GMR
The Original

A FERMIONIC MATTER CHAOTIC MAP WITH ENHANCED RANDOMNESS FOR SECURE IMAGE ENCRYPTION

Muhammad Ilyas[1], Farhan Ali[2], Awais Maqsood[3], Hasnain Kashif[3], Muhammad Ilyas[4], Shakeel Ahmad[5], Muhammad Ali[5]

[1]Department of Mathematics, University of Management and Technology, Lahore 54770, Pakistan; v31826@umt.edu.pk (M.I)
[2]School of Computer and IT, Beaconhouse National University, P.O. Box 53700 Lahore, Pakistan; farhan32748@gmail.com (F.A)
[3]EE Deptt. SEN, University of Management and Technology, Lahore 54770, Pakistan; awais.maqsood@umt.edu.pk (A.M); hasnain.kashif@umt.edu.pk (H.K);
[4]Department of software engineering, Gold campus, Superior University, Lahore 5400 Pakistan; developersai643@gmail.com (M.I)
[5]Department of Computer and Information Technology, Superior University, Lahore 5400 Pakistan; shakeel.ue.pk@gmail.com (S.A) ; ali.maths72@gmail.com (M.A)
*Correspondence: farhan32748@gmail.com

Abstract
In the digital era, information security particularly for digital images shared over the internet is of growing concern. Among the several data protection techniques, like data masking and network security, cryptography remains one of the most reliable and widely used methods. In this field chaos theory plays a vital role such as high sensitivity to initial conditions and randomness. To strengthen cryptographic keys enhancing randomness is essential to resist brute force and statistical attacks. Most existing encryption schemes employ two-dimensional chaotic maps but many still suffer from limited key spaces or weak chaotic behavior. To overcome these limitations a newly devised two-dimensional chaotic system named the Fermionic Matter Chaotic System (FMCS) is proposed in this research. This system offers an expanded range of control parameters a significantly larger key space and improved chaotic dynamics. The main objective of this study is to develop and evaluate the performance of the FMCS-based 2D chaotic map for secure image encryption. Its effectiveness will be measured using various metrics including bifurcation diagrams Lyapunov exponents entropy analysis key sensitivity histogram uniformity and correlation coefficients. Furthermore, the randomness of the generated sequences will be statistically validated using the NIST SP800-22 test suite. Finally, the FMCS will be integrated into an image encryption algorithm to assess its efficiency and robustness compared to traditional chaotic maps. The results aim to demonstrate that this newly proposed system can offer a faster more secure and highly reliable solution for protecting digital images in today's data-driven world.

1. INTRODUCTION

In today's digital world keeping information secure has become more important. Whether it's preventing cyberattacks protecting private communication or safely sharing data online security is a top priority. Data is now seen as one of the most valuable resources which makes its protection critical. With the rapid growth of technology massive amounts of data are being used across various sectors like cloud computing healthcare finance and social media. Despite the large volume of data being shared and stored, the main challenge remains: keeping that data secure. One of the most effective ways to do this is through cryptography. By using encryption, cryptography scrambles readable information into an unreadable format, making it inaccessible to anyone who isn't authorized.

Encryption is a way to protect data by turning readable information into a scram- bled format called ciphertext. This makes it impossible for anyone without the right key to understand the data, even if they intercept it [1]. Only someone with the correct decryption key can turn the data back into its original form. Between the sender and the recipient, the key function similarly to a shared secret. Figure 1, illustrates the encryption and decryption procedure. An encryption algorithm is used to transform the sender's initial plaintext into ciphertext, a certain encryption key. The ciphertext is subsequently transformed back into plaintext by the recipient using a decryption method and a decryption key. Only the intended recipient will be able to read the original message.

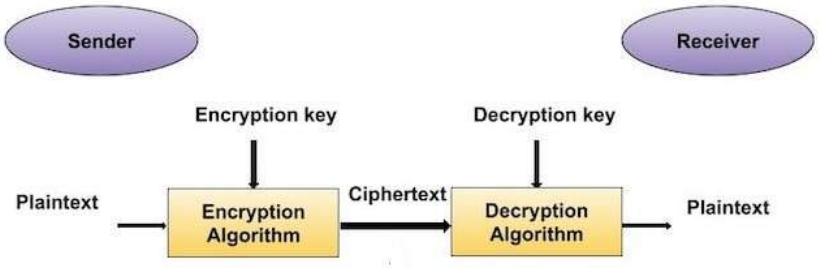


Figure 1. Process of Encryption and Decryption [2]

There are two main types of encryption. The first one is symmetric and uses the same key to encrypt or decrypt data, and the other one is asymmetric and uses different keys to encrypt and decrypt data. The role of encryption is very important for securing and preventing data from leakage in the digital world.

In the symmetrical key encryption system, the data is encrypted and detoxified using the same key. This suggests that the sender and the recipient must have the same key so that the information is transmitted stably [3]. Therefore, it is important to maintain safety. And in the hands of unauthorized people. In a situation where speed is very important, symmetrical key algorithms are ideal due to their reputation for efficiency, speed and small computing overhead costs. However, they also, especially It is also safe to distribute and store keys and expand the system for many users. Data Encryption Standard (DES) [4], Advanced Encryption Standard (AES), Enhanced DES (E-DES), and Triple DES (T-DES) [4] are a few of the popular symmetric encryption techniques. These techniques use a key to encrypt data blocks, and they can handle characters, integers, and symbols both during the encryption and decryption phases. The operation of symmetric encryption is illustrated in Figure 2. Using a secret key, it transforms legible data, known as plaintext, into unintelligible text, or ciphertext. You can safely send or store this encrypted data. The ciphertext is decrypted back into plaintext using the same secret key so that the original message may be read again. It is crucial to maintain it secure because the same key is used for both processes.

