotic properties compared with a 1D logistic map has recently been investigated.

$$\begin{cases} x_{i+1} = \gamma(1 - x_i) + \beta(y_i^2 x_i) + \alpha z_i^3 \\ y_{i+1} = \gamma(1 - y_i) + \beta(z_i^2 y_i) + \alpha x_i^3 \\ z_{i+1} = \gamma(1 - z_i) + \beta(x_i^2 z_i) + \alpha y_i^3 \end{cases} \tag{2.12}$$

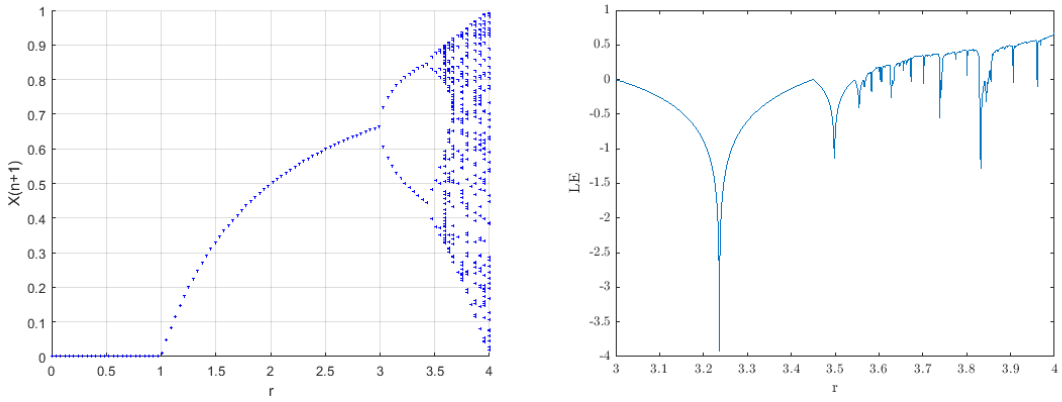
This system of equations exhibits chaotic behavior in $3.53 < \gamma < 3.81, 0 < \beta < 0.022, 0 < \alpha < 0.015$ [109].

2.2.5 Tests for chaos detection

Chaos theorists have developed multiple chaos detection techniques to examine the behavior of dynamic systems in several dimensions.

1. **Bifurcation diagram**

   The bifurcation diagram detects periodic cycles between chaotic and periodic orbits as functions of system control settings [110]. In brief, a bifurcation diagram like the one down in Figure 2.2a means that as we increase the parameter r, the steady-state behavior of the system changes in the following way (see Figure 2.3):



(a) (b)

Figure 2.2: Testing chaotic maps (LM) via: (a)Bifurcation diagram; (b)Lyapunov exponent

Here, "complex oscillations" refer to the oscillatory behavior with higher periods, after one or more period-doubling bifurcations have occurred. Recall that the change from equilibrium to oscillations is a "Hopf bifurcation", and the change from regular oscillations to complex oscillations is a "period-doubling bifurcation".

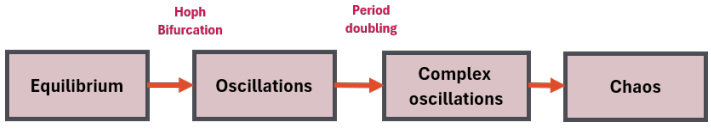


Figure 2.3: Period-doubling route to chaos

Also, within the region of complex oscillations, infinitely many more period-doubling bifurcations occur, closer and closer together. This is sometimes referred to as a "period-doubling cascade", which then leads to chaotic behavior.

2. **Lyapunov exponent**

   The Lyapunov exponent (LE) is the principal criterion of chaos and represents the growth or decline rate of small perturbations along each main axis of the phase-space system [111]. The LE of the discrete chaotic map $f(a_n)$ is defined as follows:

   Definition 2.2.1 (Lyapunov exponent) *Consider a differentiable map $f : \mathbb{R} \to \mathbb{R}$. The Lyapunov exponent (LE) of f for an orbit a_n is defined by Equation 2.13 [112]*

$$\sigma(a_1) = \lim_{n\to\infty}(\frac{1}{n}\sum_{i=0}^{n-1}|f'(a_i)|) \tag{2.13}$$

   Provided this limit exists.

   According to Theorem 2.2.1, a positive value of the Lyapunov Exponent indicates that the chaotic map has excellent chaotic properties.

   Theorem 2.2.1 *If at least one of the average Lyapunov exponents is positive, then the system is chaotic; if the average Lyapunov exponent is negative, then the orbit is periodic and when the average Lyapunov exponent is zero, a bifurcation occurs [113].*

3. **0-1 Test**

   0-1 test is chaos detection test that is used to distinguish between chaotic and regular behavior [114]. It uses 1D time series $\phi(n)$ for $n = 1, 2...$ as the input. We use the data $\phi(n)$ to drive the 2-dimensional system and $C_n \in (2, 2\pi)$.

$$\begin{cases} p(n+1) = p(n) + \phi(n)\cos C_n \\ q(n+1) = q(n) + \phi(n)\sin C_n \end{cases} \tag{2.14}$$

   Under general conditions, the limits $M(n)$ in Equation 2.15 defines the (time-averaged) mean square displacement [115], and the growth rate K described by Equation 2.16 is either 0 signifying regular dynamics or 1 signifying chaotic dynamics [114].

$$M(n) = \lim_{n\to\infty}\frac{1}{N}\sum_{j=1}^{N}([p(j+n) - p(j)]^2 + [p(j+n) - p(j)]^2) \tag{2.15}$$

$$K = \lim_{n\to\infty}\frac{\log M(n)}{\log n} \tag{2.16}$$

   The system trajectories are usually constrained in the regular case (periodic or quasi-periodic dynamics), but they often behave roughly like a two-dimensional Brownian motion with zero drift in a chaotic situation, evolving diffusively [116]. One easy way to identify whether these growth rates are diffusive or bounded is to use the mean square displacement $M(n)$, which increases linearly or is limited [116][114].

4. **3-State test**

   The 3 state test (3ST) is based on data series pattern analysis. The approach determines whether the dynamics are chaotic or regular by looking at the properties of periodic and quasi-periodic signals [117]. The 3ST looks at how a data series distribution of system states changes over time[118]. It is aimed at discriminating between the three major dynamics represented by the LE chaotic (> 0), quasi-periodic (< 0), and periodic ($= 0$) dynamics [119].

5. **Other chaos detection techniques**

   Other chaos detection techniques are described briefly in Table 2.2.

Table 2.2: Chaos detection techniques

	Detection technique	**Purpose**
LD	Lyapunov dimension	Quantifies the dimension of the chaotic attractor[93].
PP	Phase portrait	Appreciate the complexity and dynamics of a dynamical system.
SE	Simple entropy	Characterize the degree of complexity of a time series set without any prior knowledge of dynamic system that generated the dataset.
PE	Permutation entropy	Provide a quantification measure of the complexity of a dynamic system by capturing the order relations between the values of a time series and extracting the probability distribution of ordinal patterns.
AE	Approximate entropy	Determining the regularity of series of data based on the existence of patterns.
KE	Kolmogorov-Sinai entropy	Measure the long-term unpredictability of a motion by testing the degree of information loss in the motion.
JE	Joint Entropy	Characterizes the uncertainty and randomness of several signals.
CD	Correlation dimension	Measure the space dimensionality occupied by a time series.
TS	Time series	A statistical analysis of the dynamic system that represents the features of the data series.
PCM	Poincare map	Studying the fow of a system near a periodic orbit or a chaotic system.
FP	Fixed points	Reflect visually the dynamic properties of a nonlinear system.

2.3 Randomness evaluation of coupled chaotic maps via NIST tests

Coupled chaotic maps have been proposed as a method for generating pseudo-random numbers. The use of coupled map with the appropriate control parameters allows for the generation of independent and random sequences. The complexity of the generated sequences can

be increased by using a several structure with coupling factors between these maps.

Given the example of the logistic-sine system (LSS) and the logistic-tent system (LTS) proposed by Zhou et al.[12], which are a nonlinear mixture of a single logistic map, sine map, and tent map. Their main structure is illustrated in Figure 2.4.

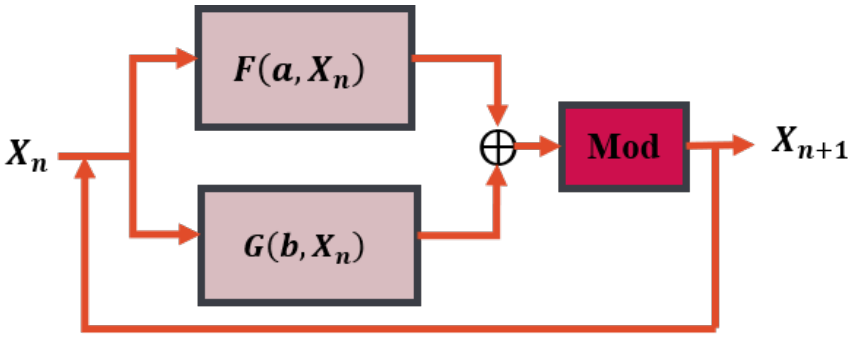


Figure 2.4: General structure of 1D coupled maps[12]

LSS and LTS are defined respectively by Equation 2.17 and 2.18 as follows:

$$\begin{aligned} X_{n+1} &= F_{LSS}(r, X_n) \\ &= L(r, X_n) + S((4-r), X_n) \mod 1 \end{aligned} \tag{2.17}$$

$$\begin{aligned} X_{n+1} &= G_{LTS}(r, X_n) \\ &= L(r, X_n) + T((4-r), X_n) \mod 1 \\ &= \begin{cases} rX_n(1-X_n) + (4-r)\frac{X_n}{2} \mod 1 & X_n < 0,5 \\ rX_n(1-X_n) + (4-r)(1-X_n)/2 \mod 1 & X_n \geq 0,5 \end{cases} \end{aligned} \tag{2.18}$$

where $r \in (0; 4]$

They are used for random number generation purposes, providing confusion and diffusion operations to improve security. These compound chaotic system exhibit more densely distributed chaotic behavior, larger key space, and key change, resulting in higher security and resistance to attacks.

Where a vital requirement for any random number generator based on chaos is to ensure that the generated sequence always benefits from a significant level of randomness. It is critical to examine such sequences by means of Lyapunov exponents, bifurcation diagrams, or other tests to accurately select the parameters of the dynamic system. However, the sequence's randomness quality varies depending on the generator's design and must be examined in different ways.

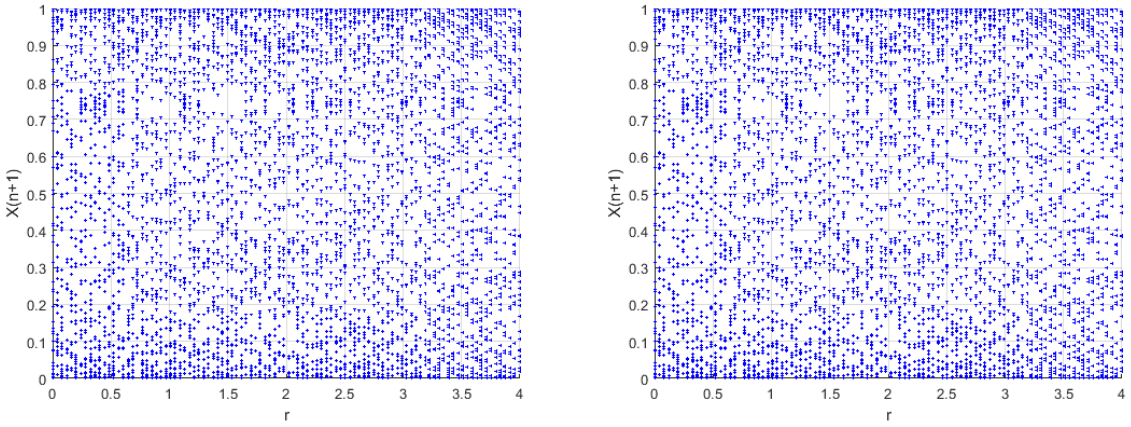
In [120], the 0-1 test and the three -state tests were used to provide a thorough investigation of the behavior of the LSS and LTS. While Zhou et al. [12] have proven that the LSS and LTS exhibit continuous chaotic behavior in the range $r \in (0, 4]$, Muthu et al. [120] portray stronger and weaker regions of chaos, with some regions exhibiting quasiperiodic behavior.

Therefore, we use of the National Institute of Standards and Technology (NIST) suite test to evaluate and compare the randomness properties of these two coupled systems, to demonstrate whether the system that has strong chaotic behavior is nominated to produce high -quality randomness, and to select the best and the appropriate coupled map for use as random number generator in our next contributions.

2.3.1 Analysis of the comportment behavior of the LSS and LTS from the literature

In this section, we cover and analysis numerous tests that are applied in the literature to evaluate the chaotic behaviors of the LSS and LTS. The results of each test is valuable for the foundation of our study.

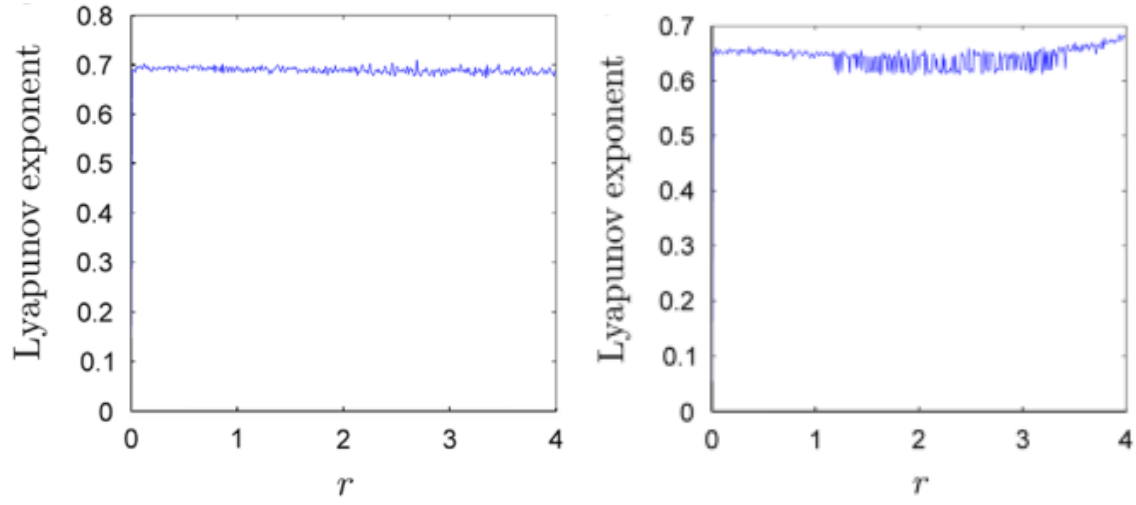
First, the output sequences $X(n + 1)$ of the chaotic systems are analysed by [12] and plotted along with the change in parameter r. Figure 2.5a and 2.5b compare the bifurcation diagrams of the LSS and the LTS. From these figures, it is obvious that the chaotic range of the LSS and LTS is inside $(0, 4]$, and their bifurcation behavior is evenly dispersed over the full space from 0 to 1. Visually, these findings are not sufficient for comparing and determining zones of chaos and regularity; consequently, identifying these zones using only bifurcation diagrams is difficult. Classification tests are useful allies for dealing with this kind of situation more clearly [121]. In general, classification by Lyapunov exponents is the most commonly used approach in the literature [122].



(a) (b)

Figure 2.5: Bifurcation diagram: (a) logistic-sine system; (b) logistic-tent system

Thus, the Lyapunov exponents of the LSS and LTS are tested also in [12], as shown in Figure 2.6. Visually, the LSS and LTS have more complex chaotic qualities, as evidenced by their Lyapunov exponents, which are greater than 0 over the whole parameter setting range r, and they consistently behave chaotically in the range $r \in (0, 4]$ [120]. However, these results are not sufficient for comparing their chaotic behavior, since the maximal Lyapunov exponent of each system is not calculated.



(a) (b)

Figure 2.6: Lyapunov exponent diagram: (a) Logistic-sine system; (b) Logistic-tent system [12]

Third, the 0-1 test was experimented by Muthu et al.[120] on the LSS and the LTS with

parameters $N = 5000$ and $X_0 = 0.01$. The K values obtained for the r values are shown in Figure 2.7, which demonstrates a slope towards 1 for all values of r in the range $[3.1, 4]$ for the LSS and LTS, demonstrating that these maps do not have a consistently chaotic character over the given range. Furthermore, Muthu et al.[120] demonstrate that the LSS possesses the strongest chaotic nature in most areas of r.

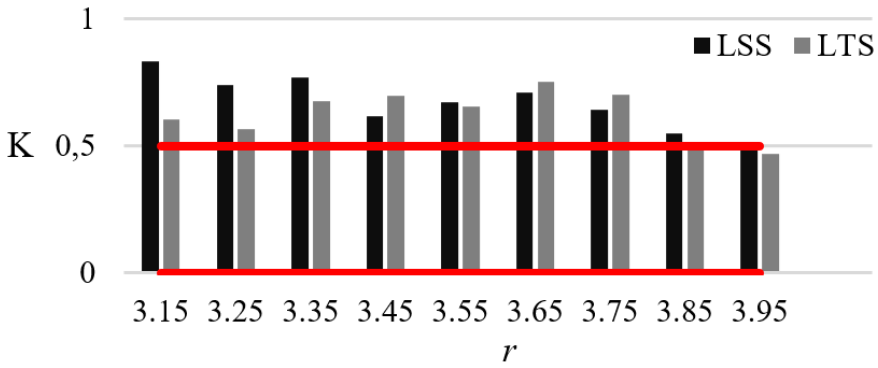


Figure 2.7: 0-1 Test results: K values obtained for the r values

At last, Muthu et al. [120] performed 3ST on the LSS and LTS in the range $r \in [3.1, 4]$. Surprisingly, three types of behavior are clearly differentiated at various r values; periodic, quasi-periodic, and chaotic, as shown in Table 2.3, which illustrate clearly that the LTS has a wider chaotic region than the LSS. Furthermore, the chaotic behavior of these maps is not uniformly distributed, and certain parts of the LSS and LTS are found to be quasiperiodic. This conclusion refutes what is asserted in [12] that LSS and LTS are chaotic throughout.

Table 2.3: Results of behavior comparison of the LSS and LTS using 3ST

r	**LSS**	**LTS**
$[3,1-3,19]$	Chaotic	Quasi-periodic
$[3,2-3,29]$	Chaotic	Quasi-periodic
$[3,3-3,39]$	Chaotic	Chaotic
$[3,4-3,49]$	Quasi-periodic	Chaotic
$[3,5-3,59]$	Quasi-periodic	Chaotic
$[3,6-3,69]$	Chaotic	Chaotic
$[3,7-3,79]$	Quasi-periodic	Chaotic
$[3,8-3,89]$	Quasi-periodic	Quasi-periodic
$[3,9-3,99]$	Quasi-periodic	Quasi-periodic

2.3.2 Experimental study and results

In order to further study and compare the random properties of the chaotic sequence generated by the LSS and LTS, the NIST test is used in this section to identify the areas of randomness and lack of randomness of these two systems, and to confirm findings in [12] and [120].

First, the LSS and LTS systems are utilized to construct the chaotic series $X(n+1)$ with control parameters r in the range $(0,4]$ and $X_0 = 0.1$ using the iterative procedures specified in Equation 2.17 and 2.18, respectively. The bit length of each sequence n was set to $1000bits$. Then, the statistical tests were performed using NIST SP 800-22. The 15 subtests that make up the NIST test may all be used to assess the randomness of the sequences. By analyzing the sequence's uniformity, the test results largely show the benefits and drawbacks of the pseudorandom sequence [123], in which the probability value (P_{value}) reflects the regularity of the sequence. The P_{value} of each subtest is compared to a tester-determined significance threshold (which, for cryptography and in the case of NIST test suite version SP800-22, is commonly set to $= 0.01$). If P_{value} is greater than α, the sequence is random; otherwise, the sequence is not random [124]. Since we cannot determine which maps exhibit superior randomness simply by performing one test for just one value of r, we have to repeat the test according to r's transitions from 3.15 to 3.95, and then we have calculated the probability that a random sequence fails one or more tests for each testing process. Figure 2.8 depicts the histogram plot of the uniformity test P_{values} at three values of r: 3.15, 3.65, and 3.95

for the two maps LSS and LTS. Table 2.4 shows the results of failed tests for all r values mentioned.

Table 2.4: Failed tests relatively to 15 tests for r listed values

r **value**	**LSS**	**LTS**
$[3,1-3,19]$	$4/15 \Rightarrow Random$	$8/15 \Rightarrow NotRandom$
$[3,2-3,29]$	$3/15 \Rightarrow Random$	$7/15 \Rightarrow NotRandom$
$[3,3-3,39]$	$3/15 \Rightarrow Random$	$3/15 \Rightarrow Random$
$[3,4-3,49]$	$3/15 \Rightarrow Random$	$8/15 \Rightarrow NotRandom$
$[3,5-3,59]$	$3/15 \Rightarrow Random$	$7/15 \Rightarrow NotRandom$
$[3,6-3,69]$	$3/15 \Rightarrow Random$	$4/15 \Rightarrow Random$
$[3,7-3,79]$	$3/15 \Rightarrow Random$	$6/15 \Rightarrow Random$
$[3,8-3,89]$	$3/15 \Rightarrow Random$	$3/15 \Rightarrow Random$
$[3,9-3,99]$	$3/15 \Rightarrow Random$	$3/15 \Rightarrow Random$

The LSS findings exhibit excellent randomness, where all P_{values} are much over greater than the significance threshold for most r values, the expected binary matrix rank test, the overlapping template matching test, and Maurer's universal statistical test. It should be noted that some of these sub-tests are not always appropriate. These sub-tests are run only if the sequence meets certain criteria (for example, passes the frequency test, has more than 500 cycles [125], or has a sufficient bit-length). However, LSS remains regarded as random even if it fails 3 to 4 tests, according to [125], where data may still be deemed random at the significance level $\alpha = 0.01$ if they fail fewer than 7 NIST statistical tests.

LTS fails multiple tests when r is in the quasi-periodic range $[3.1 - 3.29]$ and in the chaotic range $[3.4 - 3.59]$. This might be explained by the fact that the randomness of the sequences does not rely only on the chaotic state of the underlying system but also on the post-processing and generator design. It is obvious that the randomization qualities of these maps in such a range have exposed major security needs, which make their usage inappropriate for image encryption and demand a solid selection of chaotic system parameters when employing them.