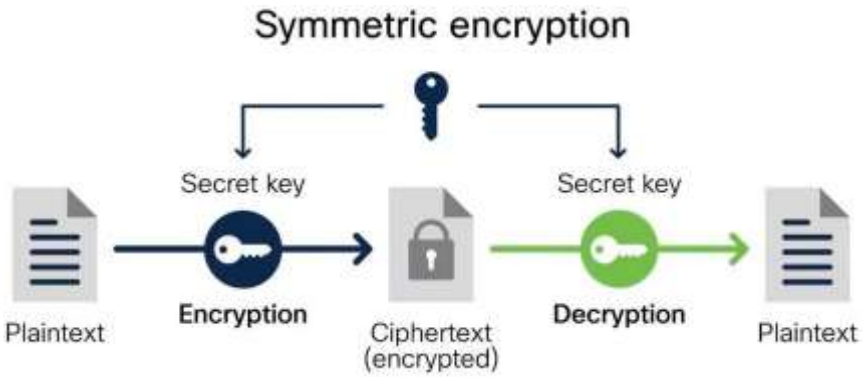


Figure 2. Symmetric Encryption [5]

The trade-off is that asymmetric algorithms are usually more complex and need more computing resources than symmetric ones. Digital signature algorithms, RSA (Rivest Shamir Adleman) [6], and Elliptic Curve Cryptography (ECC) [7] are common applications of various encryption methods. Figure 3, illustrates how asymmetric encryption works. The sender converts the original contents into a safe, unreadable format by encrypting the communication using the recipient's public key. The sender encrypts the information using the recipient's public key, transforming the original data into secure, incomprehensible ciphertext [8]. The recipient uses their private key to decrypt the communication after receiving it in order to view the original data. This technique ensures safe communication by using two distinct keys: one for locking the message and another for unlocking it.

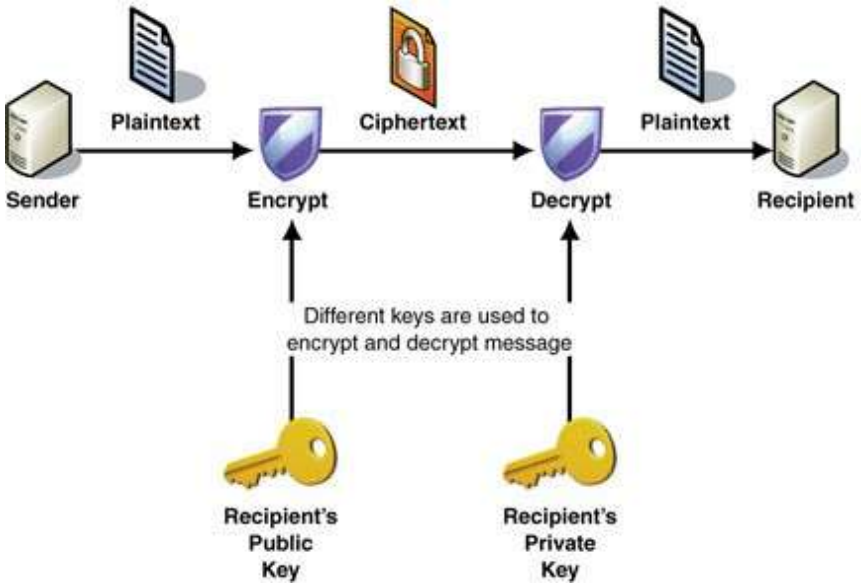


Figure 3. Asymmetric Encryption [9]

Text, audio, video, and digital photos are just a few of the numerous kinds of data that are transmitted via the internet. The format of this data frequently determines how it is encrypted. Generally speaking, text is smaller and simpler but the wording can get ruined or unintelligible if even a tiny bit of information is lost in transit. However, audio and video data are more challenging to encrypt and decode because of their magnitude. The digital images are particularly well-liked among these forms due to their ease of comprehension and perceived authenticity. The digital images are popular for sharing information because they remain visually clear even if some data is lost. This is due to the high similarity between neighboring pixels, which helps preserve image quality [10]. Even with minor errors, data within the image often stays intact. However, because of their large size, encrypting digital images requires a lot of random data to ensure strong security. In cryptography, a Random Number Generator (RNG) [**11**] is used to produce un- predictable numbers that are essential for encrypting images. These numbers play a key role in creating encryption keys and securing digital communication. There are three main types of RNGs. True Random Number Generators (TRNGs) rely on unpredictable physical processes like electronic noise to produce truly random values but they can be slower and more expensive. Pseudo Random Number Generators (PRNGs) use mathematical formulas and a starting value called a seed to generate number sequences that appear random though they follow a predictable pattern if the seed is known. Cryptographically Secure Random Number Generators (CS-RNGs) are a more advanced form of PRNGs designed specifically for cryptographic use and are built to resist prediction even if some system details are exposed. Each type has its own strengths and is chosen based on the security needs of the application.

As photographs are used more and more particularly online it is becoming more and more crucial to preserve them. Encryption which renders images unintelligible is a popular method of image security. Because chaotic systems are sensitive and unpredictable, several contemporary picture encryption techniques incorporate algorithms based on chaos. These systems, which belong to the larger class of non-linear systems, strongly resemble the fundamental ideas of cryptography, including security, randomness, and attack resistance.2D chaotic systems [12] are popular among other chaotic models for picture encryption because they provide a favorable trade-off between security and computing performance. Stronger protection may be offered by more complicated systems, such as 3D or hyper chaotic models, although their implementation is sometimes more difficult and computationally demanding. For this reason, 2D systems are a practical and effective choice for designing secure image encryption techniques.

1.1 Research Questions

- How are image encryption's security and resilience affected by a wider key space and more robust chaotic behavior?
- How effectively does the system function using common encryption standards in terms of randomness, sensitivity, and attack resistance?

1.2 Research Objectives

- To address common problems in current 2D maps, like limited unpredictability and high processing time.
- To devise a 2D chaotic system that improves randomness, flexibility, and sensitivity key elements for strong image

encryption.

- To test how well the proposed system works using standard tools like Lyapunov Exponent, correlation coefficients, entropy, histogram analysis, phase space diagram, and key sensitivity tests.

1.3 Research Methodology

The purpose of this research is to generate a 2D nonlinear chaotic system that creates pseudo random series for secure image encryption. Where control parameters, x_0, x_1, y_0, y_1 and two prime numbers p and q. The system follows a 2D chaotic equation that updates the values of and $Y\ (n\ +\ 2)$ and $X(n\ +\ 2)$ by using the current and previous values of Y and X.

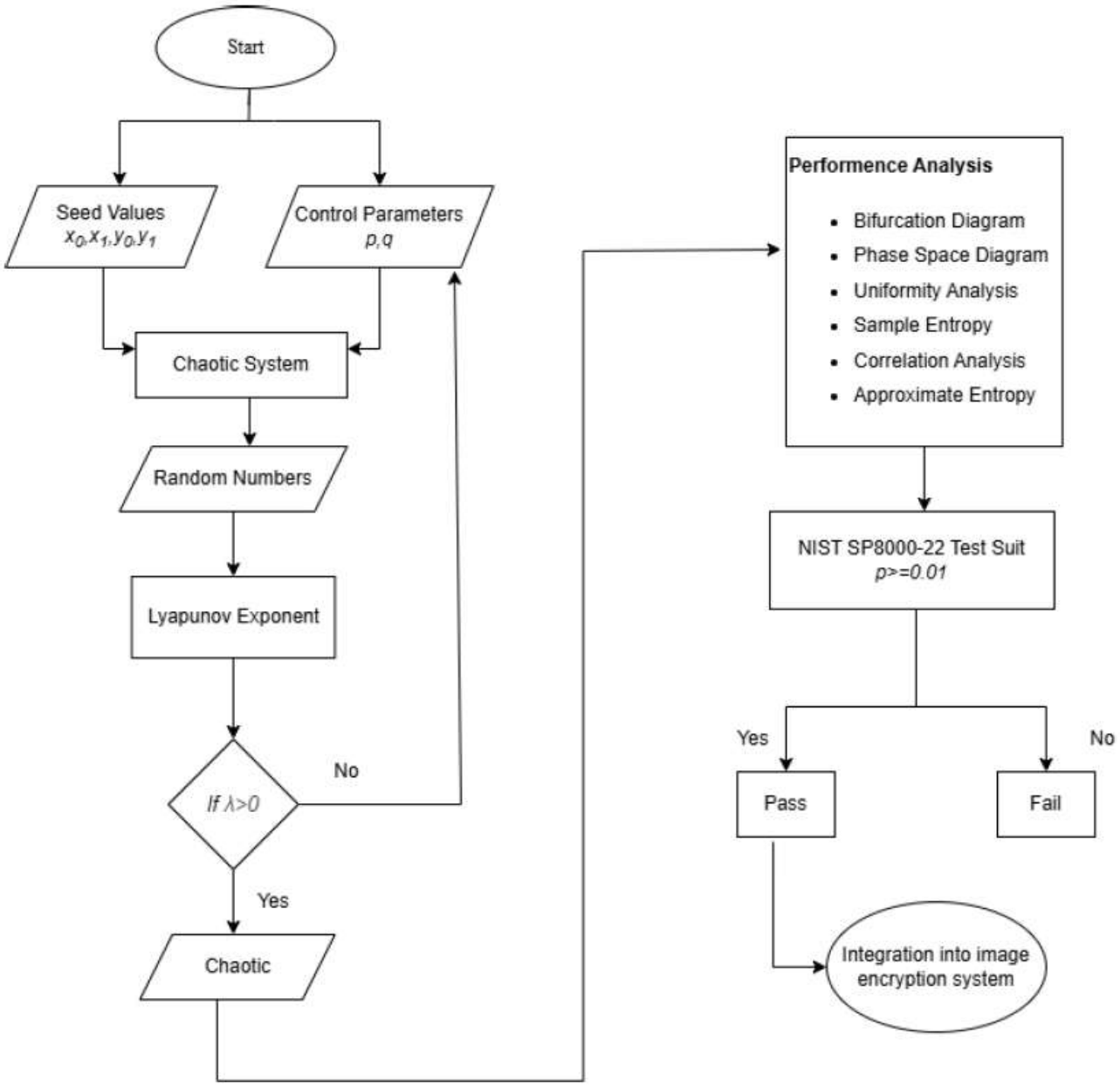


Figure 4. Proposed Methodology

The created sequence is merged into an image encryption algorithm and tests such as histogram analysis, pixel correlation, and key sensitivity are conducted to validate the encryption quality and security.

2. PRELIMINARIES STUDIES

In this part, we discuss one, two and three-dimensional chaotic system and gaps in existing techniques about uses of cryptographic, fundamentals of chaotic systems, high-quality random number generators. The need to protect image data is rising as technology develops. This is vital in domains including relational communication, military service, and medical imaging. Images keep more detailed and sensitive information than the plain text. Unauthorized access can result in serious problems including privacy violation or security risks. To avoid this image are encrypted to reduce them unintelligible. Only those can decrypt them that have correct decryption key. However, traditional encryption methods don't always work well with images. This is because similar pixels and patterns in images stay close to one another. These patterns make them more challenging to defend using text-based strategies. As a result, encryption approaches based on images work well. These methods must handle large file sizes, pixel patterns, and rapid processing. In the best methods, Chaos-based encryption is one of them. Chaotic systems are enormously unpredictable. A small change in input causes a big change in the result. This helps hide image data and protects it from brute force and statistical attacks.

Chaotic systems follow clear rules but their consequences stay highly unpredictable. Even a small change in the initial value can cause the result to be completely different. Where randomness and unpredictability are needed, they work well for encryption. These systems use mathematics but you still cannot predict them without knowing the exact initial value. This protects the encrypted data because only the person who has the right key decryption can decrypt it. Chaotic systems are fast and lightweight so they work well on low-power devices like IoT gadgets and smart- phones. Chaotic systems are fast, simple, and secure which makes them a good option for modern encryption, especially for images.