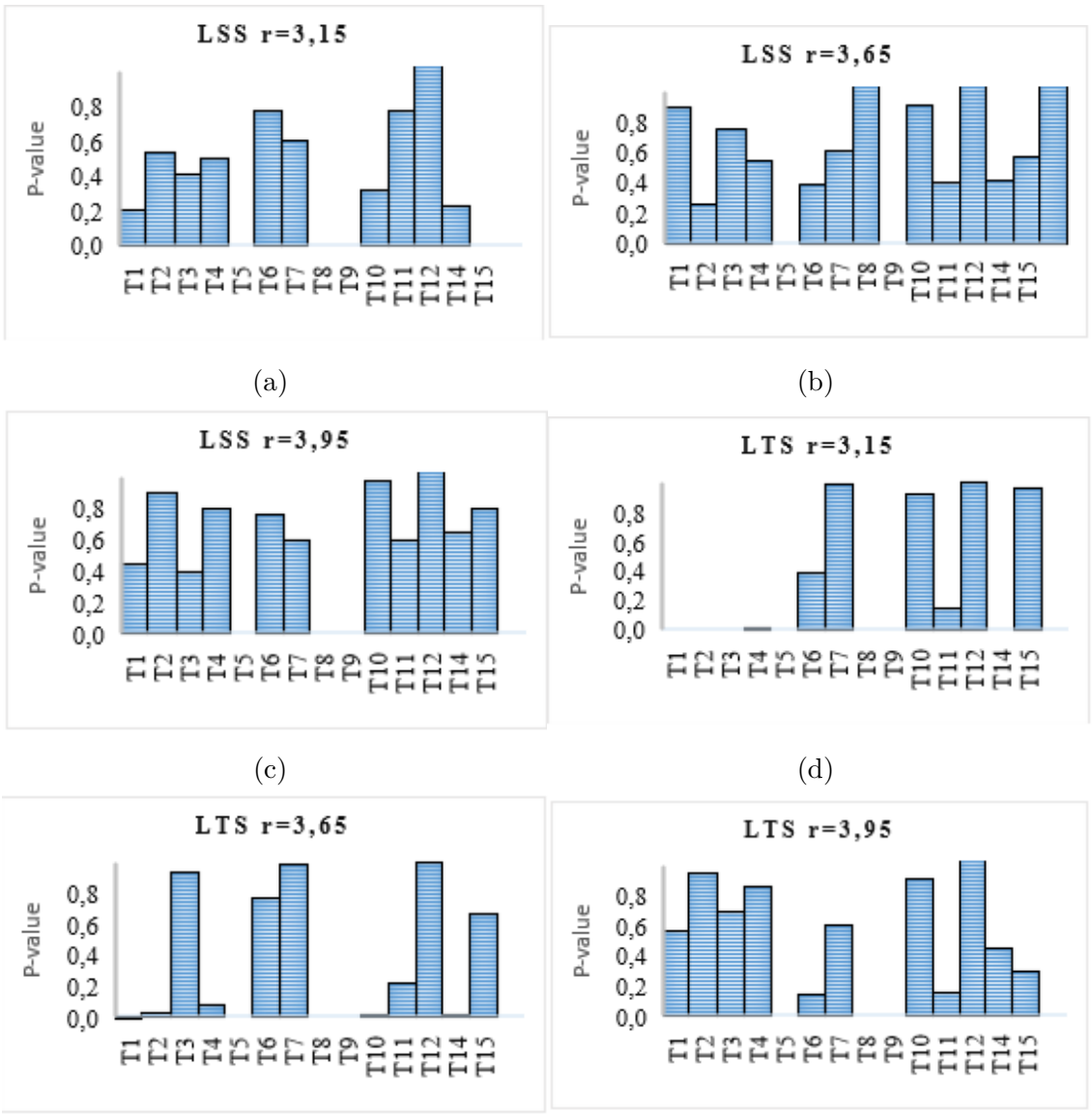


Figure 2.8: P_{value} of 15 statistical tests, the x axis lists the name of the statistical test in NIST test suit: T1- Frequency, T2- Frequency within a Block,T3- Runs,T4- Longest run of ones, T5- Rank,T6- Spectral,T7- Non-overlapping, T8- Overlapping, T9- Maurer's Universal, T10- Linear complexity, T11- Serial, T12- Approximate Entropy, T13- Cumulative sums, T14- Random Excursions, T15- Random Excursions Variant

Finally, the results revealed a strong relationship between the NIST test results and the chaotic metrics identified in the literature in the previous section. The seeds for which the maps are chaotic, are the seeds that determine a low number of failed NIST tests, which demonstrates the notion that a required criterion for a successful pseudo random number

generator is that the development of the underlying system is chaotic. Thus, according to [120], the LSS that has the strongest chaotic behavior in the range $[0-4]$ is the one with the fewest failed NIST tests, and has the ability to generate a highly random chaotic sequence.

2.4 Conclusion

The unique attributes of chaos, such as sensitivity to initial conditions, topological transitivity, and pseudorandomness, are conducive to designing and improving the output quality of pseudorandom number generators.

In this chapter, we provided an introduction to chaotic systems, their characteristics, and usual tests used for detecting their chaotic behavior. We also investigate the randomness of chaotic sequences generated by the coupled LSS and LTS via the NIST test; where we have reached that the LSS meet the strongest chaotic behavior that makes it capable of generating high random number sequence. Based on these results, we propose an effective integration of the best selected coupled chaotic map with the famous feedback shift register to improve chaos-based image encryption methods in the next chapter.

Chapter 3

Improved key-stream generator based on feedback shift register and chaos for image encryption

3.1 Introduction

Key-stream generators are methods or devices used to generate a sequence of random or pseudo-random numbers, known as a key-stream, which is then used for encryption and decryption purposes. These generators play a crucial role in ensuring the security and randomness of cryptographic algorithms. Traditional one-dimensional maps are vulnerable to attacks if they are not properly used as key stream generators. A coupled chaos map can solve such issues. In addition, their integration with other emerging technologies can open new avenues for practical application. However, the effective integration of these technologies requires further study. Feedback shift registers remain as the popular choice for integration with chaos due to their attractive feature and high efficiency when implemented in hardware or software. In this chapter, we design new key-stream generators for image encryption purposes built from feedback shift register and chaos, and we attempt to improve their output quality by introducing several combination of linear feedback shift register (LFSR), non linear feedback shift register (NLFSR), logistic map (LM), and the coupled logistic-sine map (LSS).

3.2 Problem formulation and our contribution

One-dimensional coupled chaotic maps have been proposed as random number generators in several previous works. These maps address the limitations of simple chaotic 1D maps by offering a wider range of control parameters and exhibiting complex and diverse behavior.

However, they still vulnerable to several attacks due to their discontinuous range of chaotic behavior as proved previously in chapter 2. Researchers have proposed various methods of integration of these maps with other schemes such as DNA encoding, cellular automata, and elliptic curve, at the aim of enhancing their security and making them suitable for cryptography applications.

As instance, FSR offers several advantages in various applications. It is commonly used as a pseudo-random number generator since it is capable to generate random and unpredictable sequence of keys. Its designs have been optimized to fit inside silicon substrates, making them suitable for tasks such as cryptographic keys, data whitening, and fast digital counters. Additionally, FSR-based designs have been implemented in field programmable gate array (FPGA) for Internet of things (IoT) and smart applications, providing effective results in terms of delay, speed, and area. Overall, FSRs offer benefits such as randomness generation, efficient implementation in complementary metal-oxide-semiconductor (CMOS) technology, and suitability for various applications in the field of integrated circuits and digital equipment. These characteristics makes it a worthy choice to integrate it with the used chaotic map.

In our contribution, we will focus on designing then improving a new key-stream generators based on the appropriate chaotic map with the appropriate feedback shift register. Thus, our study is based on four sub-generators; the simple 1D logistic map, 1D coupled LSS, LFSR, and NLFSR. Hence several combination are possible, three of them are chosen for the designed PRNGs, named also key-streams: LSS-LFSR-PRNG, LM-NLFSR-PRNG, and LSS-NLFSR-PRNG.

To ensure the quality of randomness of the proposed scheme, the NIST test suite is used alongside other tests. At last, the implementation of the proposed key-streams for encryption is investigated, showing suitability for secure image transmission.

3.3 Related studies

This section examines earlier efforts to generate pseudo-random number sequences using the FSR and chaos.

Beginning with Rohith et al. [126], they proposed an image encryption and decryption

technique based on a key sequence formed by a logistic map sequence and a second series of states of a LFSR. To acquire the final key sequence, the generated sequences are multiplied by 255, and a bit-by-bit XOR operation is performed on the states of an 8 bit LFSR. Therefore, the received key sequence is XORed with 8-bit grayscale image pixels to produce an encrypted image.

To create a random key that is challenging for hackers to guess, a novel form of multiple different generators, including LFSRs, NLFSRs, and feedback with carry shift registers (FCSRs), were merged in [127].

In [128], the authors proposed a hybrid random number generator (HRNG) based on a random signal generated by a chaotic oscillator and a LFSR using an XOR gate.

To boost the unpredictability of the produced sequences, Garcia et al.[64] has presented a quick and reliable encryption system with a mixed architecture based on a Modified Logistic Map (MLP) and a Linear Feedback Shift Register as a key stream generator.

The hardware implementation of a multibit and high-speed of RNG, LFSR-based PRNG that produces uniform distribution numbers was presented by Datta et al.[129], where hardware description languages (HDL) are used in the design of the PRNGs circuit.

3.4 Theoretical foundations of LFSR and NLFSR

Throughout this chapter, we use $\oplus$ and "." to denote addition and multiplication in $GF(2)$ respectively, and "+" to denote arithmetic addition. The NLFSR of n binary storage elements (represented by $x_i, i \in [0, n-1]$) is a generalised variant of the LFSR where the present state is a nonlinear feedback function of prior states. It can be viewed as finite state automata, as depicted in Figure 3.1. The bit values ${x_i}^t$, $i \in [0, n-1]$ stored in all the stages form the NLFSR's internal state [130], as revealed by $X^t = {x_0}^t, {x_1}^t, ..., {x_{(n-1)}}^t$ at clock t.

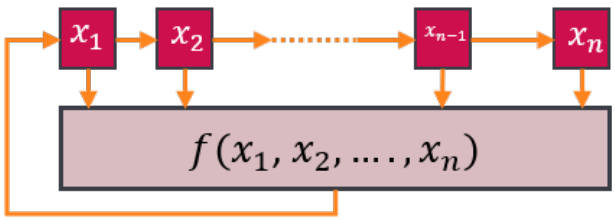


Figure 3.1: General diagram of the feedback shift register

There are two main ways in which an NLFSR can be constructed: the Fibonacci and Galois configurations [131] (see Figure 3.2). In the Fibonacci setup, the feedback is applied

at the final step, in contrast to the Galois setup, where feedback is applied at every step.

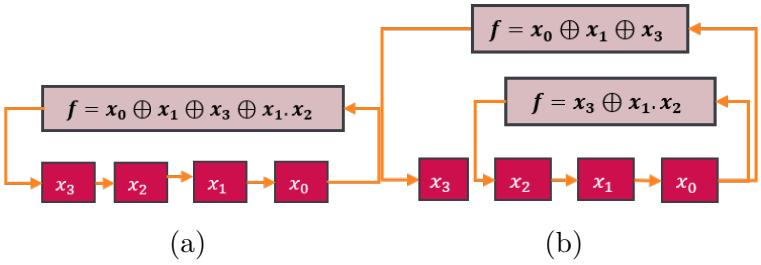


Figure 3.2: 4bit-NLFSR configurations: (a) Fibonacci; (b) Galois [130]

The feedback function or connection polynomial are additional names for the feedback path [132], and the degree of the feedback function utilized to create a series of keys is vital to the system security and must be properly set.

The feedback function $f(x_1, x_2, ..., x_n)$, computed from the content of the n stages, determines the next value of x_n. The output of an NLFSR is the sequence of bits appearing in its stage 1 [133].

The feedback function f induces the mapping $F : \{0,1\}^n \to \{0,1\}^n$ of type:

$$(x_1, x_2, ..., x_n) \to (x_2, x_3, ..., x_n, f(x_1, x_2, ..., x_n)). \tag{3.1}$$

The state of an n-stage register is a vector of values $S = (s_1, s_2, ..., s_n) \in \{0,1\}^n$ of its state variables $x_1, x_2, ..., x_n$ [133].

A cycle of length m of an $n-$stage register is a vector of states $(S_0, S_1, .., S_{m-1})$ such that $F(S_i) = S_{i+1}$, for $i \in \{0, 1, .., m-2\}$, and $F(S_{m-1}) = S_0$.

The period of a register is the length of its longest cycle [133]. The maximum possible period for an n-bit NLFSR is 2^n [134]

A necessary and sufficient condition for an NLFSR to be branchlessis that its feedback function f can be written in the form

$$f(x_1, x_2, ..., x_n) = x_1 \oplus g_i(x_2, ..., x_n) \tag{3.2}$$

where g is a Boolean function which does not depend on the variable x_1.

Implementation of the feedback shift register algorithms begins with the selection of a primitive polynomial.

The set of $n - bit$ Fibonacci NLFSRs with the period 2^{n-1} can be partitioned into 4 subsets: (1) basic, (2) reverse of basic, (3) complement of basic, and (4) reverse complement of basic. If the basic NLFSR has the feedback function of type (1), then reverse, complement, and reverse complement NLFSRs have the following feedback functions:

$$\begin{cases} f_r(x_0, x_{n-1}, ..., x_n) = x_0 \oplus g_i(x_{n-1}, x_{n-2}, ..., x_1) \\ f_c(x_0, x_{n-1}, ..., x_n) = x_0 \oplus 1 \oplus g_i(x_1, x_2, ..., x_{n-1}) \\ f_{rc}(x_0, x_{n-1}, ..., x_n) = x_0 \oplus 1 \oplus g_i(x_{n-1}, x_{n-2}, ..., x_1) \end{cases} \tag{3.3}$$

These NLFSRs generate sequences which are reverse, complement, or reverse complement of the basic sequence, correspondingly. Figure 3.3 below show an example list of feedback functions of n bit Fibonacci NLFSRs with the period 2^{n-1} and with the algebraic degree two (extended list is available on [134]).

n = 5:	n = 6:	n = 7:	n = 8:	n = 9:
0,1,2,(2,4)	0,1,2,(1,2)	0,1,2,(2,6)	0,1,5,(1,5)	0,1,6,(4,6)
0,1,3,(1,3)	0,1,2,(2,4)	0,1,4,(1,3)	0,1,6,(1,2)	0,1,6,(4,8)
0,1,3,(1,4)	0,1,3,(1,5)	0,1,5,(1,5)	0,1,6,(1,7)	0,2,4,(4,5)
0,1,3,(2,3)	0,1,4,(1,4)	0,1,5,(3,5)	0,1,6,(2,4)	0,3,4,(3,7)
0,1,(1,2),(2,3)	0,2,3,(1,3)	0,1,5,(4,6)	0,1,6,(4,5)	0,1,(1,5),(2,5)
0,1,(1,2),(3,4)	0,2,3,(1,5)	0,2,4,(1,2)	0,1,6,(5,6)	0,1,(1,6),(6,7)
0,1,(1,3),(2,3)	0,2,3,(2,3)	0,2,4,(2,5)	0,2,5,(2,4)	0,1,(1,8),(2,7)
0,1,(1,3),(2,4)	0,2,3,(2,4)	0,1,(1,2),(5,6)	0,2,5,(3,7)	0,1,(1,8),(5,6)
0,1,(1,4),(2,3)	0,1,(1,2),(4,5)	0,1,(1,5),(3,4)	0,2,5,(4,5)	0,1,(2,3),(3,8)
0,2,(1,2),(3,4)	0,1,(1,3),(3,5)	0,1,(1,6),(4,5)	0,3,4,(2,4)	0,1,(2,8),(3,7)
0,2,(1,3),(2,4)	0,1,(2,3),(2,5)	0,1,(2,3),(3,5)	0,3,4,(2,7)	0,1,(3,4),(3,5)
0,2,(1,4),(2,3)	0,2,(1,3),(2,4)	0,1,(2,5),(3,5)	0,3,4,(3,4)	0,1,(3,7),(5,8)
0,2,(1,4),(2,4)	0,2,(1,3),(3,4)	0,1,(2,5),(4,5)	0,3,4,(4,6)	0,2,(1,5),(4,6)
	0,2,(1,3),(3,5)	0,1,(3,4),(4,5)	0,3,4,(4,7)	0,2,(1,6),(2,7)
	0,2,(1,5),(2,4)	0,2,(1,2),(4,6)	0,3,4,(6,7)	0,2,(1,8),(3,4)
	0,2,(1,5),(4,5)	0,2,(1,4),(3,4)	0,1,(1,4),(2,4)	0,2,(2,7),(4,6)
	0,2,(2,3),(3,5)	0,2,(1,5),(2,6)	0,1,(1,6),(2,5)	0,2,(4,7),(5,6)
	0,2,(3,4),(3,5)	0,2,(1,6),(2,4)	0,1,(2,3),(2,4)	0,3,(1,2),(4,7)
	0,3,(1,4),(2,3)	0,2,(1,6),(3,6)	0,1,(2,4),(6,7)	0,3,(1,6),(1,7)
	0,3,(1,4),(2,4)	0,2,(1,6),(5,6)	0,1,(3,4),(4,7)	0,3,(1,7),(4,8)
	0,3,(1,4),(3,4)	0,2,(2,4),(3,5)	0,2,(1,3),(4,6)	0,3,(2,3),(4,7)
		0,2,(2,5),(4,6)	0,2,(1,3),(5,7)	0,4,(1,3),(2,8)
		0,2,(2,6),(4,6)	0,2,(1,5),(6,7)	0,4,(1,6),(3,6)
		0,2,(3,6),(5,6)	0,2,(1,7),(2,3)	0,4,(2,3),(5,8)
		0,3,(1,2),(2,3)	0,2,(3,7),(6,7)	0,4,(2,5),(2,8)
		0,3,(1,3),(1,6)	0,3,(1,2),(2,4)	0,4,(2,7),(3,8)
		0,3,(1,4),(3,6)	0,3,(1,4),(2,4)	0,4,(2,8),(6,7)
		0,3,(1,5),(3,5)	0,3,(1,6),(3,6)	0,4,(3,5),(3,7)
		0,3,(1,6),(3,4)	0,3,(1,6),(4,6)	0,1,2,3,4,(3,7)
		0,3,(2,3),(4,5)	0,3,(1,6),(4,7)	0,1,2,3,7,(4,6)
		0,3,(2,5),(3,5)	0,3,(2,3),(5,6)	0,1,2,4,7,(1,6)
		0,1,2,3,4,(1,6)	0,3,(2,4),(6,7)	0,1,2,5,6,(1,6)
		0,1,2,3,4,(2,3)	0,3,(2,6),(3,7)	0,1,2,5,6,(2,6)
		0,1,2,3,4,(2,6)	0,1,2,3,5,(2,6)	0,1,2,5,8,(2,6)
		0,1,2,3,6,(1,3)	0,1,2,3,6,(3,5)	0,1,2,6,7,(3,6)
		0,1,2,3,6,(1,5)	0,1,2,3,6,(5,7)	0,1,3,4,5,(3,7)
		0,1,2,3,6,(2,6)	0,1,2,4,5,(2,4)	0,1,3,5,7,(5,6)
		0,1,2,4,5,(1,2)	0,1,2,4,7,(1,5)	0,1,3,5,8,(3,5)
		0,1,2,4,5,(1,5)	0,1,2,5,7,(2,4)	0,1,4,6,7,(1,7)
		0,1,2,4,5,(2,6)	0,1,3,4,7,(1,4)	0,2,3,4,7,(2,8)
			0,1,3,4,7,(1,6)	
			0,1,3,4,7,(3,7)	

Figure 3.3: List of feedback functions of NLFSR for n=5,6,7,8,9 [134]

3.5 Design of secure key-stream generators based on FSR and chaos

In this section, we discuss different combinations possible for designing our key stream based on the feedback shift register and chaotic iteration as illustrated by Figure 3.4.

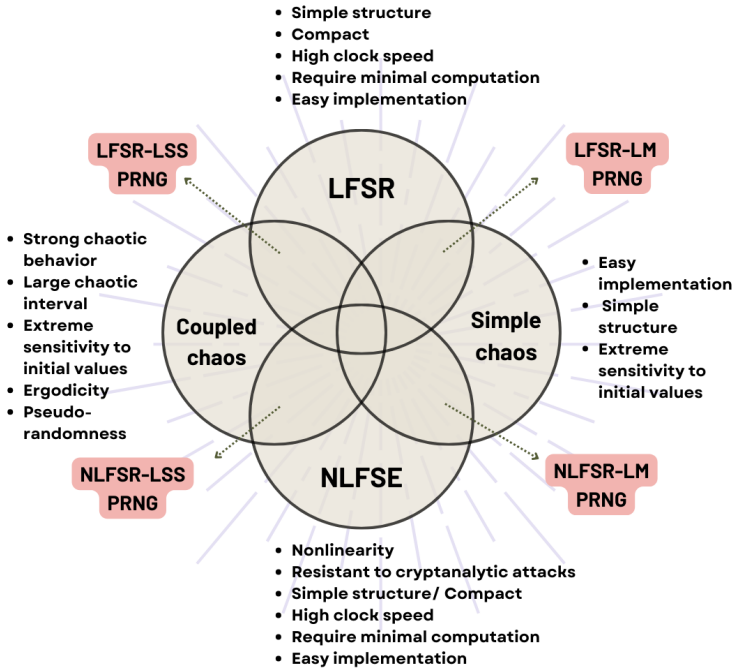


Figure 3.4: Possible combinations of key stream generators

3.5.1 Key-stream generator based on chaotic iterations

In the first step, two generators based on a simple one-dimensional chaotic map (LM) and a one-dimensional coupled map (LSS) are employed, and their properties are investigated to design our proposed generators LM-PRNG and LSS-PRNG. The choose of the appropriate chaotic maps is based on the comparative study of two coupled maps presented in chapter 2. These maps are the logistic map and the LSS presented respectively by Equation 2.4 and 2.17. By using their iterative equations, it is possible to build an endless sequence of random numbers.

The initial conditions and control parameters of the system are not set randomly, noting that little perturbation on the initial condition and control parameter may completely impact the random behavior of the system. The control parameter should be set within the chaotic range. For example, according to literature, the control parameter r of the logistic map that contributes to the excellent chaotic behavior of the function should belong to $[3.54 - 4]$. The same case with LSS, where the r value that we designate secret key1 is set according to a microscopic analysis of its chaotic behavior in the range $[3 - 4]$ done by Muthu et al.[120]. Based on [120], the best chaotic behavior of the LSS was observed in the range

$r \in [3.6 - 3.69]$. Therefore, we can utilize the starting value $X_0 = 0.1$ and r in that range to produce a sufficiently lengthy chaotic sequence.

Before that, an enhancement of the chaotic maps behavior is necessary. Since the chaotic sequence generator generates values between 0 and 1 with 10^{-15} decimals, we use Equation 3.4 to convert each value of the chaotic sequence X_n to an 8-bit integer.

$$X_i = round(X_i \times 255) \tag{3.4}$$

3.5.2 Key-stream generator based on FSRs

In this stage, two generators are designed based on an 8-bit Fibonacci LFSR (named $LFSR-PRNG$) and the Galois- NLFSR (of degree two then three)(named the $NLFSR_2 - PRNG$ and $NLFSR_3 - PRNG$) specified by the following basic polynomial respectively.

$$\begin{cases} f_{LFSR-PRNG}(x_0, x_1, .., x_{n-1}) = x_8 \oplus x_7 \oplus x_6 \oplus x_5 + 1 \\ f_{NLFSR_2-PRNG}(x_0, x_1, .., x_{n-1}) = x_0 \oplus x_1 \oplus x_6 \oplus (x_1.x_2) \\ f_{NLFSR_3-PRNG}(x_0, x_1, .., x_{n-1}) = x_0 \oplus x_6 \oplus x_7 \oplus (x_1.x_2.x_3) \end{cases} \tag{3.5}$$

The generated random sequences are periodic with a period of $2^8 - 1 = 255$. Note that in Equation 3.5 the XOR($\oplus$) operation is a linear function whereas the AND (.) is a non-linear function.

The two feedback shift registers are initiated with a random value called a seed. The register's past state may be used to determine its upcoming state deterministically. The tap arrangement and the seed value are the only factors that determine how long the cycle lasts. Therefore, in order to guarantee a very long cycle when using FSRs for random number generation, it is crucial to choose an adequate set of tap points and seeds. Hence, since the number of zeros and ones in the used feedback shift register initial states should be as equal as feasible, the default state of the shift registers is set to [10101101]. The output sequence is taken from the seventh register's tap.

3.5.3 The combined key-stream generators

The final sequence is achieved by carrying out a bitwise XOR operation between the random output of the $LM - PRNG$, $LSS - PRNG$, $LFSR - PRNG$, and $NLFSR - PRNG$ of equal lengths. Thus, we choose three combinations to be evaluated: $LSS - LFSR - PRNG$, $LM - NLFSR_2 - PRNG$, and $LSS - NLFSR_3 - PRNG$.

3.6 Performance analysis of the proposed key-stream generators

The security of an entire cryptosystem is significantly affected by the used key-stream, therefore, it is important to guarantee the security of the generated numbers used in these applications. In this part, we investigate and compare the randomness quality of our proposed key-streams, via the National Institute of Standards and Technology (NIST), which has set rules for random number generation. Then, the key sensitivity of the suggested key sequence is assessed by the Pearson's correlation coefficient and the Hamming distance.

3.6.1 Randomness evaluation via NIST

Table 3.1 and 3.2 list the NIST SP-800 test results obtained after processing 1000 binary sequences. The satisfactory findings from the NIST tests reveal that:

The simple generators based on $LM - PRNG$, $LSS - PRNG$, $LFSR - PRNG$, and $NLFSR - PRNG$, does not meet the randomness level requirements, since the majority of P_{value} for the 15 tests was < 0.01. In contrast to the sequence generated by $LSS - LFSR - PRNG$, $LM - NLFSR_2 - PRNG$, and $LSS - NLFSR_3 - PRNG$ is indeed a highly randomness properties, and does not have any statistical defects since P_{value} for the 15 tests are totally > 0.01.

Table 3.1: Results of the NIST test of randomness tested from the enhanced generated sequences

		LFSR-PRNG		**LSS-PRNG**		**LSS-LFSR-PRNG**	
n^0	**Test**	P_{value}	Conclusion	P_{value}	Conclusion	P_{value}	Conclusion
1	Frequency Test (Monobit)	0.20050	Pass	0.96011	Pass	0.13618	Pass
2	Frequency Test within a Block	2.07625 e228	Fail	$2.18769e-14$	Fail	0.24424	Pass
3	Run Test	0.26771	Pass	$2.62724e-131$	Fail	0.47077	Pass
4	Longest Run of Ones in a Block	$4.13916e-119$	Fail	$2.96302e-56$	Fail	0.03632	Pass
5	Binary Matrix Rank Test	$1.24995e-55$	Fail	0.03760	Pass	0.54427	Pass
6	Discrete Fourier Transform Test	$5.04413e-68$	Fail	$2.32810e-05$	Fail	0.52656	Pass
7	Non-Overlapping Test	$2.11149e-07$	Fail	$5.14249e-116$	Fail	0.86930	Pass
8	Overlapping Test	$1.098560e-10$	Fail	$5.86466e-28$	Fail	0.20111	Pass
9	Maurer's Universal Statistical test	-1.0	Fail	-1.0	Fail	-1.0	Fail
10	Linear Complexity Test	0.0	Fail	0.67773	Pass	0.09001	Pass
11	Serial test	0.0	Fail	$2.11547e-23$	Fail	0.21153	Pass
12	Approximate Entropy Test	0.0	Fail	$2.74079e-63$	Fail	0.50091	Pass
13	Cumulative Sums (Forward) Test	0.39379	Pass	0.41493	Pass	0.09313	Pass
14	Cumulative Sums (Reverse) Test	0.34748	Pass	0.38013	Pass	0.22585	Pass
15	Random Excursions Test	0.55150	Pass	0.07219	Pass	0.12375	Pass

Table 3.2: Continued

	NLFSR-PRNG		**LM-PRNG**		**LM-NLFSR-PRNG**		**LSS-NLFSR-PRNG**	
n^0	P_{value}	Conclusion	P_{value}	Conclusion	P_{value}	Conclusion	P_{value}	Conclusion
1	$2.14427e-10$	Fail	$7.17529e-05$	Fail	0.76415	Pass	0.589159	Pass
2	0.99999	Pass	$3.47372e-10$	Fail	0.37932	Pass	0.921522	Pass
3	0.0	Fail	$4.00730e-121$	Fail	0.55484	Pass	0.476752	Pass
4	$3.55113e-08$	Fail	$1.32771e-91$	Fail	0.12164	Pass	0.671805	Pass
5	0.405550	Pass	0.69244	Pass	0.69244	Pass	0.419681	Pass
6	$2.67766e-47$	Fail	0.22926	Pass	0.77602	Pass	0.741104	Pass
7	$3.81760e-14$	Fail	$8.69254e-41$	Fail	0.94259	Pass	0.165002	Pass
8	$8.05656e-08$	Fail	$2.79215e-35$	Fail	0.17493	Pass	0.232730	Pass
9	-1.0	Fail	-1.0	Fail	-1.0	Fail	-1.0	Fail
10	0.64701	Pass	0.59279	Pass	0.86082	Pass	0.415656	Pass
11	0.08374	Pass	$9.02224e-09$	Fail	0.08374	Pass	0.756912	Pass
12	0.0	Fail	$3.67830e-63$	Pass	0.14597	Pass	0.049818	Pass
13	$3.52703e-10$	Fail	0.00012	Fail	0.82356	Pass	0.499073	Pass
14	$3.30387e-10$	Fail	$6.49393e-05$	Fail	0.88208	Pass	0.931289	Pass
15	0.71348	Pass	0.18615	Pass	0.46045	Pass	0.761365	Pass

3.6.2 Key sensitivity evaluation

Correlation tests based on Pearson's correlation coefficient and the Hamming distance defined respectively by Equation 3.6 and 3.7 were used to examine the consistency between two generated sequences $str = [x_1...x_i]$ and $str1 = [y_1...y_j]$, and to measure the strength of linear association between sequences in order to guarantee the sensitivity of the keys.

$$R(str, str1) = \frac{\sum_{i=0}^{n-1}(x_i - \bar{x}) - (y_i - \bar{y})}{[\sum_{i=0}^{n-1}(x_i - \bar{x})^2]^{1/2}.[\sum_{i=0}^{n-1}(y_i - \bar{y})^2]^{1/2}} \tag{3.6}$$

$$d(str, str1) = \sum_{j=0}^{m-1}(x_j \oplus y_j) \tag{3.7}$$

$\bar{x}$ and $\bar{y}$ represent the mean values of str and $str1$ respectively.

In the Hamming distance test, three major binary key-stream sequences, K1, K2, and K3, were generated using slightly different starting values. Given the example of LM-LFSR-PRNG, (the same test is applied for other generators), where:

K_1 was obtained using the following initial values:
$r_0 = 3.999000000000011$, $X_0 = 0.100000000000011$, $LFSR = 10101101$.

K_2 was obtained using the following initial values:
$r_0 = 3.999000000000099$, $X_0 = 0.100000000000099$, $LFSR = 10101100$.

K_3 was obtained using the following initial values:
$r_0 = 3.999000000000059$, $X_0 = 0.100000000000067$, $LFSR = 10101001$.

The correlation results explained by Pearson correlation coefficients with the Hamming distance test of the three key sequences were constructed as described in Table 3.3.

Table 3.3: Results of Pearson correlation and the Hamming distance

Correlation test	Generator	K1 vs K2	K1 vs K3	K2 vs K3
Pearson correlation coefficients	$LSS - LFSR - PRNG$	-0,0030	0,00306	0,00023
	$LM - NLFSR_2 - PRNG$	-0,0038	0,0043	0,00043
	$LSS - NLFSR_3 - PRNG$	0.00209	0,00106	0,00170
Hamming distance	$LSS - LFSR - PRNG$	0.9953	0.99571	0.9953
	$LM - NLFSR_2 - PRNG$	0.9933	0.99581	0.9959
	$LSS - NLFSR_3 - PRNG$	0.99615	0.99594	0.99606

In our case, the relationship between $K1$, $K2$, and $K3$ is assigned a value between -1 and 1, which might imply that the three sets of keys are uncorrelated, and can be interpreted as

the three sequences of keys having different information contents. The suggested key-stream generators appears to be highly sensitive to the system parameters and initial values (keys), and they have the capacity to create uncorrelated and large number of key sequences. This capability may be helpful for many cryptographic applications.

3.7 Design of symmetric encryption algorithm based on the proposed key-stream generators

In this section, we describe the application of the developed generators for image encryption. General architecture of the proposed system based on the FSR and chaos is illustrated by Figure 3.5.

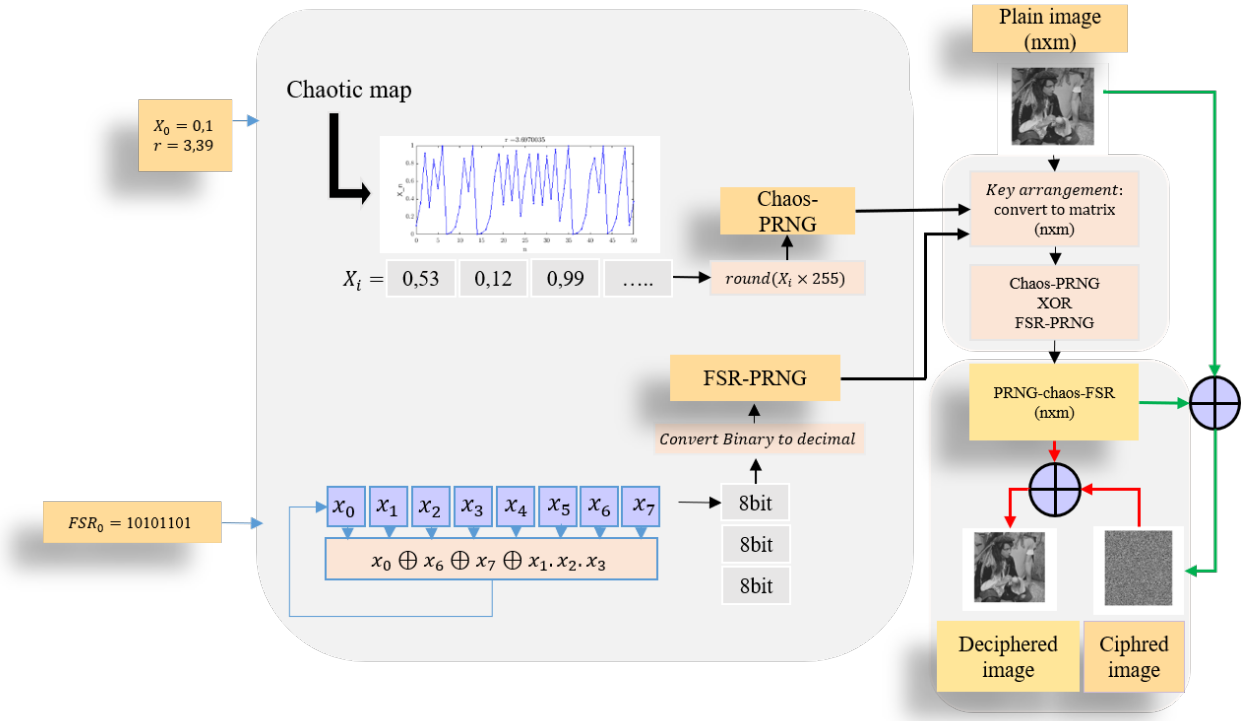


Figure 3.5: Proposed cryptosystem architecture

The procedure of encryption/decryption comprises two stages. In first, a key arrangement is performed to adapt the produced key-stream to a size similar to the source image, where we reshape the generated pseudo sequence vector to an $M \times N$ matrix named K by flowing the row-major ordering. The encryption and decryption mechanisms are outlined in the following steps: For encryption, the input image P is joined with the designed key-stream

matrix K of equal length through XOR operation to get the encrypted image $E = K \oplus P$, that may be safely transmitted in a secure communications scheme.

The decryption process is performed using the reverse technique of the encryption phases, in which we perform $E' = E \oplus K$ on the encrypted image E to obtain the decrypted image E'.

3.8 Experimental tests and security analysis

An effective image encryption technique can withstand numerous regularly used attacks, such as the brute-force attack and differential attack. This section includes thorough numbers, tables, and explanations to assess the performance and compare the implementation of the proposed cryptosystem. We used MATLAB R2018a to run a simulation experiment on a PC with a Pentium (R) Dual Core T4500 CPU running at 2.30 GHz and 2 GB of RAM under Microsoft Windows 7.

3.8.1 Statistical analysis

First, histogram analysis, the correlation coefficient tests in three directions, along with the entropy analysis, are the three metrics used to assess the efficacy of the proposed image encryption algorithm against statistical assaults.

1. **Histogram analysis**
   Figure 3.6 shows the histograms plot of the 256×256 standard Lena image, encrypted image, and decrypted image. The flat histogram distributions of the pixel values in an encrypted images is achieved through the three schemes based on the $LSS - LFSR - PRNG$, $LM - NLFSR_2 - PRNG$, and $LSS - NLFSR_3 - PRNG$. These results refers to a uniform distribution of pixel intensities in the encrypted image. This means that all intensity values occur with equal frequency, resulting in a histogram that appears flat. Hence, we ensure that the encrypted data is indistinguishable from random noise, providing a higher level of security.

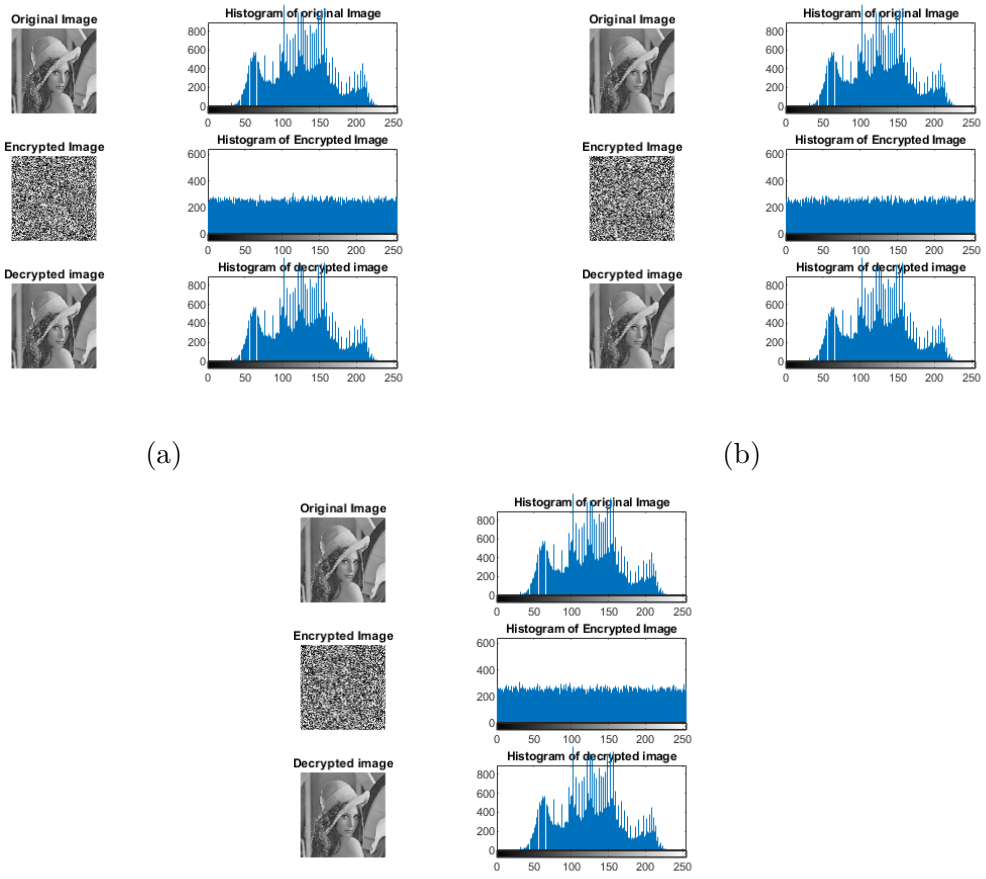


Figure 3.6: Histograms of plain, ciphered, and decrypted Lena images based on:(a)LSS-LFSR-PRNG; (b) LM-NLFSR-PRNG; (c) LSS-NLFSR-PRNG

2. **Correlation analysis**

   The statistical properties of the encryption results in correlation between neighboring pixels of Lena image, have been tested listed, and shown in Figure3.7 and Table 3.4. Our cryptosystem based on the three designed key-streams achieve near-zero correlation, indicating a high level of randomness and security in the encrypted images with resistance to statistical attacks.

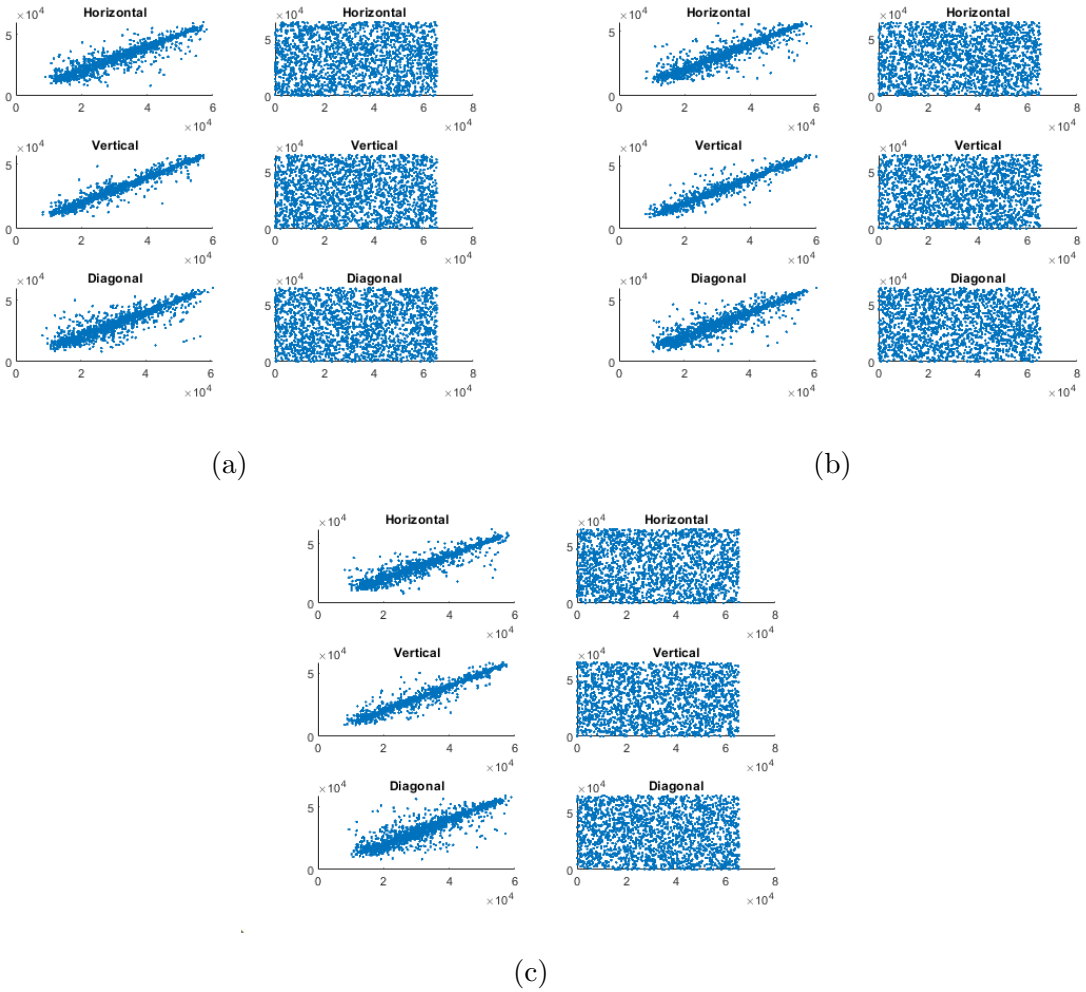


Figure 3.7: Correlation plot of plain (right) and ciphered images (left) based on:(a)LSS-LFSR-PRNG; (b) LM-NLFSR-PRNG; (c) LSS-NLFSR-PRNG

Table 3.4: Horizontal, vertical, and diagonal corr values of 2048 pairs of original, encrypted and decrypted images based on LSS-LFSR-PRNG; LM-NLFSR-PRNG; and LSS-NLFSR-PRNG

		Original	Encrypted	Decrypted
LSS-LFSR-PRNG	H	0.96663	- 0.02191	0.95839
	V	0.97026	0.01824	0.97074
	D	0.93935	0.00922	0.93859
LM-NLFSR-PRNG	H	0.9438	0.0019	0.9346
	V	0.9700	-0.0133	0.9675
	D	0.9152	-0.0205	0.9125
LSS-NLFSR-PRNG	H	0.945365	0.014755	0.9320
	V	0.945365	0.014755	0.9320
	D	0.908599	0.015830	0.911418

3. **Entropy analysis**

   The entropy of encrypted images varies depending on the key-stream used. The information entropy values are listed in Table 3.5 and remain tightly close to 8. Consequently, it is exceedingly difficult to extract visual information from ciphered images in the three cases. In particular, the cryptosystem based on the proposed LSS-NLFSR-PRNG attains maximum information entropy, this means this scheme has a random pixel value distribution, making it nearly impossible to obtain visual information from encrypted images, which confirm its security against entropy attacks.

Table 3.5: Shannon entropy results

	Original	Encrypted	Decrypted
LSS-LFSR-PRNG	7.24322	7.99744	7.24322
LM-NLFSR-PRNG	7.24322	7.9970	7.24322
LSS-NLFSR-PRNG	7.24322	7.997453	7.24322

3.8.2 Differential analysis

We computed the NPCR, UACI, and PSNR to evaluate the strength of image encryption algorithm against differential attacks. Results are resumed in Table 3.6.

The NPCR is shown to have a maximum predicted value of 99.6% when two random images are considered, while the UACI's maximum anticipated value is approximately 33%.

The proposed algorithm based on the LSS-NLFSR-PRNG achieved best result of NPCR and UACI, which are close to the aforementioned predicted values. This result guarantees necessary resistance to any type of differential cryptanalysis.

Table 3.6: NPCR, UACI, and PNSR values

	NPCR	UACI	PSNR
LSS-LFSR-PRNG	0.91585	0.27941	45.02431
LM-NLFSR-PRNG	0.996231	0.278690	45.10937
LSS-NLFSR-PRNG	0.996566	0.278823	45.024871

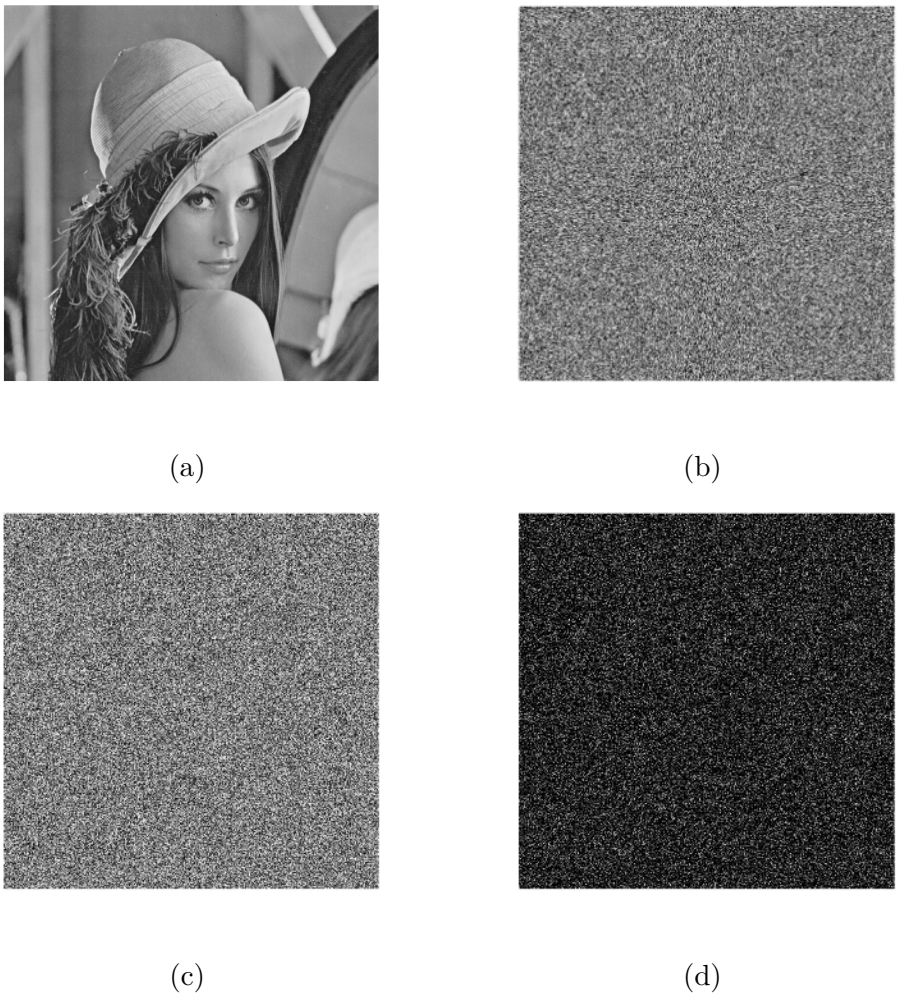
3.8.3 Key sensitivity analysis

To verify the scheme against the sensitivity of the key, a key is taken and an image is encrypted. The value of the key, or more specifically, one bit from the bit-stream used for encryption is changed, and an image is encrypted again. The same analysis was performed during the decryption process. The output images were compared to determine if there were any matches. To perform this analysis on the three schemes by the same manner, the first key set (referred as K_1) is:

$r_0 = 3.999$, $X_0 = 0.1$, $FSR_0 = 10101101$

K_1 was used to encrypt the original image in Figure 3.9b. The encrypted image is E_1 in Figure 3.8b.

To create a different key set designated as K_2, we apply a small change to $r_0 = 3.999000000000001$ while maintaining X_0 unchanged. The same manner with K_2 to obtain (E_2) in Figure 3.8c.

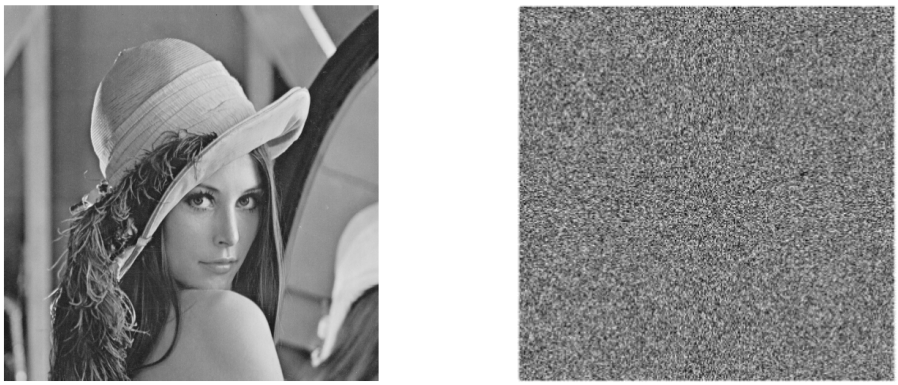


(a) (b)

(c) (d)

Figure 3.8: Key sensitivity analysis (example of the scheme based on the LSS-NLFSR-PRNG): (a) Original image; (b) Encrypted image E1 using K1; (c) Encrypted image E2 using K2; (d)$|E1 - E2|$

The simulation results and the pixel-to-pixel difference $|E_1 - E_2|$ (Figure 3.8d) clarify that the encrypted image changes significantly as a consequence of tiny changes in the original image and key.