A random number generator is a vital part of modern cryptography. Its assembly's data and creates secure keys. If patterns exist, attackers can break the system. Ancient or simple RNGs cannot meet the strict security needs of cryptography. Many numbers are not very random and repeat often. Attackers can take advantage of these weaknesses using statistics or brute force. Chaos-based RNGs are better because they produce very unpredictable numbers. They create high randomness by using the sensitive and nonlinear nature of chaotic systems. Using many chaotic systems at once can make numbers more random and still maintain high speed. This creates complex sequences that do not repeat. They are ideal for use in low-power devices like embedded systems and mobile phones.

The earliest tools for chaos-based encryption were one-dimensional chaotic maps like logistic, tent, and sine maps. They gained popularity due of their simplicity. However, they are limited. Their haphazard actions only function in narrow ranges. In certain situations, they may also start to become predictable. Re- searchers attempted to address issue by incorporating more terms or blending maps. One example that enhances unpredictability is the Digital Cosine Chaotic Map. Despite these modifications, one-dimensional maps continue to fall short of contemporary security requirements. Thus, more complex systems like two-dimensional and hybrid chaotic maps are currently being studied by researchers.

Chaotic maps in two dimensions increase sensitivity and complexity. Two sequences can be created simultaneously using maps like the Arnold Cat Map and the Logistic Sine Map. This improves the manner that image pixels are mixed. At- tackers find it more difficult to determine the true image since these maps conceal pixel position and value. Some researchers create hybrid systems by combining many maps. These strategies make use of each map's advantages while minimizing its disadvantages. Two-dimensional maps provide strong unpredictability and high entropy, according to tests. They also withstand a lot of assaults. They are therefore an excellent option for image encryption.

Chaotic systems in three dimensions are more complicated than those in two dimensions. They disperse changes in numerous directions and produce rich random sequences. They are therefore extremely safe. They provide strong defense against statistical and brute force attacks. Additionally, they offer more key space, which boosts security. To increase unpredictability in three-dimensional systems, some researchers combine the logistic, tent, and sine functions. However, these systems require greater computing power. Therefore, they are not necessarily appropriate for low power or real-time applications. Although they are strong, they are not always useful.

2.1 Literature Review

It is known that such systems generate more complex and unpredictable behavior that is ideal in activities that require encryption. To solve issues such as small key space, weak randomization, and inefficiency in computing, scholars have been striving to enhance 2D models. Table 1, gives a comparative summary of the 2D chaotic systems that have been covered in this

section, giving a summary of their principal contribution to encryption along with the challenges that are connected to them. Wang et al. [13] introduced a new 2D logistic-sine chaotic map that represents the advantages of both logistic and sine maps. Encryption is harder to attack statistically as well as brutally compared to more traditional 1D methods due to its complexity and high output of random sequences. The 2D Sine Arcsin Cos Arcsin (2D-SACA) chaotic map was created to overcome such challenges as uneven chaotic distribution and limited parameter space [14]. This approach increases the resistance and hardiness of encryption systems to differential attacks by increasing the range of control parameters that can be used. In order to increase the complexity of encryption, scholars combined the 2D-Iterative Gaussian Sine Chaotic Map (2D-IGSCM) [15] with a modified Hill cipher. This algorithm improves encryption and demonstrates great effectiveness against statistical and selected-plaintext attacks because of highly dynamic key generation and pixel manipulation procedures.

Liu et al. [16] created a 2D Nonlinear Logistic-Sine Chaotic Map (2D-NLSCM) which is based on adaptive picture scrambling and bidirectional diffusion. Their strategy enhances the performance measure such as correlation, UACI and NPCR and is also more resistant to noise and manipulation by adjusting based on the image content. A different interesting research was one where encryption and picture compression are used to produce a 2D hyper chaotic map via the Schaffer function [17]. It is suitable in storage-constrained applications since it operates by employing chaos together with compressive sensing to reduce the size of images without reducing the encryption power. The 2D Sine-Henon Hybrid Map (2D-SHHM) is a map created by researchers to maintain the local and global picture attributes [18]. It is a method that is used to combine two well-known chaotic models to enhance resilience to various types of attacks and best applied on sensitive content such as grayscale images and medical images. Li et al. [19] introduced the 2D Sine-Arcsin Hyperbolic Map (2D-SAHM) a new 2D chaotic system that used together with DNA encoding methods is an extremely complex and secure encryption structure. Another useful and efficient method is a 2D Logistic Map that was developed to be used to mimic quick pixel scrambling and diffusion [20]. This paradigm, which puts the emphasis on the speed of encryption but at the same time allows to ensure the required randomness and safety, was designed to be used in the real time processing.
Lastly, a lightweight 2D Piecewise Linear Chaotic Map (2D-PWLCM) was suggested to generate encryption keys with a high level of entropy and yet being computationally simple [21]. As an example, the initial models such as the 2D Logistic-Sine Map [13] were enhanced by complexity but had a limitation in the adjustment of the parameters. Equally, the 2D-SACA map [14] system was designed to reduce an imbalanced chaotic distribution, although its performance depended strongly on the selection of parameters. More sophisticated variants such as the 2D-IGSCM [15] and 2D-NLSCM [16] added adaptive processes and dynamic scrambling to eliminate such weaknesses but at the expense of adding to the computational complexity. Furthermore, though it has been integrated with other advanced techniques such as compressive sensing [17] and DNA encoding [19], they tend to impose a compromise between the strength of encryption and processing speed.

Lightweight solutions such as 2D-PWLCM [21] offer speed, but may fall short in entropy or resistance to differential attacks. Some models like the 2D-SHHM [18], address image structure preservation, yet may not generalize well across all image types or sizes.

Table 1. Summary of the Literature Review

Chaotic Systems	Mathematical Representation	Key Space	Key Feature	Limitations
2D-Logistic-Sine Map [13]	$\begin{cases} x_{n+1} = \mu x_n(1-x_n) + sin(y_n) \\ y_{n+1} = \mu y_n(1-x_n) + sin(x_n) \end{cases}$	2^{79}	Enhanced randomness and security strong sensitivity to initial conditions.	Slower execution for high resolution images.
2D-SACA Map [14]	$\begin{cases} x_{n+1} = sin(arcsin(cos(y_n))) \\ y_{n+1} = cos(arcsin(sin(x_n))) \end{cases}$	2^{93}	Wider parameter space and better chaotic range.	Trajectory degeneration under certain values.
2D-IGSCM+Hill Cipher [15]	$\begin{cases} x_{n+1} = sin(\pi x_n) + e^{-x_n^2} \\ y_{n+1} = sin(\pi y_n) + e^{-y_n^2} \end{cases}$	2^{93}	High unpredictability excel length for pixel permutation and large image encryption.	Sensitive to synchronization mismatches.
2D-NLSCM [16]	$\begin{cases} x_{n+1} = \mu x_n(1-x_n) + sin(y_n) \\ y_{n+1} = sin(\pi y_n) + sin(x_n) \end{cases}$	2^{119}	Adaptive scrambling strong defense against differential attacks.	Higher computation for dynamic scrambling.

2D-Hyperchaotic Schaffer Map [17]	$\begin{cases} x_{i+1} = 0.5 + \dfrac{sin^2\left(e^a\sqrt{x_i^2 + y_i^2}\right) - 0.}{[1 + 0.001(x_i^2 + y_i^2)]^2} \\ y_{i+1} = 0.5 + \dfrac{sin^2\left(e^b\sqrt{x_{i+1}^2 + y_i^2}\right) - 0}{[1 + 0.001(x_{i+1}^2 + y_i^2)]^} \end{cases}$	2^{139}	Combines compression and encryption generates hyper chaotic output.	High resource requirements.
2D-SHM [18]	$x_{n+1}^{(1)} = \mu\, sin(\pi x_n)$ $x_{n+1}^2 = 1 - ax_n^2 + y_n$ $y_{n+1} = bx_n$	2^{64}	Good for grayscale/medical images protects local and global features.	Less effective on high-color images.
2D-SAHM+DNA [19]	$\begin{cases} x_{n+1} = ax_n^2 + y_n + 1 \\ y_{n+1} = cx_n \end{cases}$	2^{60}	Combines DNA logic with chaotic maps strong crypto graphic strength.	Encoding/decoding adds complexity.
Efficient 2D Logistic Map [20]	$\begin{cases} x_{n+1} = \mu x_n(1 - x_n) \\ y_{n+1} = \lambda y_n(1 - y_n) \end{cases}$	2^{119}	Fast and lightweight suitable for real time applications.	Less suitable for highly sensitive images.
2D-PWLCM [21]	$\begin{cases} x_{n+1} = \begin{cases} \dfrac{x_n}{p}, 0 \le x_n < p \\ \dfrac{1 - x_n}{1 - p}, p \le x_n \le 1 \end{cases} \end{cases}$	2^{93}	Simple, fast and secure for low power devices.	Reduced chaos in limited precision environments.

3. CHAOTIC PRNG ANALYSIS

In this section, a novel pseudo-random number generator is introduced and the dynamical properties of chaos are discussed.

3.1 2D Logistic Map

The classical one-dimensional logistic map's complication and chaotic behavior are boosted by the introduction of coupling between two variables in the 2D logistic map [22]. Describe the 2D logistic map with iterative equation given below.

$$\begin{cases} x_{n+1} = \mu_1 x_n(1 - x_n) + \epsilon_1 y_n \\ y_{n+1} = \mu_2 y_n(1 - y_n) + \epsilon_1 x_n \end{cases} \tag{3.1}$$

where, within the interval $(0,1)$, x_n and y_n represent the system states at n^{th} iteration. μ_2 and $\mu_1$are control parameters leading the system's nonlinearity. Interaction between the two dimensions introduce by ϵ_1 and ϵ_2 that are coupling coefficients and also enhancing the system's complexity. The coupling of x and y variables allows the system to show more complex dynamics than the 1D case. This increased complexity is helpful in cryptographic applications, mainly in image encryption, as it improves the diffusion and confusion properties crucial for secure encryption schemes

3.2 A Fermionic Matter Chaotic System

A new 2D chaotic system is proposed to generate pseudo-random numbers that can be used as encryption keys in secure image encryption schemes. P and q are the two prime-number control parameters that manage the dynamics of the proposed system and are initialized using four seed values, x_0, x_1, y_0, y_1 within the range $(0,1)$. The chaotic behavior of the system is caused by the nonlinear connection between state x and y, which is described by the following iterative equations.

$$\begin{cases} x_{n+2} = p.\left(\dfrac{x_{n+1}(2x_n + 3) + 2(y_n + x_{n+1}^2)}{\sqrt{y_n + x_{n+1}^2}}\right) mod\ 1 \\ y_{n+2} = q.\left(2\sqrt{x_n + y_{n+1}^2} + \dfrac{y_n(2y_n + 3)}{\sqrt{x_n + y_{n+1}^2}}\right) mod\ 1 \end{cases} \tag{3.2}$$

Numerous prime values of p and q have been observed to show strong chaotic characteristics in the system, and here $n = 1, 2, 3, \ldots$ etc. With a precision of 10^{-9}, numerical simulations are being performed, and further research is being carried out to examine its dynamic properties and key space expansion.

3.3 Dynamical Properties of FMCS

To evaluate the effectiveness of the proposed Fermionic Matter Chaotic System (FMCS) in comparison with the traditional Logistic map, several dynamic properties are analyzed. The assessment begins by examining the uniformity of the output sequences to determine how evenly the values are distributed. Further, the randomness of the generated sequences is tested using the NIST SP800-22 statistical suite to ensure that the pseudo-random number generator (PRNG) meets the necessary standards for cryptographic security.