(a) (b)

Figure 3.9: Key sensitivity analysis (example of the scheme based on the LSS-NLFSR-PRNG):(a) Decrypted with K1; (b) Decrypted with a wrong key K2

Figure 3.9 show the decryption results, where only the correct key ($K1$) allows for full reconstruction of the plain image. However, even a slight modification in the security key ($K2$) leads to image decryption failure.

3.8.4 Key space

To evaluate the cipher's security, a brute-force assault, known as a chosen-plaintext attack, is used. Since the approach uses double precision floating point numbers in representing real numbers on computers, the cost to estimate r is 10^{-15}, in accordance with IEEE-754 standards. The total number of keys used as a component of the key generator system is represented by the key space employed in the proposed work. For our three schemes, we have:

The 8-bit LFSR/NLFSR generated sequences are $key1 = (2^8 - 1) \approx 255$. Thus, total key-space of the LFSR/NLFSR is 2^8.

The precision of the initial value X_0 variation along with bifurcation parameter r of the used chaotic maps is 10^{-15}. The key-space for X_0 is $key2 \approx 10^{15}$ and for the bifurcation parameter r is $key3 \approx 10^{15}$.

The iterations number depend on the used image, if we take the example of the Lena image of 256×256, so the iteration number $n = 65536$ is key4 $\approx 65.10^3$.

If the length of each parameter and the initial value are set to 15 decimals, the key space of the proposed algorithm is 2^{126} for LM-NLFSR and 2^{224} for the LSS-LFSR and LSS-NLFSR

proposed systems.

To fight brute force attacks, the key space of the image encryption technique should be greater than 2^{100}. Consequently, the integration of the coupled LSS map greatly improves the key space available for the proposed cryptosystem to fend against brute-force attacks compared to conventional LM.

3.9 Conclusion

In this chapter, we presented a method for constructing a high-quality key-stream based on the feedback shift registers of two types (linear and nonlinear) and chaotic maps (simple and coupled) to boost the security level compared to existing simple generators. The LSS and NLFSR are the core aspects of our proposed generator; therefore, they have many more benefits than simply using the logistic map or the basic LFSR. By construction, the sequence of bits generated by the LSS-NLFSR-PRNG is aperiodic because the sequence of numbers generated by the LSS is aperiodic. Additionally, the use of Galois NLFSRs in encryption algorithms allows high-quality pseudo random number generation with good statistical properties, high throughput, high speed, ease of implementation, and increased immunity to statistical attacks. Thus, our cryptosystem provides a balance between security and computational complexity, making them more efficient for real-time applications. In the next chapter, we will discuss the impact of diffusion-confusion complexity on improving cryptosystem security.

Chapter 4

Efficient confusion-diffusion structures for image encryption

4.1 Introduction

The complexity of encryption algorithms has a significant impact on their performance and effectiveness in ensuring data security especially when we manipulate images. The computational speed of algorithms is essential for high-speed computing devices, and the efficiency of algorithms directly influences the overall performance of cryptosystems. Hence, the choice of the best confusion-diffusion architecture may affects factors such as, time complexity, and space complexity during encryption and decryption processes. In this chapter, we compare the performance of three chaos based image encryption algorithms of different confusion-diffusion architecture, in term of computational speeds, memory usage, and resistance to several attacks.

4.2 Proposed approaches

The notable nonlinear features of chaos make it an appropriate candidate for several cryptographic applications. However, the dimension of the chaotic map affects the encryption reliability and resistance to attacks. Therefore, the choice of the appropriate chaotic maps with the appropriate dimension in the confusion and diffusion steps is crucial in achieving excellent encryption quality and ensuring the robustness of the encryption system.

In this section, three scenarios are proposed to study the impact of the confusion-diffusion complexity in improving security and robustness of a cryptosystem.

In the first scenario, we design our image cryptosystem based on the sequence produced by the simple 1D logistic map and XOR operation. In second, we upgrade the complexity of the confusion-diffusion architecture, where, we use the 1D logistic map for confusion and the Chebyshev system in the diffusion. At last, the 3D intertwining logistic map which exhibit more robust hyperchaotic behavior across a broader range of chaos is investigated. The details of the three algorithms are discussed in the provided subsections.

4.2.1 Algorithm 1: 1D LM-based cryptosystem

In this algorithm, we expand on the two encryption levels used in this study. Our construction method consists of two steps.

- Design of a key generator based on LM.

- Design of the image encryption algorithm based on the sequence produced by the above generator and XOR operation.

An encryption key is required to encrypt the image. To achieve this, we utilize the LM, which may construct a pseudo-random number generator using Equations 2.4. The generated sequence is converted to an 8-bit integer through multiplication by 255 the rounding to the nearest decimal value.

Using the constructed remote sequence of keys after conversion to binary, we apply a bitwise XOR operation between each pixel of the plain image and the corresponding bit of the proposed key matrix of equal size.

During the decryption step, the encryption process is reversed by selecting the same secret key used to encrypt the plain image and performing an XOR operation between every element of the encrypted image matrix and every element of the secret key matrix.

4.2.2 Algorithm 2: 1D LM-Chebyshev based cryptosystem

This algorithm presents an efficient cryptosystem for image security that exploits the advantages of chaotic maps. In the proposed approach, the logistic map and Chebyshev system are employed. The optimal sequence generated by the LM is used to scramble the image pixels, whereas the Chebyshev system is employed to generate a secret key for diffusion. P represents the plain image, E represents the ciphered image, X represents the chaotic variables, C represents the permuted vector, K represents the diffusion key, and G represents the secret matrix. Steps of confusion and diffusion are as follows:

1. Load the image P of size $M \times N$ to be encrypted and convert it into a vector of pixel

values by applying a row major alignment.

$$P = \{P(1), P(2), ..., P(M \times N)\} \tag{4.1}$$

2. Iterating the LM for $M \times N$ times, and then obtaining a vector of sequences X_{LM} with $M \times N$ chaotic variables.

$$X_{LM} = \{X_{LM}(1), X_{LM}(2), ...X_{LM}(M \times N\} \tag{4.2}$$

3. Generating the permutation vector V by sorting X in ascending order.

$$V = \{V(1), V(2)...V(M \times N)\} \tag{4.3}$$

4. Apply permutation operations on the image pixels P with the permutation vector V.

$$C(i) = P(V(i)) \tag{4.4}$$

5. Iterating the Chebyshev system for $M \times N$ times, and then obtaining a second sequence X_{CH} with $M \times N$ chaotic variables.

$$X_{CH} = \{X_{CH}(1), X_C H(2), , X_{CH}(M \times N)\} \tag{4.5}$$

6. Rounding to the nearest decimal value then quantizing X_{CH}.

$$K_1(i) = round(X(i) \times 255) \tag{4.6}$$

7. Converts $K_1(i)$ vector to a binary matrix $K_b(i, j)$.

8. Circshift the elements of the binary matrix in a circular manner along one dimension.

$$R(i, j) = Circlhift(K_b(i, j), 1) \tag{4.7}$$

9. Converts $R(i, j)$ to a decimal vector $K_2(i)$.

10. Perform $K_1(i) \oplus K_2(i)$ to get secret matrix G sized $M \times N$. Where $\oplus$ represents the XOR operation of the corresponding elements.

11. Transposition of the permuted vector $C(i)$ is performed by swapping the X and Y indices of its array representation.

12. Encryption process is performed by calculating the vector E

$$E = C'(i) \oplus G \tag{4.8}$$

13. Reshape the vector E to a matrix of $M \times N$ size.

14. Decryption process is a reverse process of our encryption scheme.

4.2.3 Algorithm 3: 3D intertwining LM-based cryptosystem

The principles of the 3D ILM-cosine, which were used to produce keys for the proposed image encryption scheme, along with the confusion-diffusion steps are annotated in this section.

Intertwining logistic map (ILM)

In 2014, Wang and Xu [135] proposed an intertwining relation between different LM sequences[136], which indicates that the ILM has more dynamic behavior than the LM[136]. The equations for the ILM sequence are as follows:

$$\begin{cases} x_{i+1} = (\eta\sigma y_i(1-x_i) + z_i) \mod 1 \\ y_{i+1} = (\eta\vartheta y_i + z_i(1+x_{i+1}^2)) \mod 1 \\ z_{i+1} = (\eta(y_i+1+x_i+1+\kappa)\sin z_i) \mod 1 \end{cases} \tag{4.9}$$

This system of equations exhibits chaotic behavior for η in the range of $[0, 4)$, $\eta > 33.5$, $\vartheta > 37.9$, and $\kappa > 35.7$.

Intertwining logistic map-cosine (ILM-Cosine)

The ILM-cosine expressed by Equation 4.10 is the result of combining the ILM with a cosine function with the aim of improving the ILM output nonlinearity. This system of equations exhibits chaotic behavior when η is in the range of $[0, 4)$, $\sigma > 33.5$, $\vartheta > 37.9$, and $\kappa > 35.7$.

$$\begin{cases} x_{i+1} = \cos\left((\eta\sigma y_i(1-x_i) + z_i) \mod 1+\vartheta\right) \\ y_{i+1} = \cos\left((\eta\vartheta y_i + z_i(1+x_{i+1}^2)) \mod 1+\vartheta\right) \\ z_{i+1} = \cos\left((\eta(y_i+1+x_i+1+\kappa)\sin z_i) \mod 1+\vartheta\right) \end{cases} \tag{4.10}$$

The proposed image encryption scheme based on the ILM-cosine is shown in Figure 4.1.

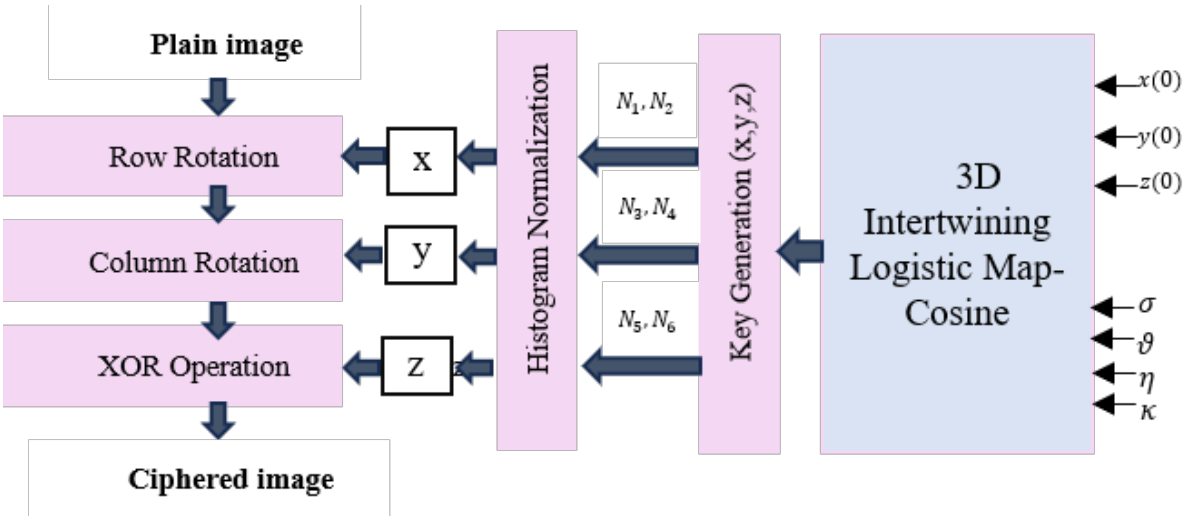


Figure 4.1: Encryption steps based on 3D Intertwining LM-Cosine

To create a cryptosystem strong enough to encrypt images, the following five crucial procedures are needed:

1. **Key generation**

   In this step, we use Equation 4.10 to generate a pseudorandom bit sequence based on the 3D ILM-cosine chaos sequences. The initial conditions and parameter values are considered keys to the cryptosystem.

   $x(1) = 0.2350$, $y(1) = 0.3500$, $z(1) = 0.7350$, $\eta = 3.7700$,$\sigma = 33.6$, $\vartheta = 39.69$, $\kappa = 36.58$.

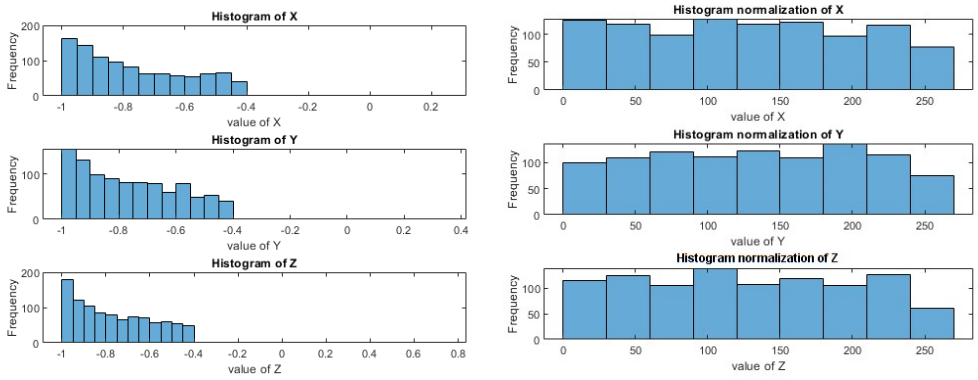
2. **Histogram normalization**

   The generated values and histogram generation of the 3D ILM-cosine chaotic sequence x, y, and z obtained using Equation 4.10 are depicted in Figure 4.2a. The resulting chaotic sequence histogram has a nonuniform distribution, which may affect the security of the system. Consequently, we use a normalizing (equalization) technique for x, y, and z using Equation 4.11 to further strengthen the security of the resulting histograms by a sufficiently large number because the map only generates floating-point values between 1 and -1.

$$\begin{cases} x = int(x \times N_1) \mod N \\ y = int(y \times N_3) \mod M \\ z = int(z \times N_5) \mod 256 \end{cases} \tag{4.11}$$

   where N_1, N_3 and N_5 are large random numbers that are chosen to be equal to or greater than 100,000 for simplicity, while M and N are chosen to be equal to the image

dimension (256×256). It is clear from Figure 4.2b that after applying the above constraints, we obtain an equalized histogram for x, y, and z.



(a) (b)

Figure 4.2: Histograms of generated sequences x,y, and z:(a) original histograms;(b) normalized histograms

3. **Row rotation**

   The steps used to rotate a gray image of $M \times N$ dimensions are as follows:

   - Applying an offset value N_2,
   - Choosing M elements of the chaos sequence x starting from the offset value N_2,
   - The chaos value x, obtained using Equation 4.11 is used to rotate the row.

4. **Column rotation**

   The steps used to rotate the column are similar to those of row rotation and can be applied as follows:

   - Applying an offset value N_4,
   - Choose N elements of the chaos sequence y starting from the offset value N_4.

5. **XOR operation**

   The sequence acquired from the row and column rotations is finally subjected to an XOR operation to produce new pixel values that are distinct from the original values. The XOR operation is performed using the following steps.

   - Converting the $M \times N$ image to a new $1 \times MN$ image,
   - XOR the chaos sequence z starting from N_6.

4.3 Simulation results and analysis

(a) (b) (c)

Figure 4.3: Images used for the analysis:(a) Heart MRI Scan Image (M1); (b) Abdomen sonography image (M2); (c) Chest X-ray Image (M3)

In this section, various tests often used to analyze the statistical metrics and security of cryptosystems are employed to evaluate the performance of the proposed scheme. The tests for the performance analysis of the proposed scheme were conducted on a PC of Core (TM) i3-4030U CPU @ 1.90 GHz with 4 GB of RAM.

In analyzing the proposed solution, all of the standard test images were obtained from The Intramural Research Program of the NClinical Center and the National Library of Medicine. The collection in Figure 4.3 comprises X-ray images (chest X-ray images), sonography images (abdominal sonography images), and MRI images (heart MRI images). These images are .png images with 256×256-pixel resolution.

In this section, numerous tests were performed on the tested images to assess and compare the security and statistical capabilities of the three proposed chaos based image-encryption algorithms.

The histograms of the plain images and encrypted images generated by the three schemes shown respectively in Figure 4.4a, 4.4b, and 4.4c share no similarity. The encrypted images by algorithm 2 and 3 has a more uniform histogram and and evenly distributed. However, the histogram of the encrypted image by algorithm 1 which use a simple cryptosystem and 1D LM is slightly flat. This signifies that the choice of complex confusion-diffusion architecture based on chaos has a significant impact in obtaining uniformly distributed pixel values, and the use of high dimensional chaotic maps makes it difficult for attackers to predict the relationship between neighboring pixels, thereby increasing the security of the encryption algorithm.

Second, the entropy of images with 8-bit pixel values should be close to 8. With a mean

entropy value of 7.99, the 3rd algorithm based on the 3D ILM is more entropy-rich than the aforementioned methods. That makes it more resistant to ciphertext-only attacks.

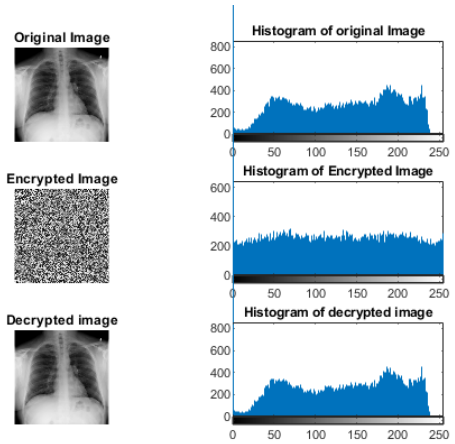
When subjected to differential analysis, the scheme based on the 3D ILM yielded good results, with a mean score comparable to those theory. Thus, high dimensional chaotic maps is more advantageous in enhancing resistance to differential attacks by introducing randomness and complexity into the encryption process.

When analyzing computational processing results, it is clear that the execution time of the encryption algorithm and the memory usage are influenced by the complexity of the chaotic map used. The schemes based on 1D chaotic map uses the least amount of memory and encrypts a 256×256 image in less than 0.2 seconds.

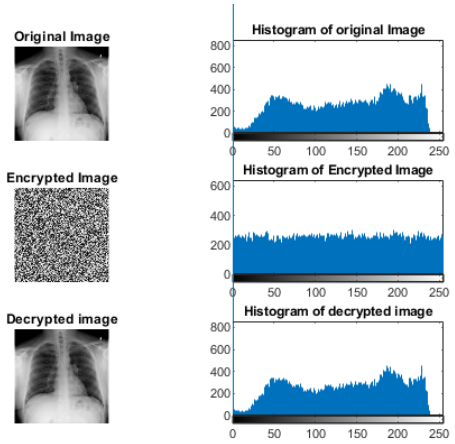
3D ILM chaotic map achieving the highest execution time observed, this can be explained by the fact that the 3D chaotic maps are extremely sensitive to the initial values of their parameters, and even a minor change in these parameters can result in a big change in the output, this sensitivity can potentially slow down the encryption process as more calculations and iterations may be required. This demonstrate that the choice of chaotic map can significantly impact the execution time of the encryption algorithm.

As results, the dimension of the chaotic map used in a cryptosystem has an impact on its performance and security. Higher dimensional chaotic systems, such as the used 3D ILM, exhibit better dynamical behavior and superior performance in terms of ergodicity, complexity, and randomness. They are capable of generating unpredictable keystreams, which are highly suitable for encryption and may enhances security and efficiency of the cryptosystem. However, while these systems are able to enhance the abilities of resisting exhaustive and statistical attacks, they fail in maintaining encryption speed in comparison to simple 1D maps.

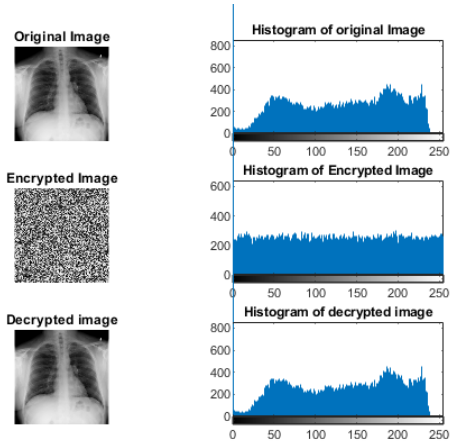
1D chaotic map with a wider and continuous chaotic range, larger Lyapunov exponent, and more complex behavior such as the used Chebyshev system also improves the security of an image cryptosystem.



(a)



(b)



(c)

Figure 4.4: Histogram results of plain, encrypted, and decrypted images based on: (a)Algorithm 1;(a)Algorithm 2;(a)Algorithm 3

Table 4.1: Comparison of the three algorithms

Metrics	Ideal value	Image used	Algorithm 1	Algorithm 2	Algorithm 3
Entropy	Equal to 8	Heart MRI Scan image	7.9730	7.98784	7.9974
		Abdomen Sonography image	7.9730	7.98087	7.9968
NPCR	Equal to 99.609375%	Heart MRI Scan image	99.020385742	99.588012	99.6047
		Abdomen Sonography image	99.020385	99.4537353	99.6551
UACI	Equal to 33.463541%	Heart MRI Scan image	33.249912	33.748551	33.9776
		Abdomen Sonography image	35.90232101	36.672303	36.9822
PSNR	High as much as it can be	Heart MRI Scan image	40.1368	40.652935	40.6106
		Abdomen Sonography image	37.8826	38.89886	38.6624
Run-Time (Mean)	Small as much as it can be	-	0.074915	0.245006	0.5599
Memory usage (Mean)	Low as much as it can be	-	0.155648	0.352256	0.88064
Contrast analysis	High	Heart MRI Scan image	11131.75	11109.54	10917.88
		Abdomen Sonography image	11787.86	10924.73	10953.55
Key space	Large as much as it can be	-	10^{34}	10^{64}	10^{135}

4.4 Conclusion

Chaos based image encryption plays a critical role in protecting sensitive information in different domains. In this chapter, we discussed the practical implications of image encryption based on different chaotic maps. The dimension of the used chaotic map affects the encryption reliability and resistance to attacks, and the choice of the appropriate chaotic map should be carefully considered to ensure efficient and timely encryption. In addition, the complexity of the confusion-diffusion architecture has a direct impact on its security. A more complex algorithm generally provides a higher level of security, as it makes it more difficult for attackers to decipher the encrypted data. In the next chapter, a new chaotification framework for enhancing chaotic behavior of 1D chaotic map is proposed.

Chapter 5

Design of a new tangent chaotification model for improving chaotic maps behavior with application to image encryption

5.1 Introduction

The most frequently used image encryption based on chaotic systems paves the way toward the efficient transfer of data through the cloud network. However, it presents several shortcomings, such as the uneven distribution of conventional chaotic maps, narrow and insufficient key space, and a small amount of key change in a single one-dimensional chaotic system. In this chapter, we propose a new nonlinear chaotification system capable of producing new 1-D discrete-chaotic maps that exhibit better dynamical behavior, Lyapunov exponent, bifurcation, and larger chaotic interval compared with their seed maps, by applying tangent nonlinear transforms to the outputs of the existing chaotic maps. To illustrate this, we analyzed the chaotic complexity of three proposed chaotic maps: the enhanced tangent logistic map(T-LM), enhanced tangent sine map(T-SM), and enhanced tangent Chebyshev system(T-CH). Second, we propose a new encryption algorithm in which the optimal sequence generated by the T-LM system is used to scramble the image pixels, whereas T-CH is employed to generate a secret key for diffusion.

5.2 Basic chaotification schemes: Related works

Several studies have been conducted to reduce the downsides of 1D chaotic maps. In this section, we review basic chaotification methods that are developed in the literature to enhance 1D chaotic map behavior.

5.2.1 Modular chaotification framework

Hua et al.[137] developed a simple yet effective modular chaotification framework that use modular operations to increase the dynamic complexity of a 1D chaotic map. In particular, the control value of the 1D chaotic map is initially increased before utilizing the modular operation to transform the output states into a set range.

$$M(x_n) = x_{n+1} = G(x_n, p) \mod \theta \tag{5.1}$$

This process is illustrated in Figure 5.1 and described by Equation 5.1, where $G(x_n, p)$ denotes a 1D chaotic map, p is a parameter of $G(x_n, p)$, and θ is the modulus coefficient. It is clear that x_n is within the range $[0, \theta)$.

Figure 5.1: Modular chaotification structure

Additionally, a two-dimensional modular chaotification system represented by Equation 5.2 is suggested in [138] to address the shortcomings of the current 2D chaotic maps, in areas of discontinuous, constrained chaotic ranges, and incomplete output distributions. The $2D$ generated sequence can significantly increase the chaotic complexity of a 2D chaotic map by using a modular operation as a bounded transformation and applying it to the map's outputs.

$$M(x, y) = F(x, y) \mod N \tag{5.2}$$

x and y are two variables, the modulus coefficient N is a positive integer, and $F(x, y)$ is a $2D$ chaotic map.