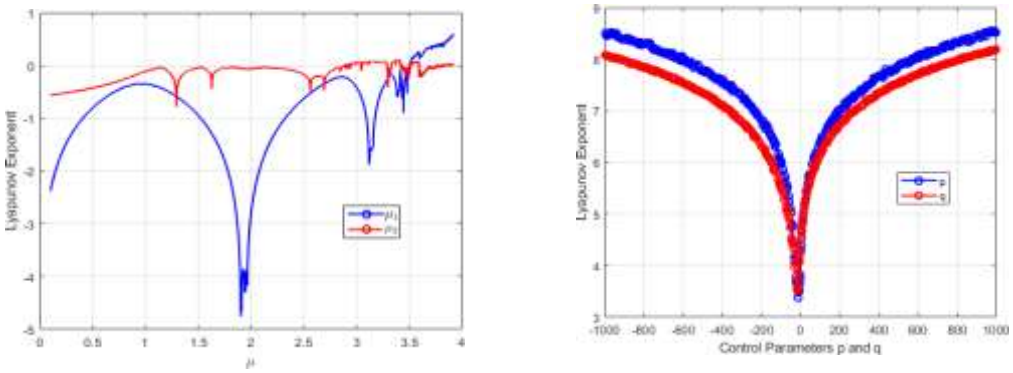
3.3.1 Lyapunov Exponent

The Lyapunov exponent, presented by mathematician Aleksandr Lyapunov in the late 19nth century, works as a vital metric in dynamical systems. It is used to calculate the sensitivity of a dynamic system to its initial conditions. An evaluation of exponential divergence or convergence along close paths reveals this sensitivity, expressed by the Lyapunov exponent. A positive Lyapunov exponent specifies that the system is chaotic and the negative exponent shows a non-chaotic system [23]. The Lyapunov exponent is expressed in equation (3),

$$\lambda = \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} ln|f'(x_i)| \tag{3.3}$$

The derivative of a variable n is written as $f'(n)$. To measure the chaotic behavior in the 2D logistic map, we calculate the Lyapunov exponent by varying the control parameters μ_1 and μ_2. First, μ_1 is varied from 0 to 4 in small increments of 0.001 while keeping μ_2 fixed, then μ_2 is varied while μ_1 remains constant. Figure 5 (a) illustrates the Lyapunov exponents got during these deviations. The system determines chaotic behavior when the Lyapunov exponent is positive and negative exponents indicate non-chaotic regions corresponding to specific intervals of $\mu_1$and μ_2. The suggested 2D system's chaotic behavior was evaluated by calculating Lyapunov exponents by varying the two control parameters p and q, which are both prime numbers. These parameters were tested across the range of 1000 to 1000, using only prime numbers within that range. The Figure 5 (b) presents the Lyapunov exponent values:

- The blue curve represents the results of p
- The red curve shows the results for q



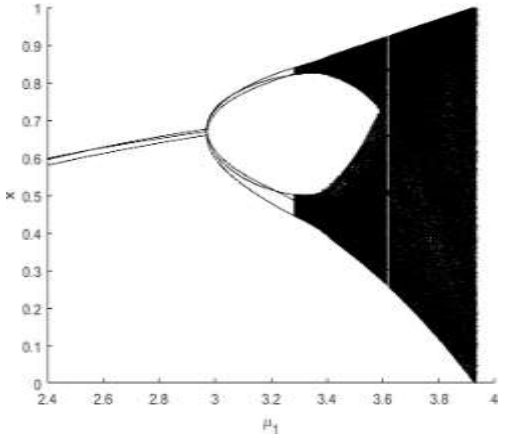
(a) Lyapunov Exponent vs μ_1 and μ_2 (b) Lyapunov Exponent vs p and q

Figure 5. Lyapunov Exponent Analysis (a) 2D Logistic map (b) FMCS using prime number-based parameters p and q.

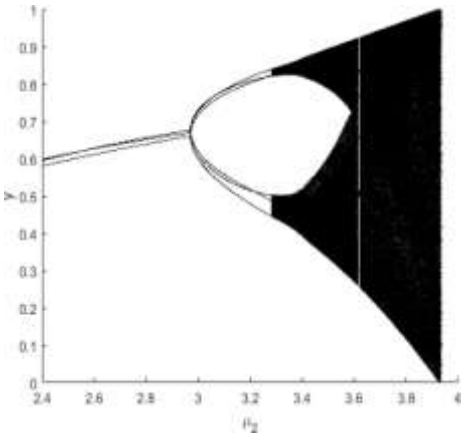
The graph shows that both curves remain consistently positive throughout the tested range, which shows the chaotic nature of the system. The lowest values appear near zero but still stay high adequate, presentation that the system keeps strong unpredictability. This shows that the suggested system offers a wide chaotic range when using large prime values for p and q, enhancing its efficiency for encryption applications that need high randomness and sensitivity to initial conditions.

3.3.2 Bifurcation Analysis

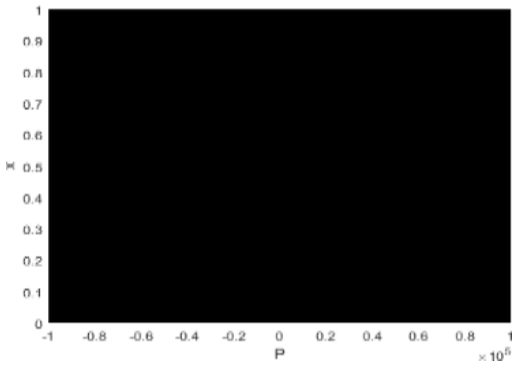
By using a bifurcation diagram, it is possible to see how a system's behavior changes when one of its control parameters is adjusted [24]. Figure 6 (a), shows the growth of the variable x_n as the parameter μ_1 increases. Initially, the system displays stable fixed points but as μ_1 increases the system goes through a series of period doubling bifurcations, finally leading to chaotic behavior where x_n varies unpredictably. Similarly, figure 6(b) shows a similar change for y_n where μ_2 is varied. These diagrams clearly show how minor changes in the control parameters can extremely alter the system's dynamics. It helps us to easily see when the system is stable and when it becomes unpredictable. The bifurcation diagram of the logistic map is shown in Figure 6 (a), (b) and Figure 6 (c), (d) displays the bifurcation diagram for the fermionic matter chaotic system.