5.2.2 Fusion chaotification framework

By combining the dynamics of two seed maps in a nonlinear manner, Zhou et al.[12] suggested a fusion operation that could create new chaotic maps. Its definition is given by Equation

5.3, and its structure is illustrated in Figure 5.2. The input was simultaneously fed into two seed maps throughout each iteration and the outputs of the two seed maps were subsequently merged via modular arithmetic.

$$x_{i+1} = f(x) \oplus g(x) = (g(x_i) + f(x_i)) \mod 1 \tag{5.3}$$

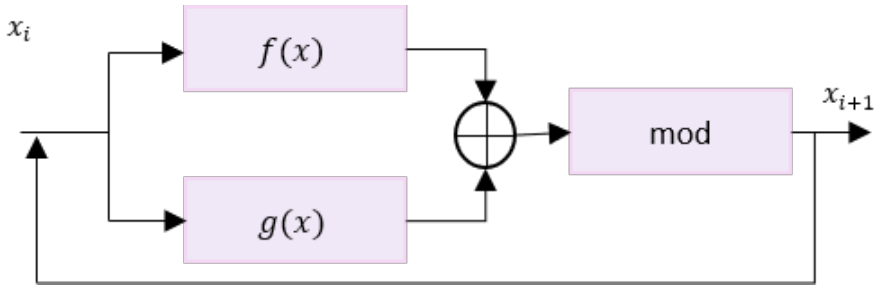


Figure 5.2: Fusion chaotification model structure

5.2.3 Cascade chaotification framework

Motivated by cascade structures in electronic circuits, this technique was tested and subsequently applied before by Hua et al. [139] under the title of series-wound framework on logistic, sine and Gaussian maps, then generalized as a cascade structure framework by Zhou et al.[12].

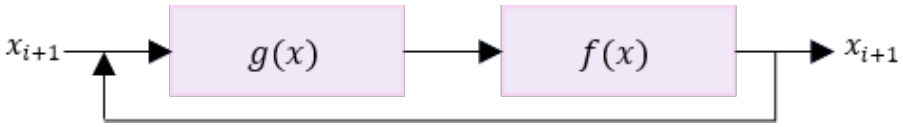


Figure 5.3: Cascade chaotification model structure

It can create new $1-D$ chaotic maps by combining two already-existing $1-D$ chaotic maps. Figure 5.3 shows the cascade structure, where $G(x)$ and $F(x)$ are two seed maps that connect the two seed maps in series. The output of $G(x)$ is fed into $F(x)$'s input, and $F(x)$'s output is then fed back into $G(x)$'s input for recursive iterations, as defined mathematically by Equation 5.4, where $G(x)$ and $F(x)$ are two seed maps.

$$x_{n+1} = F(G(x_n)) \tag{5.4}$$

By repeatedly cascading a chaotic map with itself, the cascade operation may also be applied to many chaotic maps; this framework is known as the scalar cascade [140] proposed

later by Zhou et al.[141]. Its structure is shown in Figure 5.4. The scalar cascade operation exhibits all cascade operation characteristics.

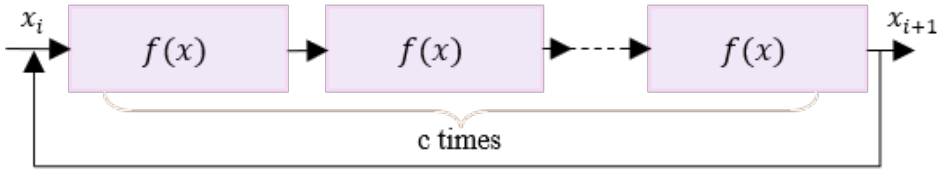


Figure 5.4: Scalar cascade chaotification model structure

5.2.4 Cosinus chaotification framework

The cosine function is a trigonometric real value function. It has been extensively utilized in several fields, including geometry, geodesy, and navigation, and has been used to investigate periodic events because it is a periodic function [142]. Researchers [143],[142] have leveraged the properties of the cosine function to enhance the chaotic behavior of different chaotic maps. Thus, cosine chaotification approaches have been proposed to improve the chaotic behavior and complexity of current discrete chaotic systems via several models.

Model 1

The cosine function is applied as a nonlinear transform to the outputs of three different examples in this model, this approach is simple but effective compared with other techniques that use two or more existing chaotic maps as seed maps and have complex nonlinear frameworks [144]. The technique proposed by Natiq et al.[144] is defined by Equation 5.5:

$$\begin{cases} x_1(n+1) = b\cos\left(F_1(x_1(n), x_2(n).., x_m(n))\right), \\ x_2(n+1) = b\cos\left(F_2(x_1(n), x_2(n).., x_m(n))\right), \\ x_3(n+1) = b\cos\left(F_3(x_1(n), x_2(n).., x_m(n))\right), \\ x_m(n+1) = b\cos\left(F_m(x_1(n), x_2(n).., x_m(n))\right) \end{cases} \tag{5.5}$$

Model 2

Hua et al.[143] proposed a cosine-transform-based chaotic system for generating chaotic maps with complex behaviors. In addition to providing a fundamental foundation, this scheme can combine any two existing chaotic maps, known as seed maps $f(a, x_i)$ and $g(b, x_i)$, with a shifting constant β and then perform a cosine transform to create new maps. a and b are their control parameters, and the variable β is a shifting constant. This framework is

defined by Equation 5.6. As indicated in Figure 5.5, the combination operation can effectively reduce the chaotic dynamics of two seed maps, and the cosine transform exhibits a complex nonlinearity.

$$x_{i+1} = (\pi(f(a, x_i) + g(b, x_i) + \beta)) \tag{5.6}$$

Model 3

The chaotification model proposed by Alawida et al.[145] uses an existing chaotic map rather than a linear function and introduces a new control parameter as part of an exponential function, as defined by Equation 5.7, to invoke large differences in seed values. This can be directly formulated as a cascade system to increase the chaotic complexity of a cosine-based system, where F is the seed map, and G is the cosine function. K was selected in the range of $[10, 24]$ for two reasons. First, when the angle of the cosine function is sufficiently large, even small differences in $F(x_n)$ will lead to widely diverging outputs.

$$X_{i+1} = \cos(2^{((k+F(x_n))}) \tag{5.7}$$

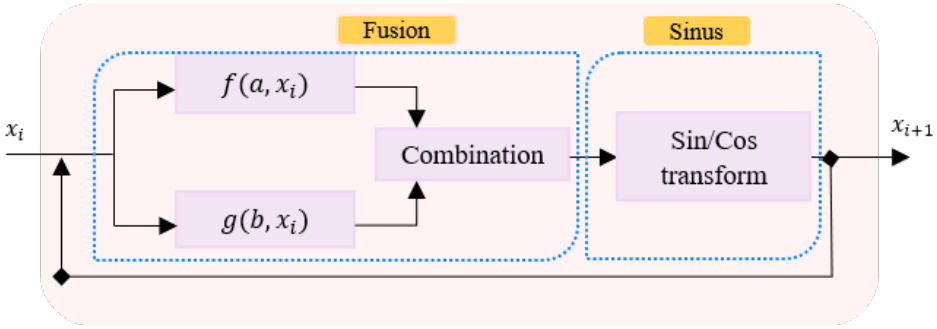


Figure 5.5: Cosine/Sine chaotification framework structure

5.2.5 Sinus chaotification framework

The sine function has restricted orbits and complex nonlinear features. In a wide parameter range, this approach can significantly improve the chaotic performance of existing $1D$ chaotic maps. The first model, as shown in Figure 5.6 is a simple framework designed by [146] to enhance the chaos complexity of existing $1D$ chaotic maps ($F(p, x_i)$ where p is a control parameter and x_i is the input) by using a sine function as a nonlinear transform and applying it to the outputs of $1D$ chaotic maps.

The sine chaotification structure described by Equation 5.8 can significantly increase the chaotic complexity of these maps.

$$x_{i+1} = \sin(\pi F(p, x_i)) \tag{5.8}$$

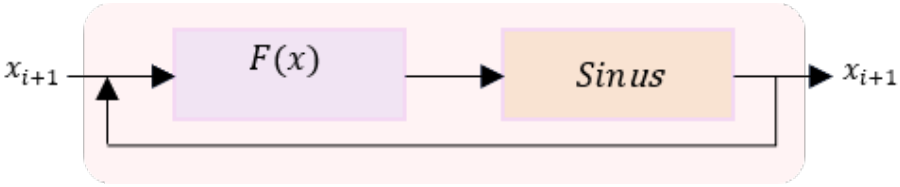


Figure 5.6: Sinus chaotification framework structure

Like in the case of the cosine chaotification framework, an extended version of the $2D$ sine chaotification framework was proposed in [147] to enhance the chaotic complexity of $2D$ chaotic systems, because of the ability of $2D$ chaotic systems to reconcile the trade-off between the implementation cost and chaotic performance.

$$x_{i+1} = N(x_i) = \sin(\pi(f(a, x_i) + g(b, x_i)) \tag{5.9}$$

The model described by Equation 5.9 uses the sine transform as a chaotification framework and applies it to each output of $2D$ chaotic systems, in addition to the combination operation, which is used to linearly combine the outputs of the two seed maps, whereas the sine transform performs a nonlinear transformation to the combination results, as shown in Figure 5.5. Because sine transform is a bounded function for any input, this method can generate chaos over a large parameter range. Here, the input x_i is fed concurrently into $f(a, x_i)$ and $g(b, x_i)$ in each iteration, and the outputs of $f(a, x_i)$ and $g(b, x_i)$ are subsequently subjected to the sine transform. The seed maps may be any 1D chaotic maps that are already in existence. The seed maps $f(a, x_i)$ and $g(b, x_i)$ may be configured by the user as identical or dissimilar chaotic maps.

5.3 Proposed chaotification system

1D chaotic systems such as the logistic map (LM), sine map (SM), and Chebyshev system (CH) are the most common low-dimensional chaotic map categories that are excellent for practical applications, because they have a straightforward chaotic structure, are simple to implement using hardware and software, and have good chaotic properties. However, they also have drawbacks such as constrained chaotic behavior ranges, uneven output distributions, and attack susceptibility.

Based on the related studies presented above, this section discusses the proposed chaotification method to address improving chaotic map properties with an evaluation of the chaos performance of the newly obtained chaotic maps.

We aim to exploit cascading, modular, sine and cosine chaotification scheme pros to design a new chaotification scheme based on a new tangent nonlinear transform, able to enhance the chaotic map properties and produce new systems. We choose the tangent function which is a real-valued trigonometric function due to the fact that it is a periodic function, it has been widely employed in many disciplines, including geometry, geodesy, and navigation, and is not investigated and not exploited in previous researches. As a result, three chaotic maps are designed; the enhanced tangent logistic map (T-LM), enhanced tangent sine map (T-SM), and enhanced tangent Chebyshev system (T-CH), which are capable of generating highly random sequences and might be used for cryptography applications. Second, based on the two maps exhibiting better performance, we propose a new encryption algorithm in which the T-LM system is used to scramble the image pixels, whereas T-CH is employed for diffusion.

In the first stage of our new chaotic system shown in Figure 5.7, the reciprocal function of the tangent (cotangent) function is applied as a nonlinear transform to the outputs of the chaotic seed map $f(r, x_n)$, which is then fed into the quantization function $G(k)$'s input, where the K range has been experimentally confirmed by bifurcation and $LE(k = 8$ in this study). The modular chaotification step is then applied to the output of $G(k)$, where " mod " denotes the modular operation, and is used to ensure that the generated chaotic sequence is contained within the $[0, 1]$ range. The seed map $f(r, x_n)$ may be any 1-D chaotic system with control parameter r that currently exists. The new chaotic system is described using Equation 5.10.

$$X_{n+1} = T(r, k, X_n) = \tan^{-1}(f(r, X_n))G(k) \mod 1 \tag{5.10}$$

Where $G(k) = 2^k, 8 \leq G(k) \leq 16$

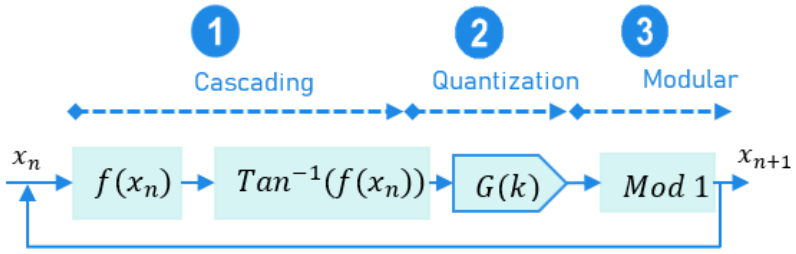


Figure 5.7: Proposed tangent chaotification system structure

Equation 5.10 is used to illustrate the enhanced tangent logistic map (T-LM), enhanced tangent sine map (T-SM), and enhanced tangent Chebyshev system (T-CH) presented in Table 5.1.

Table 5.1: Standard and proposed enhanced chaotic maps

Chaotic map	Name	Iteration function	Parameter
Standard Logistic map	LM	$X_{n+1} = rX_n(1 - X_n)$	$r \in [0, 4], X_n \in [0, 1]$
Enhanced Tangent logistic map	T-LM	$X_{n+1} = \tan^{-1}(rX_n(1 - X_n))2^8 \mod 1$	$r \in [0, 4], X_n \in [0, 1]$
Standard Sine map	SM	$X_{n+1} = r\sin(\pi X_n)$	$r \in [0, 4], X_n \in [0, 1]$
Enhanced Tangent Sine map	T-SM	$X_{n+1} = \tan^{-1}(r\sin(\pi X_n))2^8 \mod 1$	$r \in [0, 4], X_n \in [0, 1]$
Standard Chebyshev system	CH	$X_{n+1} = \cos(r\arccos X_n)$	$r > 0, X_n \in (-1, 1)$
Enhanced Tangent Chebyshev system	T-CH	$X_{n+1} = \tan^{-1}(cos(r\arccos X_n))2^8 \mod 1$	$r > 0, X_n \in (-1, 1)$

5.4 Behavior analyses of the proposed chaotic systems

To demonstrate the excellent performance of the proposed chaos system, we present and analyze the chaotic behavior through several chaos tests.

5.4.1 Bifurcation diagram

In first, the relationship between the control parameter r and the chaotic behavior of the standard logistic, sine, Chebyshev system and their enhanced maps is represented by the bifurcation diagrams depicted in Figure 5.8, which provide details of the system dynamics, as follows:

The only range in which the logistic, sine maps and Chebyshev system may exhibit chaotic behavior is $[3, 4]$. However, even within this range, several characteristics prevent these maps from exhibiting chaotic behavior. The blank zone in their bifurcation diagrams provides evidence of this. Second, the data range of chaotic sequences is less than $[0, 1]$, indicating a nonuniform distribution in that range. The bifurcation diagrams of the enhanced $T - LM$, $T - SM$ and $T - CH$. Within the range of $[0, 4]$ display uniform distributions.

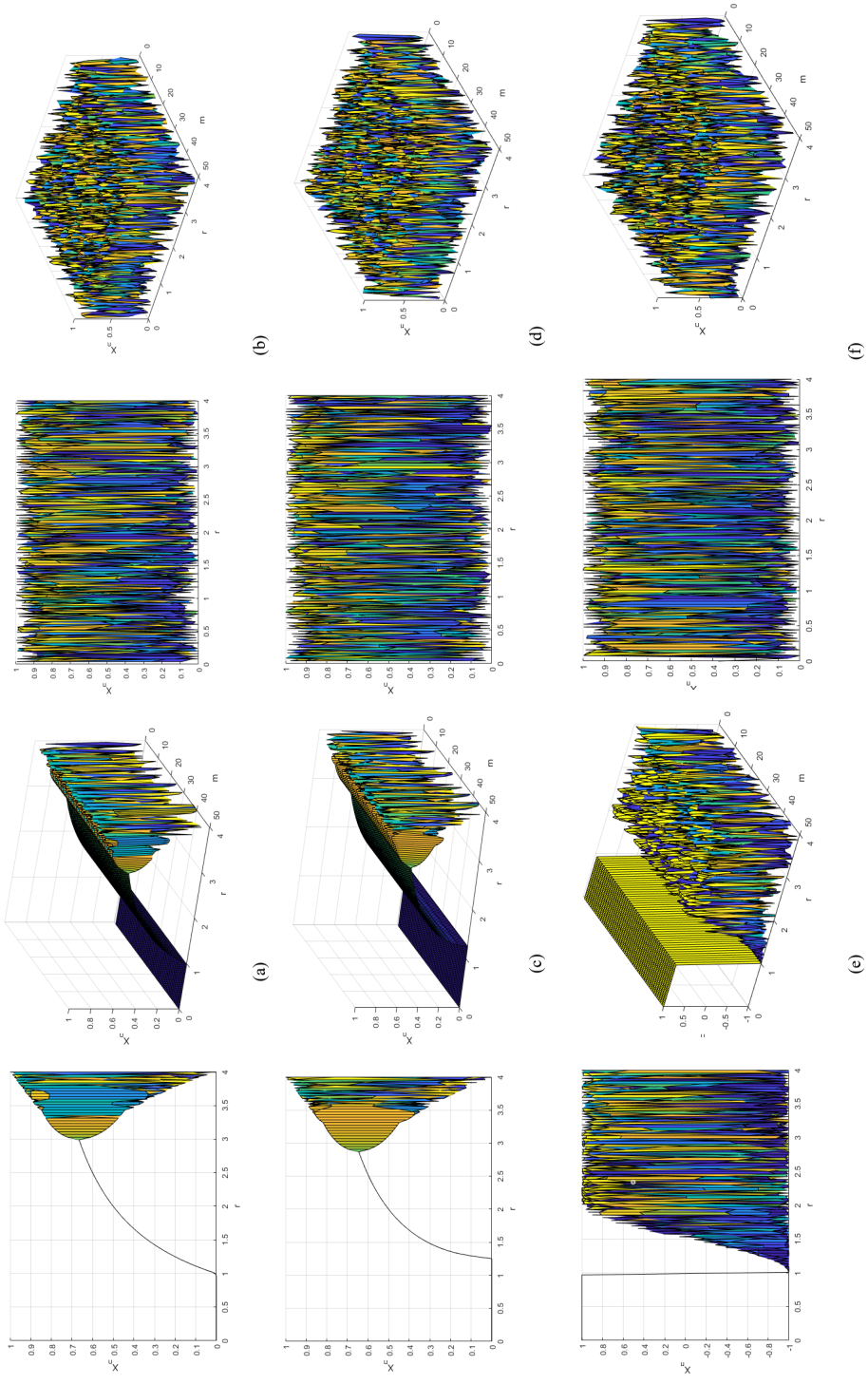
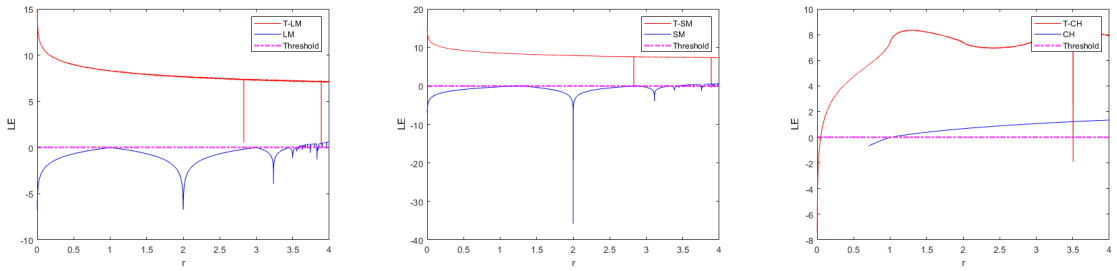


Figure 5.8: 1D/3D Bifurcation diagram plot visualization: (a) Logistic map;(b) Enhanced T-LM;(c) Sine map; (d) Enhanced T-SM; (e) Chebyshev system; (f) Enhanced T-CH

5.4.2 Lyapunov Exponent

Second, according to Figure 5.9, the LE of the three enhanced maps with their seed maps were estimated using the system parameters $(r, X_0) = (3.99, 0.1)$. It is clear that the $T-LM$, $T-SM$ and $T-CH$ systems obtain positive LEs in every parameter setting, while their seed maps have positive LEs in only a few parameters (the LE of the standard logistic and sine mapping is positive when $r > 3.5$, and when $r > 1$ for the Chebyshev system). This indicates that our proposed model can generate new chaotic maps that have wider chaotic ranges with more complex chaotic behaviors than their seed maps. The LE of the enhanced maps and their seeds may also be computed for a sample of points near the attractor to obtain an average LE. Table 5.2 lists the average LE computed for the standard and enhanced chaotic maps for two values of the parameter r. The numerical results agree quite well with the Definition 2.2.1.



(a) (b) (c)

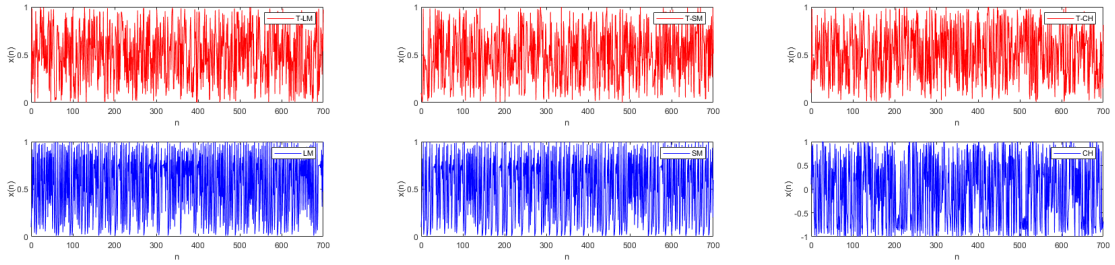
Figure 5.9: Lyapunov exponent:(a) LM and T-LM; (b) SM and T-SM; (c) CH and T-CH

Table 5.2: Average LE computed using the first derivative method

Maps	$r = 1.99$	$r = 3.99$
LM	-4.60486	0.640111
T-LM	0.0000935538	0.000128085
SM	-4.3936	0.649876
T-SM	0.000165365	0.0000871103
CH	-0.000130063	-0.0000927121
T-CH	7.62308	7.84133

5.4.3 Time series

Figure 5.10 shows the time responses of the standard logistic map, sine map, and Chebyshev system with their enhanced versions of iterations for parameter values: $r = 3.99$, $X_0 = 0.1$. The results reveal that the conventional logistic map iterated fewer than 350 times before entering a period (the same situation for the sine map and Chebyshev system, which iterated less than 200 times). The sequence formed by the improved chaotic map did not enter a cycle despite more than 5000 iterations. These results indicated that the improved approach successfully delayed the entry of the map into the cycle which proves its nonperiodicity.

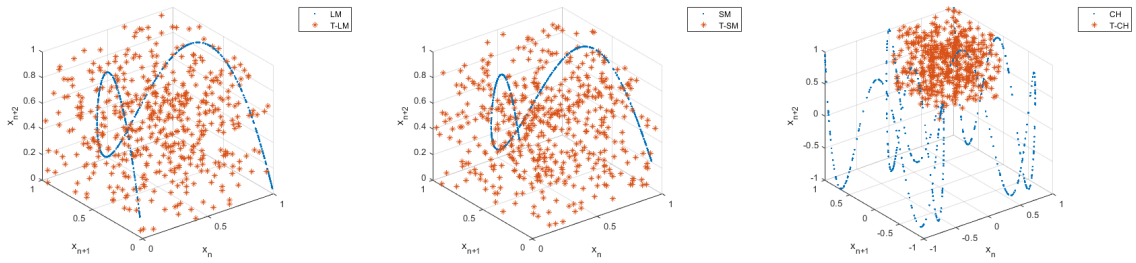


(a) (b) (c)

Figure 5.10: Time series plot: (a) T-LM and LM;(b) T-SM and SM; (c) T-CH and CH

5.4.4 Phase diagram

The phase diagram of the discrete dynamical system $x_{n+1} = f(x_n)$ defines the discrete sets generated by successive iterations of x_n and x_{n+1}. Figure 5.11 shows the mapping diagrams of the standard logistic map, sine map, Chebyshev system, and their enhanced maps. The phase diagrams of the original logistic, sine, and Chebyshev maps are a fixed upside-down U with an extremely low density that does not traverse the entire diagram space, whereas those of the improved maps ($T-LM$, $T-SM$, and $T-CH$) have no fixed shape, are much denser than the original, and present a full mapping. The full mapping of the proposed systems corresponds to a stronger chaotic intensity and larger iteration interval. In conclusion, the improved maps exhibit better chaotic performance than the original maps.

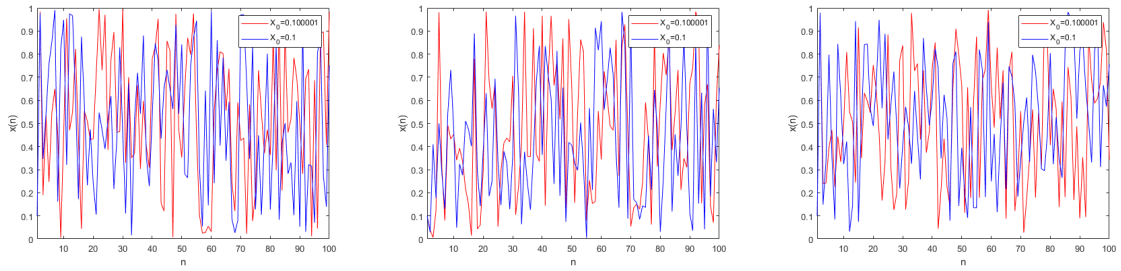


(a) (b) (c)

Figure 5.11: Phase diagram (Mapping diagram) for growth rate parameter values $r = 3.99$ and initial condition $X_0 = 0.1$ of:(a) Standard logistic map and T-LM;(b) Standard sine map and $T - SM$; (c) Standard Chebyshev system and T-CH

5.4.5 Sensitivity to initial conditions

We produced two sequences using the proposed enhanced chaotic maps with the same parameters alongside initial values that differ by 10^{-6} to demonstrate their sensitivity. Focusing on the first 100 numbers, Figure 5.12 shows the output sequences, in which the first iterations are slightly similar, but the values of the output sequences significantly differ after approximately 20 iterations, proving the high sensitivity to the initial value of the proposed enhanced T-LM, T-SM and T-CH.



(a) (b) (c)

Figure 5.12: Sensitive dependence of initial conditions for growth rate parameter values $r = 3.99$ and variable initial condition $X_0 = 0.1$ of: (a) $T - LM$; (b) $T - SM$; (c) $T - CH$

Table 5.3: Results of the NIST test of randomness of the enhanced generated sequences

Test	**T-LM**		**T-SM**		**T-CH**	
	P_{value}	Conclusion	P_{value}	Conclusion	P_{value}	Conclusion
Frequency Test (Monobit)	0.527089	Pass	0.849515	Pass	0.100096	Pass
Frequency Test within a Block	0.999644	Pass	0.045894	Pass	0.690441	Pass
Run Test	0.1108407	Pass	0.128740	Pass	0.817046	Pass
Longest Run of Ones in a Block	0.3394374	Pass	0.146969	Pass	0.745521	Pass
Binary Matrix Rank Test	-1.0	Fail	-1.0	Fail	-1.0	Fail
Discrete Fourier Transform (Spectral) Test	0.383988	Pass	0.383988	Pass	0.561657	Pass
Non-Overlapping Template Matching Test	0.850728	Pass	0.850728	Pass	0.987271	Pass
Overlapping Template Matching Test	nan	Fail	nan	Fail	nan	Fail
Maurer's Universal Statistical test	-1.0	Fail	-1.0	Fail	-1.0	Fail
Linear Complexity Test	0.029634	Pass	0.919688	Pass	0.985608	Pass
Serial test	0.938168	Pass	0.778440	Pass	0.900023	Pass
	0.996920	Pass	0.570425	Pass	0.926785	Pass
Approximate Entropy Test	0.994387	Pass	0.999706	Pass	0.999174	Pass
Cummulative Sums (Forward) Test	0.941731	Pass	0.508615	Pass	0.200192	Pass
Cummulative Sums (Reverse) Test	0.823132	Pass	0.676714	Pass	0.133271	Pass
Random Excursions Test	0.422512	Pass	0.656029	Pass	0.720532	Pass

5.4.6 NIST statistical test

Table 5.3 lists the NIST SP-800 test results obtained after processing 1000 binary sequences. Each value in the P_{value} column represents the results of the uniformity testing of the P_{value} computed for a given test. In conclusion, the sequence generated by the enhanced $T-LM$, $T-SM$ and $T-CH$ are accepted as random because the majority of P_{value} for each of the 15 tests were ≥ 0.01, indicating that our proposed system is indeed a highly effective implementation of a random number generator.

5.4.7 01- test

Table 5.4: 0-1 test results: k values obtained for the r values mentioned

Maps	$r = 0.99$	$r = 1.99$	$r = 2.99$	$r = 3.99$
LM	-0.460023996545	-0.154134859472	-0.166985740582	0.996279819756
T-LM	0.999268180600	0.999247856285	0.999131880504	0.997458644034
SM	-0.171583740804	-0.127622977572	0.998154669423	0.999438040534
T-SM	0.995856794258	0.998731524058	0.996521827130	0.996790949789
CH	-0.112821072653	0.998385881096	0.995745765214	0.999097442215
T-CH	0.997264092579	0.998744686251	0.996919721000	0.993903370223

In the first step of the 0-1 test, and for $c \in (0, \pi)$, we compute and plot the translation variables p and q. Typical plots of p and q for regular and chaotic dynamics are shown in Figure 5.13. It rigorously shows that p and q are bounded when $r = 1.99$ in the case of standard logistic and sine maps, indicating that the underlying dynamics are regular (periodic or quasiperiodic). When r approaches 3.99, p and q behave asymptotically similarly to Brownian motion for all maps.

However, even at $r = 1.99$ of their enhanced version, the $p-q$ graph exhibits Brownian motion in Figure 5.13(b),(d) and (f), symbolizing chaotic behavior in all ranges of $r \in [0-4]$

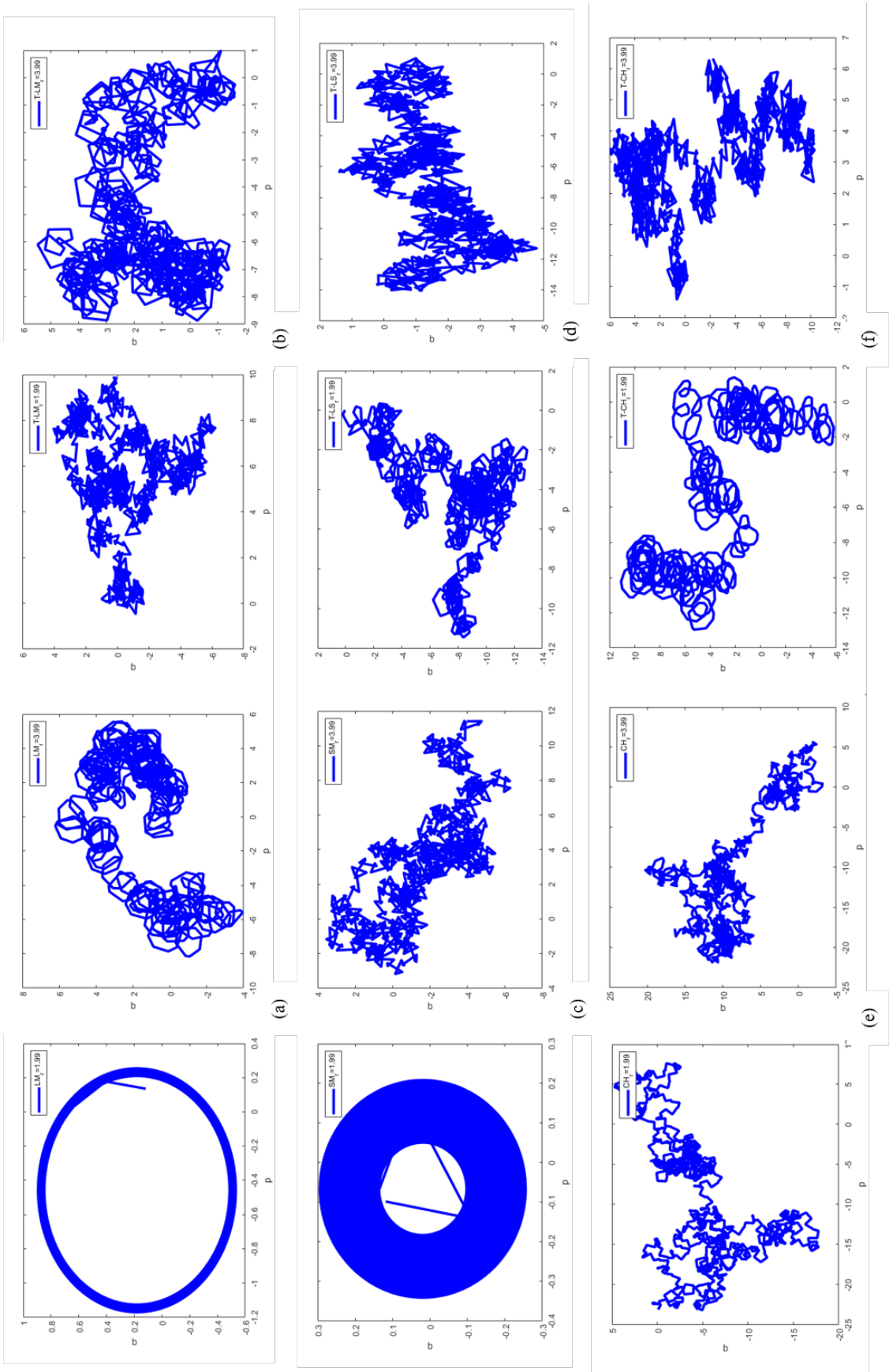


Figure 5.13: Plot of p versus q for $r = 1.99$ and $r = 3.99$ of :(a) LM; (b) T-LM ;(c) SM; (d) T-SM; (e) CH; (f) T-CH

Then, we computed the asymptotic growth rate K of the mean-square displacement tabulated in Table 5.4. As presented in Figure 5.14, the asymptotic growth rate K of the proposed enhanced maps approaches 1 for all ranges of r, proving the existence of chaos.

Finally, after computing $M(n)$, the mean square displacement scales linearly with time in the case of the enhanced chaotic maps (Figure 5.15). This indicates that the motion is chaotic, while the oscillatory plot of the standard logistic and sine maps when $r = 1.99$ is an indication of regularity.

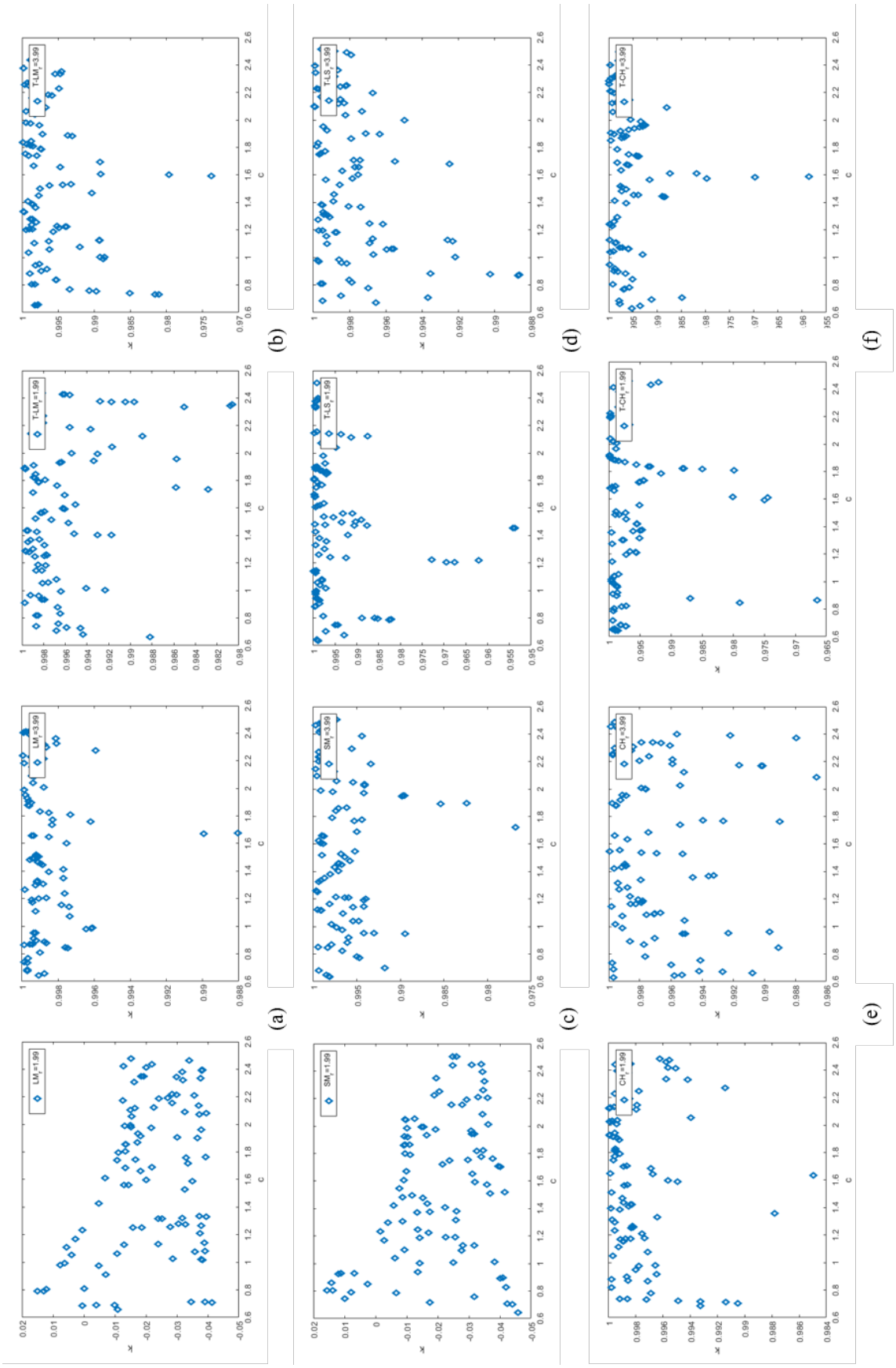


Figure 5.14: Plot of K versus c for $r = 1.99$ and $r = 3.99$ of :(a) LM; (b) T-LM ;(c) SM; (d) T-SM; (e) CH; (f) T-CH

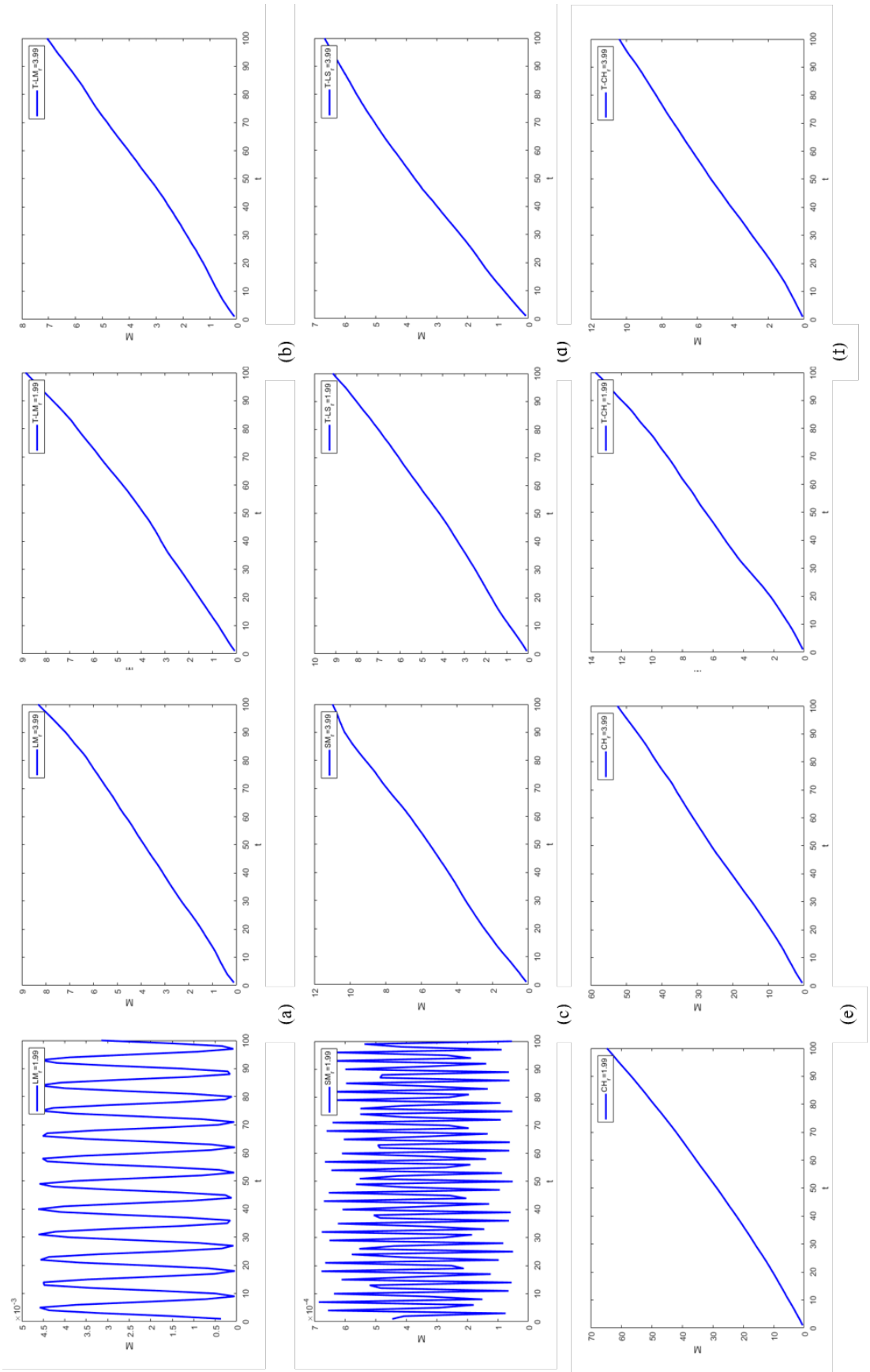


Figure 5.15: Mean square displacement for $r = 1.99$ and $r = 3.99$ of :(a) LM; (b) T-LM ;(c) SM; (d) T-SM; (e) CH; (f) T-CH

5.5 Application to image cryptography

Selecting the most accurate chaotic map for permutation and diffusion is a very essential stage for designing a cryptosystem. Through the results we previously obtained and demonstrated, it seems that the enhanced $T-CH$ is an excellent system that provides the broadest chaotic range (proved through BD) with strong chaotic behavior in all ranges of $r \in [0-4]$ (proved through LE and 0-1 test), which makes it a suitable choice for diffusion phase. On the other hand, $T-LM$ and T-SM showed almost the same behavior for most tests; however, based on the NIST test, $T-LM$ showed decent performance in terms of randomization; thus, it was selected for permutation.

In this section, we propose a novel image cryptosystem where the enhanced $T-LM$ is used to permute the plain image, and the enhanced $T-CH$ system is used for the diffusion of the permuted image. P represents the plain image, E represents the cipher image, X represents the chaotic variables, C represents the permuted vector, K represents the diffusion key, and G represents the secret matrix.

First, we iterate the $T-LM$ $M \times N$ times to obtain a vector of sequences X_{T-LM} with $M \times N$ chaotic variables:

$$X_{T-LM} = \{X_{T-LM}(1), X_{T-LM}(2), .., X_{T-LM}(M \times N)\} \tag{5.11}$$

Then, as presented in Figure 5.16, we generate the permutation vector V by sorting X_{T-LM} in ascending order:

$$V = \{V(1), V(2), ..., V(M \times N)\} \tag{5.12}$$

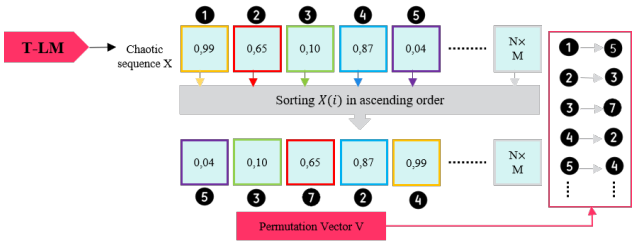


Figure 5.16: Permutation vector generation block diagram

Second, we load the image P of size $M \times N$ to be encrypted and convert it into a vector of pixel values by applying a row major alignment:

$$P = \{P(1), P(2), .., P(M \times N)\} \tag{5.13}$$

We then apply permutation operations on the image pixels P with the permutation vector V:

$$C(i) = P(V(i)) \tag{5.14}$$

Figure 5.17 illustrates the operation.

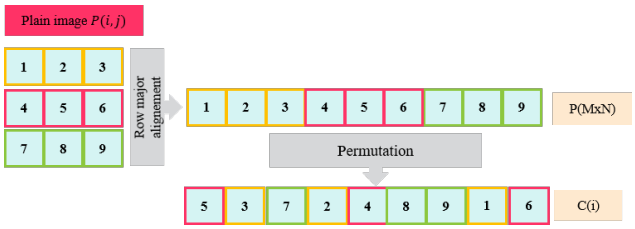


Figure 5.17: Image pixel's permutation block diagram

In the diffusion step, we iterate first the enhanced T-CH system for $M \times N$ times, and then obtaining a second sequence $X_{(T-CH)}$ with $M \times N$ chaotic variables:

$$X_{T-CH} = \{X_{T-CH}(1), X_{T-CH}(2), ..., X_{T-CH}(M \times N)\} \tag{5.15}$$

Then, we quantize then rounding X_{T-CH} to the nearest decimal:

$$K_1(i) = round(X_{T-CH}(i) \times 255) \tag{5.16}$$

In the second step, we convert the $K_1(i)$ vector into binary. Then, we circshift the elements of the binary vector in a circular manner along one dimension $R(i) = Circleshift(K_1(i), 1)$ as illustrated in Figure 5.16. We Convert $R(i)$ to a decimal vector $K_2(i)$.

$K_1(i) \oplus K_2(i)$ is performed to obtain secret matrix G sized $M \times N$, where $\oplus$ represents the XOR operation of the corresponding elements.

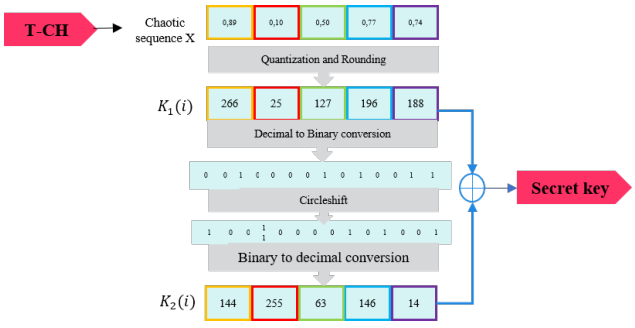


Figure 5.18: Generation of the secret key for diffusion

In the encryption step, the permutated vector $P(V(i))$ is XORed with the secret key vector and then reshaped to a matrix of $M \times N$ size (through a row major order) which

represents the encrypted image E (Figure 5.17). The decryption process is roughly the reverse of the encryption process.

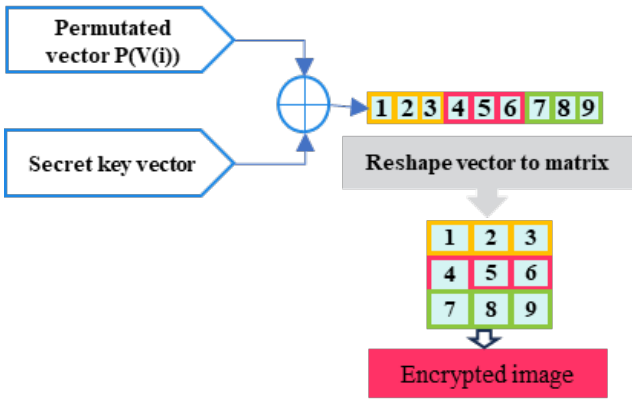


Figure 5.19: Final step of encryption processes

5.6 Tests and evaluation of the proposed cryptosystem

Simulation experiments of the proposed cryptography technique were performed using a computer machine with Windows 10, Intel(R) Core (TM) i3-4030U CPU @1.90 GHz, and 4 GB RAM. The compilation software application employed is MATLAB R2023a. In this section, the proposed cryptography method is analyzed against multiple metrics to test its security and efficiency.

5.6.1 Visual analysis

One of the most significant metrics used to analyze the resilience and effectiveness of cryptography algorithms is visual encryption/decryption analysis. Figure 5.20(c) and Figure 5.20(e) demonstrate the ciphering/deciphering findings of the proposed encryption algorithm on standard Lena, Baboon, Baot, and Pepper. The proposed cryptography algorithm succeed in hiding images features, and can effectively and efficiently recover and decipher the images.

The histogram plot is used also to demonstrate the pixel strength distribution and rates of an image. According to Figure 5.20, the histogram distribution of the plain images fluctuates substantially from that of the ciphered images, concealing the tangible form of the examined images, showing that there are no barely identifiable arrangements/shapes in the consistently encrypted images. Additionally, the histogram distributions of the deciphered images are analogous to those of the plain images. Thus, with enhanced quality, our cryptography

algorithm can effectively and profitably obtain the histogram distribution of the plain images. Consequently, the histogram distribution results validate the trustworthiness of the proposed cryptography algorithm.