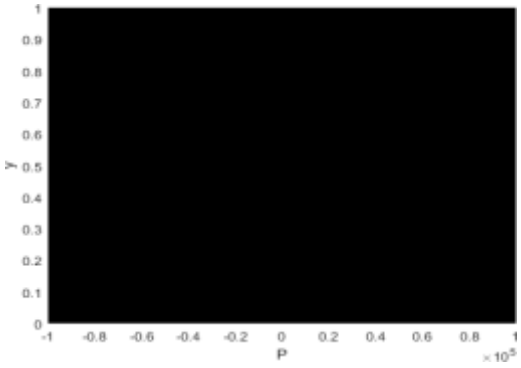
(a) μ_1 (Vary) for 2D Logistic Map



(b) μ_2 (Vary) for 2D Logistic Map



(c) x vs p for FMCS

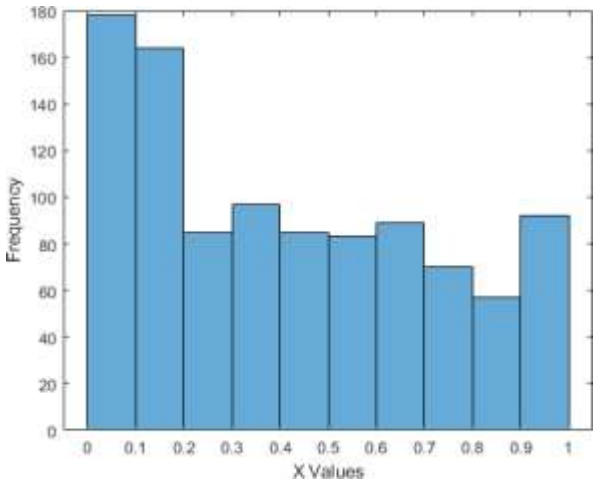


(d) y vs p for FMCS

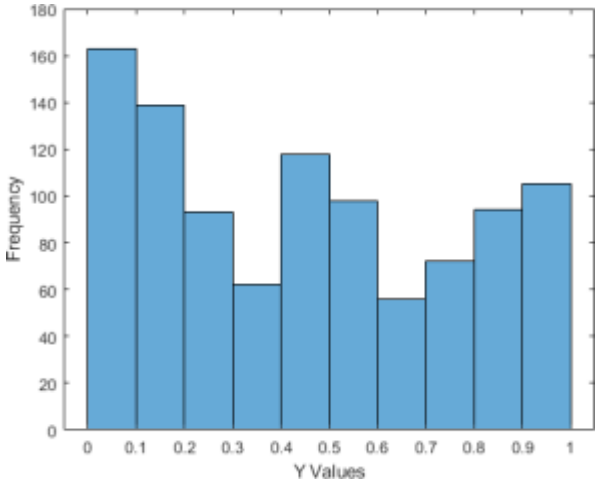
Figure 6. bifurcation analysis 2D Logistic map (a)-(b) and Fermionic Matter Chaotic System (c)-(d).

3.3.3 Uniformity Analysis

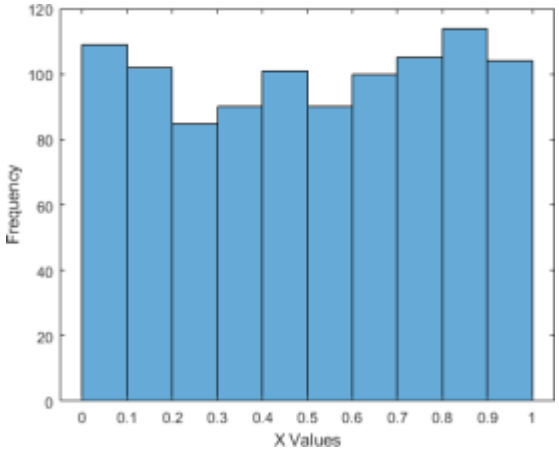
Uniformity analysis examines whether the random number generator covers the intended range. A system with uniform distribution is the least predictable dynamical system. The graphical depiction of the random numbers can be used to visually analyze the degree of homogeneity [25]. An iterative distribution graph produced from the two chaotic systems is shown in Figure 7. The distribution of the 2D Logistic map is shown in Figure 7(a) and (b), and it is not uniformly distributed. The random numbers produced by the Fermionic Matter Chaotic System (FMCS) are evenly spread throughout the range [0, 1] without any clusters, patterns, or vacations, as seen in Figure 7(c) and (d). Thus, it demonstrates that the suggested system keeps better chaotic behavior.