To verify our proposed cryptosystem against the sensitivity of the key, a key is taken and an image is encrypted (Baboon image as example); then, the value of the key is changed, and an image is encrypted again. The same analysis is also performed in the decryption process. The output images were compared to determine whether there was any match. To perform this analysis, the first key set (referred as K_1) is:

$$\begin{cases} r_{T-LM} = 3.699, X_0 = 0.1 \\ r_{T-CH} = 3.699, X_0 = 0.1 \end{cases} \tag{5.17}$$

K_1 is used to encrypt the original image (Figure 5.21(a)). The encrypted image is E_1 (Figure 5.21(b)). To create a different key set designated as K_2, we apply a small change to $r_0 = 3.699000000000001$ while maintaining X_0 unchanged. K_2 is used to encrypt the original image in Figure 5.21(a) and obtain another encrypted image (E_2) in Figure 5.21(c).

The simulation results and the pixel-to-pixel difference $|E_1 - E_2|$ clarify that the encrypted image changes significantly as a consequence of tiny changes in the original image and key.

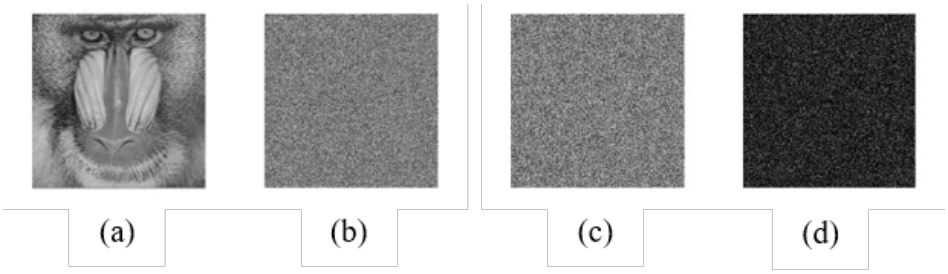


Figure 5.21: Key sensitivity analysis: (a) Original image; (b) Encrypted image E1 using K_1; (c) Encrypted image E2 using K_2; (d)|E1-E2|

Figure 5.22(a) and (b) represent the decryption results. Only the correct key (K_1) allows for full reconstruction of the plain image. However, even a slight modification in the security key (K_2) would lead to image decryption failure (Figure 5.22(b)). Hence, the proposed technique has great key sensitivity in both encryption and decryption procedures.

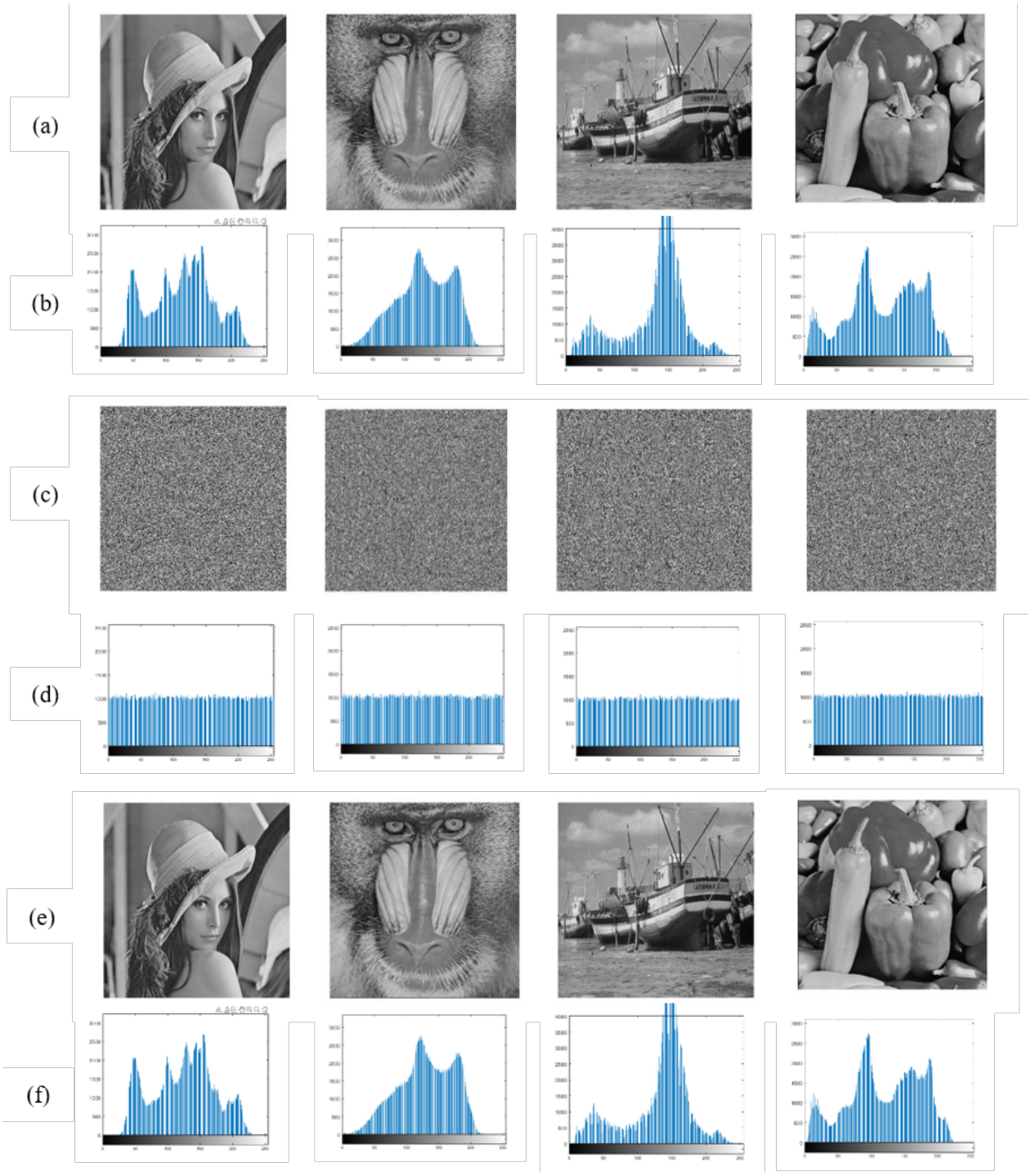


Figure 5.20: Visual result with histogram of: (a) Original images; (b) Histograms of original images; (c) Encrypted images; (d) Histogram of encrypted images; (e) Decrypted images; (f) Histograms of decrypted images

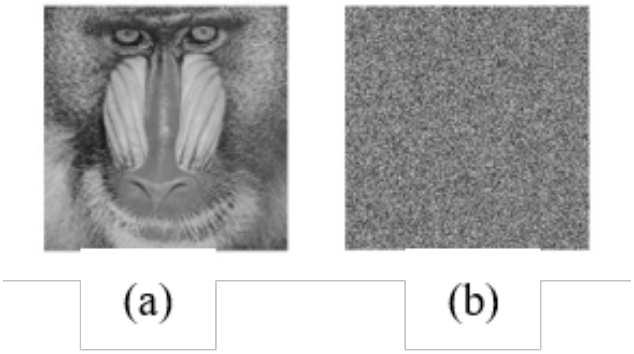


Figure 5.22: Key sensitivity analysis: (a) Decrypted with correct key (K_1); (b) Decrypted with wrong key (K_2)

5.6.2 Diffusion and quality analysis

The IE values of the original, encrypted and decrypted images using the proposed cryptosystem are listed in Table 5.5 and remain tightly close to 8. As a result, it is exceedingly difficult to extract visual information from ciphered images.

Table 5.5: Entropy score results

Entropy	**Size**	**Original**	**Encrypted**	**Decrypted**
Baboon	(512×512)	7.358336	7.999130	7.358336
Peppers	(512×512)	7.593654	7.999407	7.593654
Boat	(512×512)	7.191370	7.999306	7.191370
Lena	(256×256)	7.568278	7.997069	7.568278
Lena	(512×512)	7.445567	7.999296	7.445567
House	(256×256)	6.496137	7.996924	6.496137

The correlation distributions for Lena, Baboon, Boat, and Peppers are shown in Figure 5.23. The neighboring pixels of the original images clearly show a significant correlation, whereas there is no association between the neighboring pixels of the ciphered image. The correlation coefficients between neighboring pixels of the other images were then calculated in the horizontal, vertical, and diagonal directions. It is evident that the correlation coefficients of the encrypted images listed in Table 5.6 are very low and almost zero in all directions. This demonstrates the effectiveness of our system against statistical attacks.

Figure 5.23: Plot of 2048 pairs of original (left) and encrypted (right) images of :(a) Lena image; (b)Baboon image; (c) Boat image;(d) Peppers image

Table 5.6: Horizontal, vertical, and diagonal corr values of 2048 pairs of original, encrypted and decrypted images

	Images	**H**	**V**	**D**
Baboon (512 × 512)	Plain image	0.877458	0.752179	0.688948
	Encrypted image	0.006586	0.003231	0.014287
	Decrypted image	0.857260	0.767713	0.688945
Peppers(512 × 512)	Plain image	0.978434	0.978598	0.969805
	Encrypted image	-0.01114	-0.015605	-0.006433
	Decrypted image	0.974132	0.978476	0.959845
Boat (512 × 512)	Plain image	0.944906	0.969785	0.916866
	Encrypted image	0.006929	0.004725	0.005771
	Decrypted image	0.935935	0.972577	0.922063
Lena (256 × 256)	Plain image	0.941732	0.961876	0.921423
	Encrypted image	-0.009373	-0.000614	0.003381
	Decrypted image	0.935019	0.974460	0.914812
Lena (512 × 512)	Plain image	0.972887	0.988436	0.961178
	Encrypted image	0.041911	-0.019721	0.003213
	Decrypted image	0.978204	0.979625	0.962220
House (256 × 256)	Plain image	0.980126	0.954224	0.940687
	Encrypted image	0.041911	-0.019721	0.003213
	Decrypted image	0.977627	0.945177	0.941592

5.6.3 Avalanche effect analysis

The rate of NPCR is shown to have a maximum predicted value of 99.61% when two random images are taken into account, while the UACI's maximum anticipated value is about 33%. Table 5.8 shows that the values for the NPCR are close to the aforementioned predicted values, obtained UACI values are nearly equal to the defined critical values. Hence, the proposed encryption technique guarantees the necessary resistance to any kind of differential cryptanalysis.

Second, the key space used by the proposed method represents the sum of the keys utilized as part of the key-generator system (see Table 5.7). The total key-space is $\approx 10^3 \times 10^{15} \times 10^{15} \times 10^{15} \times 10^{15} \approx 10^{63}$ which is roughly superior 2^{100}. Thus, the suggested

approach has a large key space to fend against brute force attacks.

Table 5.7: Initial conditions and parameters that form the secret key

Keys	**Symbol**	**Definition**
Key 1	$X_0 T-LM$	Initial condition of the enhanced $T-LM$ map
Key 2	$X_0 T-CH$	Initial condition of the enhanced $T-CH$ map
Key 3	r_{T-LM}	Bifurcation parameter of the enhanced $T-LM$ map
Key 4	r_{T-CH}	Bifurcation parameter of the enhanced $T-CH$ map
Key 5	n	Iteration number (depend on the input image size)

In general, as shown in Table 5.8, we compare the proposed cryptosystem in terms of various performance metrics with recent literature in Ref [148], [149], [150] [151], and [152]. Our algorithm achieves the expected average of entropy, PSNR, NPCR and UACI since they fit within the range of comparable research, with satisfactory comparable computational execution time and can resist common attacks.

Table 5.8: Performance comparison of various encryption schemes

Image	**Test**	**Proposed**	**Ref** [148]	**Ref**[149]	**Ref**[150]	**Ref**[151]	**Ref**[152]
Lena	Entropy	7.999296	7.9989	7.9994	7.999355	7.9994	7.9994
	PSNR	44.478900	45.621	-	9.1370	-	-
	NPCR(%)	99.601364	99.587	99.6124	-	99.6109	99.6136
	UACI(%)	28.563031	30.701	33.4474	-	33.4779	33.4651
	Elapsed time (s)	0.483554	1.5	-	-	-	-
Baboon	Entropy	7.999130	7.9989	7.9992	7.999250	-	7.9994
	PSNR	45.080905	44.636	-	8.8478	-	-
	NPCR(%)	99.588012	99.616	99.5983	-	99.6141	99.6071
	UACI(%)	27.834538	27.886	33.4797	-	33.3760	33.4226
	Elapsed time (s)	0.482553	1.5069	-	-	-	-
Boat	Entropy	7.999306	7.9977	7.9993	-	-	-
	PSNR	45.080905	44.636	-	8.8478	-	-
	NPCR(%)	99.600982	99.601	-	-	-	-
	UACI(%)	28.563031	28.482	-	-	-	-
	Elapsed time (s)	0.489511	0.56989	-	-	-	-
Pepper	Entropy	7.999407	7.9989	7.9994	7.999301	-	-
	PSNR	43.723919	45.716	-	8.8439	-	-
	NPCR(%)	99.573898	99.601	99.6090	-	99.6170	-
	UACI(%)	29.530711	31.036	33.4744	-	33.4168	-
	Elapsed time (s)	0.479660	1.5813	-	-	-	-
	Key space	2^{210}	2^{798}	2^{212}	-	2^{199}	2^{425}

5.7 Conclusion

To produce highly independent and uniformly distributed random chaotic sequences, this chapter introduced a tangent nonlinear transformation technique for enhancing and resilient chaotification in discrete-time chaotic systems. To demonstrate the excellent performance of the proposed chaotic system, the chaotic characteristics of the proposed T-LM, T-SM, and T-CH maps, along with wide-ranging dynamical tests, including the bifurcation diagram, trajectory analysis, Lyapunov exponent, phase diagram, NIST, and 01 test, were used to demonstrate the unpredictability and ergodicity of the three enhanced chaotic maps. Addi-

tionally, compared with their seed maps, the proposed maps demonstrate a broader chaotic range, more complicated behavior, and a more competitive complexity order. Following the validation of its performance as an efficient chaotic system, the proposed enhanced T-LM and T-CH maps were employed in permutation and diffusion to design a new encryption scheme that facilitates the secure exchange of digital images via networks.

General conclusion

Powerful cryptographic systems require a qualified random number generator. Most of these random numbers are generated by chaotic generators that use chaotic maps. Traditional one-dimensional maps are vulnerable to attacks if they are not properly used. A hybrid chaos map can solve such issues. Thus, cryptographers must design schemes that have a sufficiently large key space by employing different 1D chaotic map output sequences. In addition, encryption algorithms should strike a balance between security and computational efficiency.

In view of this topic, we have proposed the following contributions:

In the first contribution, new key-stream generators based on the feedback shift register and chaos named LSS-LFSR-PRNG, LM-NLFSR-PRNG, and LSS-NLFSR-PRNG were presented and successfully applied and tested for image encryption after passing fifteen rigorous NIST tests. These generators provide high cryptographic performance, fulfill statistical security requirements (high entropy, uniform distribution, and near-zero correlation between ciphered images), and offer various intriguing characteristics, such as sufficient key space reaches even 2^{224}.

In the second contribution, three chaos-based image encryption algorithms (1D LM-based cryptosystem, 1D LM-Chebyshev-based cryptosystem, and the 3D ILM-based cryptosystem) were developed using several confusion-diffusion architectures, that provide robust security while minimizing computational overhead. By discussing the impact of choosing the right chaotic dimension in enhancing the security of the encryption algorithm, these cryptosystems were tested and compared across key spaces, resistivity to several attacks, execution times, and amounts of memory usage. Based on the good achieved results of entropy, correlation, the NPCR, and UACI, we demonstrated that high -dimensional chaotic maps are capable of generating unpredictable keystreams, that are highly suitable for encryption and may enhance the security and efficiency of the cryptosystem; however, these maps fail to maintain

computational complexity in terms of time execution and memory usage in comparison to simple 1D maps.

In the last contribution, we presented a tangent nonlinear transformation method for improving chaotification in discrete-time chaotic systems, with the aim of creating highly independent and uniformly distributed random chaotic sequences. We have demonstrated the excellent performance in terms of nonlinearity, unpredictability, and nonperiodicity of the proposed chaotic system by means of several tests, such as a bifurcation diagram, trajectory analysis, Lyapunov exponent, phase diagram, NIST, and 01 test. Additionally, compared with their seed maps, the proposed maps (T-LM, T-SM, and T-CH) demonstrate a broader chaotic range that extends over [0-4], and more complicated behavior. After validating its performance as an efficient chaotic system, the proposed enhanced maps were employed in permutation and diffusion to design a new encryption scheme that facilitates the secure exchange of digital images via networks.

Challenges

Despite significant advancements in image encryption techniques, several challenges remain in the field to provide a foundation for future research. These challenges constitute obstacles in reaching robust, secure, and efficient image encryption solutions. First, chaotic maps can achieve low performance if the input parameters are not chosen wisely. This problem should be solved by using metaheuristic techniques in image encryption to obtain optimized initial conditions.

Second, some tested schemes based on newly designed chaotic maps have good statistical results in terms of randomness, but fail when ciphertext and plaintext attacks are conducted. Thus, the complete range of the chaotic and nonchaotic regions of the proposed schemes must be presented, which is crucial for creating a highly secure nonlinear key for an encryption algorithm that renders it impervious to several types of assaults.

Third, computational complexity and extensive hardware requirements are still open areas of research in this field, and the majority of designed chaotic systems still suffer from digital implementation, which makes real-world applications impractical.

Fourth, different image formats are available, such as JPEG, PNG, BMP, and TIFF, each with its own characteristics. It is difficult to create image encryption algorithms that support various image formats while maintaining their security features. Encryption methods must possess the ability to adjust to various image structures and guarantee strong security for a variety of image file formats.

Future directions

Future research should focus on a wider range of topics. First, we can explore the integration of image encryption with other emerging technologies; for example, combining chaotic systems with DNA coding, S-boxes, machine learning models, deep learning networks and complex mathematical models, to solve the problem of image security and open new avenues for research into practical application. However, the effective integration of these technologies requires further study.

Chaotic maps may lead to erroneous findings since their encryption is very sensitive to the initial conditions. Thus, we aim to explore metaheuristic techniques to provide a solution for wisely selecting the input parameters to obtain optimized results. In addition, as multimedia data, including images, audio, and video data, become more prevalent, we aim to develop an encryption algorithm that handles multiple types of data simultaneously. This method can synchronize and preserve the integrity of the whole multimedia stream while encrypting multimedia material effectively and securely.

Additionally, hardware implementation of the designed algorithm is envisaged across FPGA, and its ability to generate true random numbers for encryption purposes can overcome issues related to security, key management, robustness, and scalability.

Finally, to keep up with the rapid development of quantum cryptography, which has gained significant attention in recent years due to its potential for providing ultimate security assurance, we may explore quantum-resistant images by leveraging quantum principles such as entanglement and superposition to develop quantum-safe encryption algorithms.

Bibliography

[1] Komal Sahota. Digital transmission in computer network. *Journal of Analog and Digital Communications*, 7(2):8–12, 2022.

[2] Luca Crocetti, Stefano Di Matteo, Pietro Nannipieri, Luca Fanucci, and Sergio Saponara. Design and test of an integrated random number generator with all-digital entropy source. *Entropy*, 24(2):139, 2022.

[3] Rameez Raja Kureshi and Bhupesh Kumar Mishra. A comparative study of data encryption techniques for data security in the iot device. In *Internet of Things and Its Applications: Select Proceedings of ICIA 2020*, pages 451–460. Springer, 2022.

[4] Aarti Devi, Ankush Sharma, and Anamika Rangra. A review on des, aes and blowfish for image encryption & decryption. *International Journal of Computer Science and Information Technologies*, 6(3):3034–3036, 2015.

[5] Shahtaj Shaukat, ALI Arshid, Amna Eleyan, Syed Aziz SHAH, Jawad AHMAD, et al. Chaos theory and its application: an essential framework for image encryption. *Chaos Theory and Applications*, 2(1):17–22, 2020.

[6] Xiuli Chai, Xiaoyu Zheng, Zhihua Gan, Daojun Han, and Yiran Chen. An image encryption algorithm based on chaotic system and compressive sensing. *Signal Processing*, 148:124–144, 2018.

[7] Rong Chen, Xiaomeng Li, Lin Teng, and Xingyuan Wang. An image encryption algorithm based on the lscmm chaotic map and bidirectional dynamic diffusion. *Multimedia Tools and Applications*, 83(2):3681–3706, 2024.

[8] Fatih Özkaynak. Cryptographically secure random number generator with chaotic additional input. *Nonlinear Dynamics*, 78(3):2015–2020, 2014.

[9] Gokce Yildirim and Erkan Tanyildizi. Random number generator based on optimization and chaotic system. In *2023 11th International Symposium on Digital Forensics and Security (ISDFS)*, pages 1–4. IEEE, 2023.

[10] Haini Zeng and Deli Chen. Image encryption algorithm based on logistic-sine compound chaos. In *2020 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*, pages 120–123. IEEE, 2020.

[11] Yicong Zhou, Zhongyun Hua, Chi-Man Pun, and CL Philip Chen. Cascade chaotic system with applications. *IEEE transactions on cybernetics*, 45(9):2001–2012, 2014.

[12] Yicong Zhou, Long Bao, and CL Philip Chen. A new 1d chaotic system for image encryption. *Signal processing*, 97:172–182, 2014.

[13] Zongchao Qiao. *Nonlinear dynamics, applications to chaos-based encryption.* PhD thesis, École centrale de Nantes, 2021.

[14] Robert M Haralick and Linda G Shapiro. Glossary of computer vision terms. *Pattern Recognit.*, 24(1):69–93, 1991.

[15] Khanhvi Tran, Johan Peter Bøtker, Arash Aframian, and Kaveh Memarzadeh. Artificial intelligence for medical imaging. In *Artificial Intelligence in Healthcare*, pages 143–162. Elsevier, 2020.

[16] Nuria Rodríguez-Ortega. Techno-concepts for the cultural field: n-dimensional space and its conceptual constellation. *Multimodal Technologies and Interaction*, 6(11):96, 2022.

[17] James C Grande. Principles of image analysis, 2012.

[18] Yevgeniya Sulema, Etienne Kerre, and Oksana Shkurat. Vector image retrieval methods based on fuzzy patterns. *International Journal of Modern Education and Computer Science*, 12(3):8, 2020.

[19] Jun Haeng Lee. Handling digital images for publication. *Sci Ed*, 1(2):58–61, 2014.

[20] Punyaban Patel, Bibekananda Jena, Bibhudatta Sahoo, Pritam Patel, and Banshidhar Majhi. Study of noise removal techniques for digital images. In *Biometrics: Concepts, Methodologies, Tools, and Applications*, pages 1105–1144. IGI Global, 2017.

[21] Chen Fangfang, ZHU Jianle, CAO Peng, and Huo Peijun. Print anti-replication technology based on am/fm hybrid halftone. In *2017 IEEE 17th International Conference on Communication Technology (ICCT)*, pages 329–332. IEEE, 2017.

[22] Xinyun Yao. Design and optimization of a 4-bit absolute-value detector using half adder and comparator. In *Journal of Physics: Conference Series*, volume 2435, page 012009. IOP Publishing, 2023.

[23] Euclid Seeram and Euclid Seeram. Digital image processing concepts. *Digital Radiography: Physical Principles and Quality Control*, pages 21–39, 2019.

[24] Gulshan Shrivastava, Aakanksha Pandey, and Kavita Sharma. Steganography and Its Technique: Technical Overview. In Vinu V. Das, editor, *Proceedings of the Third International Conference on Trends in Information, Telecommunication and Computing*, Lecture Notes in Electrical Engineering, pages 615–620, New York, NY, 2013. Springer.

[25] Ahmed Hamdy El-Kady, Syeda Halim, Hans Pasman, and Faisal Khan. Security of digitalized process systems. In *Methods in Chemical Process Safety*, volume 6, pages 479–523. Elsevier, 2022.

[26] KM Iftekharuddin and F Ahmed. Detection| image post-processing and electronic distribution. 2005.

[27] Talbi Mourad. Image watermarking using data compression. In *Image Watermarking Techniques*, pages 1–12. Springer, 2023.

[28] Jörg Schwenk. *Guide to Internet Cryptography: Security Protocols and Real-World Attack Implications*. Springer Nature, 2022.

[29] Samiksha Sharma. Cryptography: An art of writing a secret code. *International Journal of Computer Science & Technology*, 8491(1), 2017.

[30] Unsub Zia, Mark McCartney, Bryan Scotney, Jorge Martinez, Mamun AbuTair, Jamshed Memon, and Ali Sajjad. Survey on image encryption techniques using chaotic maps in spatial, transform and spatiotemporal domains. *International Journal of Information Security*, 21(4):917–935, 2022.

[31] Ashish S Dongare, AS Alvi, and NM Tarbani. An efficient technique for image encryption and decryption for secured multimedia application. *International Research Journal of Engineering and Technology (IRJET)*, 4(4):3186–3190, 2017.

[32] Hadjer Bourekouche, Samia Belkacem, and Noureddine Messaoudi. Lightweight medical image encrypting and decrypting algorithm based on the 3d intertwining logistic map. *International Journal of Informatics and Applied Mathematics*, 6(2):46–62.

[33] A John Blesswin, G Selva Mary, and S Manoj Kumar. Multiple secret image communication using visual cryptography. *Wireless Personal Communications*, 122(4):3085–3103, 2022.

[34] Sahar Haddad. *Protection of encrypted and/or compressed medical images by means of watermarking*. PhD thesis, Ecole nationale supérieure Mines-Télécom Atlantique, 2020.

[35] Amany Sarhan, Fatma Elgendy, Tarek Eltobely, and Osama S Faragallah. C8. an efficient approach for image cryptosystem based on chaotic confusion-diffusion mechanisms. In *2012 29th National Radio Science Conference (NRSC)*, pages 211–221. IEEE, 2012.

[36] Rajkumar Banoth and Rekha Regar. An introduction to classical and modern cryptography. In *Classical and Modern Cryptography for Beginners*, pages 1–46. Springer, 2023.

[37] Thomas W Edgar and David O Manz. Science and cyber security. *Research methods for cyber security*, pages 33–62, 2017.

[38] Chandra Prakash Dewangan, Shashikant Agrawal, Akash Kumar Mandal, and Archana Tiwari. Study of avalanche effect in aes using binary codes. In *2012 IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, pages 183–187. IEEE, 2012.

[39] Kuen Park and Heejo Lee. A taxonomy of online game security. In *Encyclopedia of Internet technologies and applications*, pages 606–611. IGI Global, 2008.

[40] Sura F Yousif. Encryption and decryption of audio signal based on rsa algorithn. *International Journal of Engineering Technologies and Management Research*, 5(7):57–64, 2018.

[41] Akash Kumar Mandal, Chandra Parakash, and Archana Tiwari. Performance evaluation of cryptographic algorithms: Des and aes. In *2012 IEEE Students' Conference on Electrical, Electronics and Computer Science*, pages 1–5. IEEE, 2012.

[42] Mousa Farajallah, Safwan El Assad, and Olivier Deforges. Cryptanalyzing an image encryption scheme using reverse 2-dimensional chaotic map and dependent diffusion. *Multimedia Tools and Applications*, 77:28225–28248, 2018.

[43] Jun-xin Chen, Zhi-liang Zhu, Chong Fu, Hai Yu, and Yushu Zhang. Reusing the permutation matrix dynamically for efficient image cryptographic algorithm. *Signal Processing*, 111:294–307, 2015.

[44] Rizwan Haider. Modified and efficient image encryption algorithm based on chaos theory, dna complementary rules and sha-256, 2017.

[45] Rajkumar Banoth and Rekha Regar. *Classical and Modern Cryptography for Beginners*. Springer Nature, 2023.

[46] Sharmila Ghosh, Purba Pal, and Nirmalya Kar. An analysis of chaos-based cryptographic algorithms. In *2023 Fifth International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, pages 1–7. IEEE, 2023.

[47] Sanaz Rahimi Moosavi and Arman Izadifar. End-to-end security scheme for e-health systems using dna-based ecc. In *Silicon Valley Cybersecurity Conference*, pages 77–89. Springer, 2021.

[48] B Rahul, K Kuppusamy, and A Senthilrajan. Dynamic dna cryptography-based image encryption scheme using multiple chaotic maps and sha-256 hash function. *Optik*, 289:171253, 2023.

[49] Morteza SaberiKamarposhti, Amirabbas Ghorbani, and Mehdi Yadollahi. A comprehensive survey on image encryption: Taxonomy, challenges, and future directions. *Chaos, Solitons & Fractals*, 178:114361, 2024.

[50] Mandeep Kaur, Surender Singh, and Manjit Kaur. Computational image encryption techniques: a comprehensive review. *Mathematical Problems in Engineering*, 2021:1–17, 2021.

[51] Snehlata Yadav and Namita Tiwari. Recent advancements in chaos-based image encryption techniques: a review. *Social Networking and Computational Intelligence: Proceedings of SCI-2018*, pages 639–647, 2020.

[52] Shi Liu, Changliang Guo, and John T Sheridan. A review of optical image encryption techniques. *Optics & Laser Technology*, 57:327–342, 2014.

[53] Poonam Yadav, Hukum Singh, and Kavita Khanna. Introducing real-time image encryption technology using key vault, various transforms, and phase masks. *Multimedia Tools and Applications*, pages 1–19, 2023.

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

[55] Chiranjoy Chattopadhyay, Bikramjit Sarkar, and Debaprasad Mukherjee. Encoding by dna relations and randomization through chaotic sequences for image encryption. *arXiv preprint arXiv:1505.01795*, 2015.

[56] Walid El-Shafai, Iman M Almomani, and Aala Alkhayer. Optical bit-plane-based 3d-jst cryptography algorithm with cascaded 2d-frft encryption for efficient and secure hevc communication. *IEEE Access*, 9:35004–35026, 2021.

[57] Walid El-Shafai, Iman M Almomani, and Aala Alkhayer. Optical bit-plane-based 3d-jst cryptography algorithm with cascaded 2d-frft encryption for efficient and secure hevc communication. *IEEE Access*, 9:35004–35026, 2021.

[58] Rajesh Kumar, Ajit Kumar Verma, Tarun K Sharma, Om Prakash Verma, and Sanjay Sharma. *Soft Computing: Theories and Applications: Proceedings of SoCTA 2022*. Springer, 2023.

[59] Seyed Mohammad Seyedzadeh and Sattar Mirzakuchaki. A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map. *Signal processing*, 92(5):1202–1215, 2012.

[60] Amina Souyah and Kamel Mohamed Faraoun. Fast and efficient randomized encryption scheme for digital images based on quadtree decomposition and reversible memory cellular automata. *Nonlinear Dynamics*, 84(2):715–732, 2016.

[61] Zhenlong Man, Jinqing Li, Xiaoqiang Di, and Ou Bai. An image segmentation encryption algorithm based on hybrid chaotic system. *IEEE access*, 7:103047–103058, 2019.

[62] N Mahendiran and C Deepa. A comprehensive analysis on image encryption and compression techniques with the assessment of performance evaluation metrics. *SN Computer Science*, 2(1):29, 2021.

[63] Joseph Yves Effa, Jean De Dieu Nkapkop, Mihaela Cislariu, and Monica Borda. Comparative analysis of different structures of chaos-based cryptosystems: A survey. *Acta Technica Napocensis*, 57(2):36, 2016.

[64] Moises García-Martínez and Eric Campos-Cantón. Pseudo-random bit generator based on multi-modal maps. *Nonlinear Dynamics*, 82(4):2119–2131, 2015.

[65] Benyamin Norouzi, Sattar Mirzakuchaki, Seyed Mohammad Seyedzadeh, and Mohammad Reza Mosavi. A simple, sensitive and secure image encryption algorithm based on hyper-chaotic system with only one round diffusion process. *Multimedia tools and applications*, 71:1469–1497, 2014.

[66] Anand B Joshi and Abdul Gaffar. A technique for securing digital audio files based on rotation and xor operations. *Soft Computing*, pages 1–18, 2023.

[67] Yanpeng Zhang, Wenjie Dong, Jing Zhang, and Qun Ding. An image encryption transmission scheme based on a polynomial chaotic map. *Entropy*, 25(7):1005, 2023.

[68] Luigi Troiano, Alfredo Vaccaro, Nishtha Kesswani, Irene Díaz Rodriguez, Imene Brigui, and David Pastor-Escuredo. *Key Digital Trends in Artificial Intelligence and Robotics: Proceedings of 4th International Conference on Deep Learning, Artificial Intelligence and Robotics,(ICDLAIR) 2022-Progress in Algorithms and Applications of Deep Learning*, volume 670. Springer Nature, 2023.

[69] Kurunandan Jain, Aravind Aji, and Prabhakar Krishnan. Medical image encryption scheme using multiple chaotic maps. *Pattern Recognition Letters*, 152:356–364, 2021.

[70] Vinod Patidar, NK Pareek, and KK Sud. A new substitution–diffusion based image cipher using chaotic standard and logistic maps. *Communications in Nonlinear Science and Numerical Simulation*, 14(7):3056–3075, 2009.

[71] James E. Gentle. *Random Number Generation and Monte Carlo Methods*. Statistics and Computing. Springer New York, New York, NY, 1998.

[72] Andrew Rukhin, Juan Soto, James Nechvatal, Miles Smid, Elaine Barker, Stefan Leigh, Mark Levenson, Mark Vangel, David Banks, Alan Heckert, et al. *A statistical test suite for random and pseudorandom number generators for cryptographic applications*, volume 22. US Department of Commerce, Technology Administration, National Institute of . . . , 2001.

[73] Limin Fan, Hua Chen, and Si Gao. A general method to evaluate the correlation of randomness tests. In *Information Security Applications: 14th International Workshop, WISA 2013, Jeju Island, Korea, August 19-21, 2013, Revised Selected Papers 14*, pages 52–62. Springer, 2014.

[74] Mohammed Bakiri. *Hardware implementation of a pseudo random number generator based on chaotic iteration*. PhD thesis, Bourgogne Franche-Comté, 2018.

[75] Mohammed Bakiri, Christophe Guyeux, Jean-François Couchot, and Abdelkrim Kamel Oudjida. Survey on hardware implementation of random number generators on fpga: Theory and experimental analyses. *Computer Science Review*, 27:135–153, 2018.

[76] Elena Almaraz Luengo and Javier Román Villaizán. Cryptographically secured pseudo-random number generators: Analysis and testing with nist statistical test suite. *Mathematics*, 11(23):4812, 2023.

[77] Werner Schindler. Random number generators for cryptographic applications. *Cryptographic Engineering*, pages 5–23, 2009.

[78] Sandhya Rangineni. *Cryptographic analysis of random sequences*. PhD thesis, Oklahoma State University, 2011.

[79] Ganiyev Salim Karimovich, Khudoykulov Zarif Turakulovich, and Halimtoeva Ikbola Ubaydullayevna. Computer's source based (pseudo) random number generation. In *2017 International Conference on Information Science and Communications Technologies (ICISCT)*, pages 1–6. IEEE, 2017.

[80] Marius Iulian Mihailescu and Stefania Loredana Nita. *Cryptography and Cryptanalysis in MATLAB*. Springer, 2021.

[81] Ciprian RĂCUCIU, Constantin DOCHIŢOIU, Nicolae JULA, and Marian PEARSICĂ. Aspects of image transmission on the governemental communications systems. 2005.

[82] Von Neumann. Various techniques used in connection with random digits. *Notes by GE Forsythe*, pages 36–38, 1951.

[83] Michael Mascagni and Ashok Srinivasan. Parameterizing parallel multiplicative lagged-fibonacci generators. *Parallel Computing*, 30(7):899–916, 2004.

[84] Luca Crocetti, Pietro Nannipieri, Stefano Di Matteo, Luca Fanucci, and Sergio Saponara. Review of methodologies and metrics for assessing the quality of random number generators. *Electronics*, 12(3):723, 2023.

[85] KV Ramya, Manjunatha Reddy HS, C Bharathi, and Mamata Dhananjaya. Cryptographic strength in resource-constrained iot: Xtea vs. rectangle. *International Journal of Intelligent Systems and Applications in Engineering*, 12(8s):599–605, 2024.

[86] Trishansh Bhardwaj et al. Pseudo random bit generation using arithematic progression. In *2015 Fifth International Conference on Advanced Computing & Communication Technologies*, pages 361–366. IEEE, 2015.

[87] George Datseris and Ulrich Parlitz. Dynamical systems. In *Nonlinear Dynamics: A Concise Introduction Interlaced with Code*, pages 1–19. Springer, 2022.

[88] George Datseris and Ulrich Parlitz. *Nonlinear dynamics: a concise introduction interlaced with code*. Springer Nature, 2022.

[89] Michael J Grimble and Paweł Majecki. *Nonlinear Industrial Control Systems*. Springer, 2020.

[90] Elbert EN Macau. *A mathematical modeling approach from nonlinear dynamics to complex systems*, volume 22. Springer, 2018.

[91] Shahtaj Shaukat, ALI Arshid, Amna Eleyan, Syed Aziz SHAH, Jawad AHMAD, et al. Chaos theory and its application: an essential framework for image encryption. *Chaos Theory and Applications*, 2(1):17–22, 2020.

[92] Hena Rani Biswas, Md Maruf Hasan, and Shujit Kumar Bala. Chaos theory and its applications in our real life. *Barishal University Journal Part*, 1(5):123–140, 2018.

[93] Joan S Muthu and P Murali. Review of chaos detection techniques performed on chaotic maps and systems in image encryption. *SN Computer Science*, 2:1–24, 2021.

[94] Priyajit Biswas, Shyamalendu Kandar, and Bibhas Chandra Dhara. An image encryption scheme using sequence generated by interval bisection of polynomial function. *Multimedia Tools and Applications*, 79:31715–31738, 2020.

[95] Robert L Devaney. Dynamics of simple maps. In *Proceedings of Symposia in Applied Mathematics*, volume 39, pages 1–24, 1989.

[96] Joan S Muthu and P Murali. Review of chaos detection techniques performed on chaotic maps and systems in image encryption. *SN Computer Science*, 2:1–24, 2021.

[97] Mahdieh Ghazvini, Mojdeh Mirzadi, and Negin Parvar. A modified method for image encryption based on chaotic map and genetic algorithm. *Multimedia Tools and Applications*, 79:26927–26950, 2020.

[98] Parsa Sarosh, Shabir A Parah, and G Mohiuddin Bhat. Fast image encryption framework for medical images. In *2021 2nd International conference on intelligent engineering and management (ICIEM)*, pages 149–154. IEEE, 2021.

[99] R Ponuma and R Amutha. Compressive sensing based image compression-encryption using novel 1d-chaotic map. *Multimedia Tools and Applications*, 77:19209–19234, 2018.

[100] Liu Rui. New Algorithm for Color Image Encryption Using Improved 1D Logistic Chaotic Map. *The Open Cybernetics & Systemics Journal*, 9(1), April 2015.

[101] Kanika Suneja, Shelza Dua, and Mohit Dua. A review of chaos based image encryption. In *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, pages 693–698. IEEE, 2019.

[102] Robert M May and George F Oster. Bifurcations and dynamic complexity in simple ecological models. *The American Naturalist*, 110(974):573–599, 1976.

[103] R Ponuma and R Amutha. Compressive sensing based image compression-encryption using novel 1d-chaotic map. *Multimedia Tools and Applications*, 77:19209–19234, 2018.

[104] Sandeep Draksharam, Dheeraj Katravulapalli, K Rohith Krishna, and V Thanikaiselvan. Analysis of hybrid chaotic image encryption. In *2018 Second international conference on electronics, communication and Aerospace technology (ICECA)*, pages 697–703. IEEE, 2018.

[105] Iztok Fister Jr, Matjaž Perc, Salahuddin M Kamal, and Iztok Fister. A review of chaos-based firefly algorithms: perspectives and research challenges. *Applied Mathematics and Computation*, 252:155–165, 2015.

[106] Nitya Ranjan Manihira and Alpesh Kumar Dauda. Image encryption using chaotic maps and dna encoding. *International Journal of Engineering Research & Technology*, 10(11):1–5, 2022.

[107] Roshini Premnath, Sridevi Arumugham, Sivaraman Rethinam, C Lakshmi, and Amirtharajan Rengarajan. Performance evaluation of chaotic maps & attractors in image encryption. In *2019 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–5. IEEE, 2019.

[108] Yang Liu, Jindong Zhang, Dongqi Han, Peibin Wu, Yiding Sun, and Young Shik Moon. A multidimensional chaotic image encryption algorithm based on the region of interest. *Multimedia Tools and Applications*, 79:17669–17705, 2020.

[109] Gurupungav Narayanan, Rishika Narayanan, Nihal Haneef, Nagaratna B Chittaragi, and Shashidhar G Koolagudi. A novel approach to video steganography using a 3d chaotic map. In *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)*, pages 955–959. IEEE, 2019.

[110] Bowen Zhang and Lingfeng Liu. A novel fast image encryption algorithm based on coefficient independent coupled exponential chaotic map. *Physica Scripta*, 2024.

[111] Abir Lassoued, Olfa Boubaker, Rachid Dhifaoui, and Sajad Jafari. Experimental observations and circuit realization of a jerk chaotic system with piecewise nonlinear function. In *Recent advances in chaotic systems and synchronization*, pages 3–21. Elsevier, 2019.

[112] Sudesh Kumari, Renu Chugh, and Radu Miculescu. On the complex and chaotic dynamics of standard logistic sine square map. *Analele ştiinţifice ale Universităţii" Ovidius" Constanţa. Seria Matematică*, 29(3):201–227, 2021.

[113] R Parvaz and Mohammad Zarebnia. A combination chaotic system and application in color image encryption. *Optics & Laser Technology*, 101:30–41, 2018.

[114] Georg A Gottwald and Ian Melbourne. The 0-1 test for chaos: A review. *Chaos detection and predictability*, pages 221–247, 2016.

[115] Lucia Inglada-Perez. A comprehensive framework for uncovering non-linearity and chaos in financial markets: Empirical evidence for four major stock market indices. *Entropy*, 22(12):1435, 2020.

[116] Charalampos Haris Skokos, Georg A Gottwald, and Jacques Laskar. *Chaos detection and predictability*, volume 1. Springer, 2016.

[117] Jean Sire Armand Eyebe Fouda. Application of ordinal-array-based indicators to strange nonchaotic attractors. 2017.

[118] JS Armand Eyebe Fouda, J Yves Effa, Martin Kom, and Maaruf Ali. The three-state test for chaos detection in discrete maps. *Applied Soft Computing*, 13(12):4731–4737, 2013.

[119] JS Armand Eyebe Fouda and Wolfram Koepf. Efficient detection of the quasi-periodic route to chaos in discrete maps by the three-state test. *Nonlinear dynamics*, 78(2):1477–1487, 2014.

[120] Joan S Muthu, Aditya Jyoti Paul, and P Murali. An efficient analyses of the behavior of one dimensional chaotic maps using 0–1 test and three state test. In *2020 IEEE Recent Advances in Intelligent Computational Systems (RAICS)*, pages 125–130. IEEE, 2020.

[121] Michał Melosik and W Marszalek. On the 0/1 test for chaos in continuous systems. *Bulletin of the Polish Academy of Sciences: Technical Sciences*, (3), 2016.

[122] Michel Tosin, Marcos Vinicius Issa, Diego Matos, Alexandre Do Nascimento, and Americo Cunha Jr. Employing 0-1 test for chaos to characterize the chaotic dynamics of a generalized gauss iterated map. In *XIV Conferência Brasileira de Dinâmica, Controle e Aplicações (DINCON 2019)*, 2019.

[123] Xingyuan Wang, Nana Guan, Hongyu Zhao, Siwei Wang, and Yingqian Zhang. A new image encryption scheme based on coupling map lattices with mixed multi-chaos. *Scientific reports*, 10(1):9784, 2020.

[124] Matthias Kaas-Mason, Goran Prpić, Suthongchai Suriyasuphapong, and Nicholas Bailey. Comparison of Pseudo, Chaotic and Quantum Random Number Generators and their use in Cyber Security.

[125] Kinga Marton and Alin Suciu. On the interpretation of results from the nist statistical test suite. *Science and Technology*, 18(1):18–32, 2015.

[126] S Rohith, KN Hari Bhat, and A Nandini Sharma. Image encryption and decryption using chaotic key sequence generated by sequence of logistic map and sequence of states of linear feedback shift register. In *2014 international conference on advances in electronics computers and communications*, pages 1–6. IEEE, 2014.

[127] BASHEER H Ali, MOHAMMED J Zaite, and ABDULLAH S Al-Hashimi. Design and implementation of a key generator-based stream cipher for securing text data. *Journal of Engineering Science and Technology*, 14(6):3372–3386, 2019.

[128] Nattagit Jiteurtragool and Masayoshi Tachibana. Hybrid random number generator based on chaotic oscillator. pages MIT–133, October 2016.

[129] Debarshi Datta, Bipa Datta, and Himadri Sekhar Dutta. Design and implementation of multibit LFSR on FPGA to generate pseudorandom sequence number. pages 346–349, March 2017.

[130] Ge Yao and Udaya Parampalli. Improved transformation algorithms for generalized galois nlfsrs. *Cryptography and Communications*, 14(2):229–258, 2022.

[131] Punam Kumari Kumar and Bhaskar Mondal. Lightweight stream cipher for health care iot. In *2023 IEEE 2nd International Conference on Industrial Electronics: Developments & Applications (ICIDeA)*, pages 444–449. IEEE, 2023.

[132] Subhrajyoti Deb, Bhaskar Biswas, and Nirmalya Kar. Study of nlfsr and reasonable security improvement on trivium cipher. In *Information Systems Design and Intelligent Applications: Proceedings of Second International Conference INDIA 2015, Volume 1*, pages 731–739. Springer, 2015.

[133] Elena Dubrova. Generation of full cycles by a composition of nlfsrs. *Designs, codes and cryptography*, 73(2):469–486, 2014.

[134] Elena Dubrova. A list of maximum-period nlfsrs, 2012.

[135] Xingyuan Wang and Dahai Xu. Image encryption using genetic operators and intertwining logistic map. *Nonlinear Dynamics*, 78:2975–2984, 2014.

[136] Mohit Dua, Drishti Makhija, Pilla Yamini Lakshmi Manasa, and Prashant Mishra. 3d chaotic map-cosine transformation based approach to video encryption and decryption. *Open Computer Science*, 12(1):37–56, 2022.

[137] ZhongYun Hua, BingHang Zhou, YinXing Zhang, and YiCong Zhou. Modular chaotification model with fpga implementation. *Science China Technological Sciences*, 64(7):1472–1484, 2021.

[138] Zhongyun Hua, Yinxing Zhang, and Yicong Zhou. Two-dimensional modular chaotification system for improving chaos complexity. *IEEE Transactions on Signal Processing*, 68:1937–1949, 2020.

[139] Zhongyun Hua, Yicong Zhou, and CL Philip Chen. A new series-wound framework for generating 1d chaotic maps. In *2013 IEEE digital signal processing and signal processing education meeting (DSP/SPE)*, pages 118–123. IEEE, 2013.

[140] Zhongyun Hua and Yicong Zhou. One-dimensional nonlinear model for producing chaos. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 65(1):235–246, 2017.

[141] Yicong Zhou, Zhongyun Hua, Chi-Man Pun, and CL Philip Chen. Cascade chaotic system with applications. *IEEE transactions on cybernetics*, 45(9):2001–2012, 2014.

[142] Moatsum Alawida, Azman Samsudin, Je Sen Teh, et al. Digital cosine chaotic map for cryptographic applications. *IEEE Access*, 7:150609–150622, 2019.

[143] Zhongyun Hua, Yicong Zhou, and Hejiao Huang. Cosine-transform-based chaotic system for image encryption. *Information Sciences*, 480:403–419, 2019.

[144] Hayder Natiq, Santo Banerjee, and MRM Said. Cosine chaotification technique to enhance chaos and complexity of discrete systems. *The European Physical Journal Special Topics*, 228:185–194, 2019.

[145] 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*, pages 130–134, 2019.

[146] Zhongyun Hua, Binghang Zhou, and Yicong Zhou. Sine chaotification model for enhancing chaos and its hardware implementation. *IEEE Transactions on Industrial Electronics*, 66(2):1273–1284, 2018.

[147] Zhongyun Hua, Yicong Zhou, and Bocheng Bao. Two-dimensional sine chaotification system with hardware implementation. *IEEE Transactions on Industrial Informatics*, 16(2):887–897, 2019.

[148] Bhaskar Mondal and Jyoti Prakash Singh. A lightweight image encryption scheme based on chaos and diffusion circuit. *Multimedia Tools and Applications*, 81(24):34547–34571, 2022.

[149] Qiqi Cun, Xiaojun Tong, Zhu Wang, and Miao Zhang. A new chaotic image encryption algorithm based on dynamic dna coding and rna computing. *The Visual Computer*, pages 1–20, 2023.

[150] Xingyuan Wang and Shengnan Chen. An image encryption algorithm based on pixel bit operation and nonlinear chaotic system. *The Visual Computer*, 39(7):3123–3144, 2023.

[151] Dawei Ding, Wei Wang, Zongli Yang, Yongbing Hu, Jin Wang, Mouyuan Wang, Yan Niu, and Haifei Zhu. An n-dimensional modulo chaotic system with expected lyapunov exponents and its application in image encryption. *Chaos, Solitons & Fractals*, 174:113841, 2023.

[152] Shaohui Yan, Yu Cui, Lin Li, Binxian Gu, and Yu Ren. Dynamical analysis of a novel chaotic system and its application to image encryption. *Microelectronic Engineering*, page 112054, 2023.