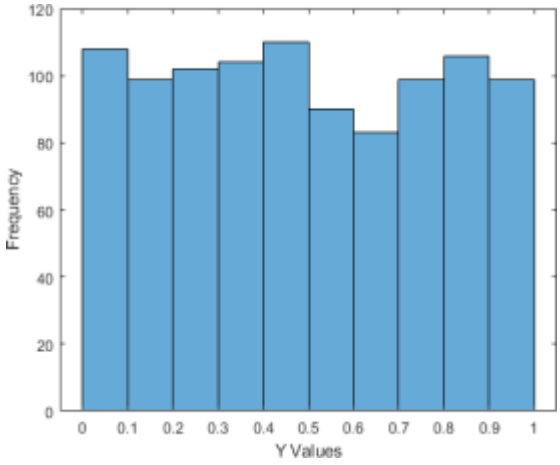
(a) x for 2D Logistic Map



(b) y for 2D Logistic Map



(c) x for FMCS

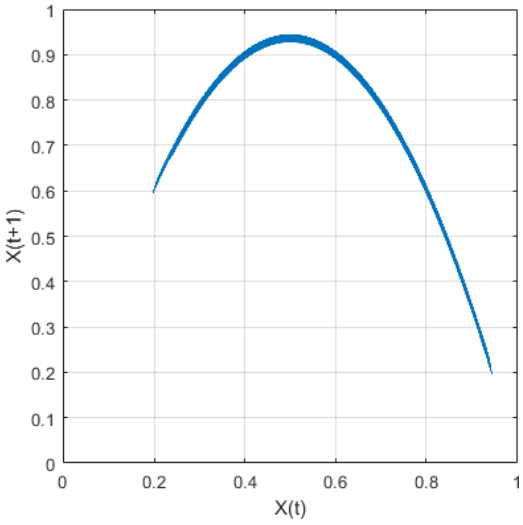


(d) y for FMCS

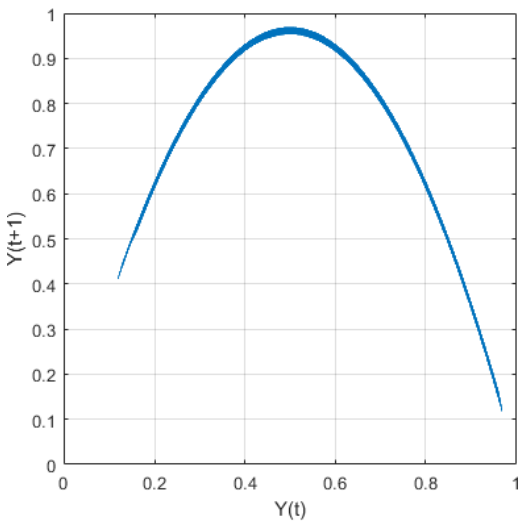
Figure 7. uniformity analysis 2D Logistic map (a)-(b) and Fermionic Matter Chaotic System (c)-(d).

3.3.4 Phase Space Analysis

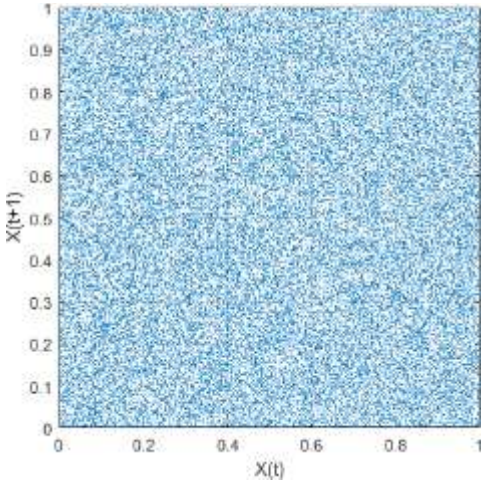
The phase space or state space of a PRNG represents all possible internal states that determine the next pseudo-random number [26]. A phase space diagram illustrates these states in a multidimensional space by plotting internal variables on different axes. Although the 2D logistic map exhibits chaotic behavior, its phase space remains relatively smooth and structured. In contrast, the proposed 2D chaotic system, employing prime parameters and modular operations, generates denser and more irregular trajectories.



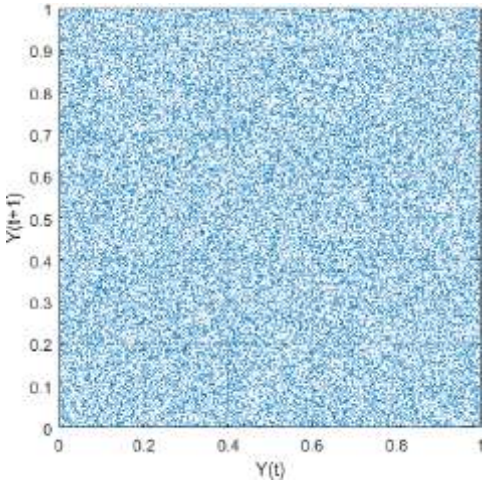
(a) x for 2D Logistic Map



(b) y for 2D Logistic Map

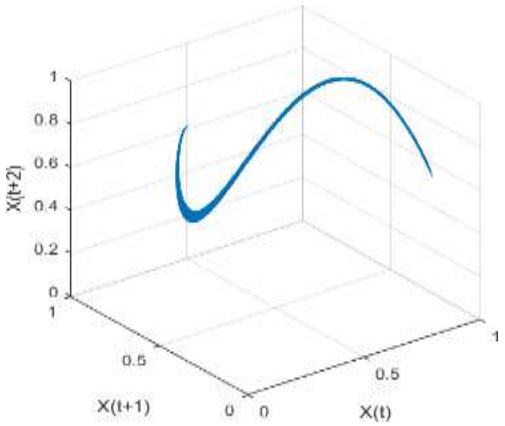


(c) x for FMCS

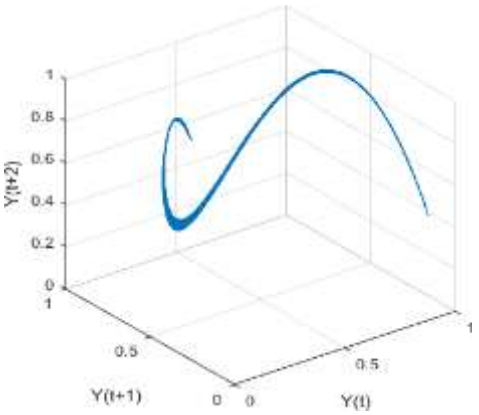


(d) y for FMCS

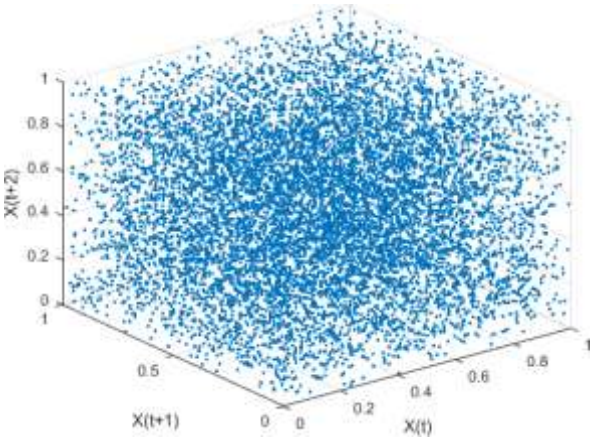
Figure 8. 2D logistic map, 2D phase space analysis (a)-(b) and FMCS (c)-(d).



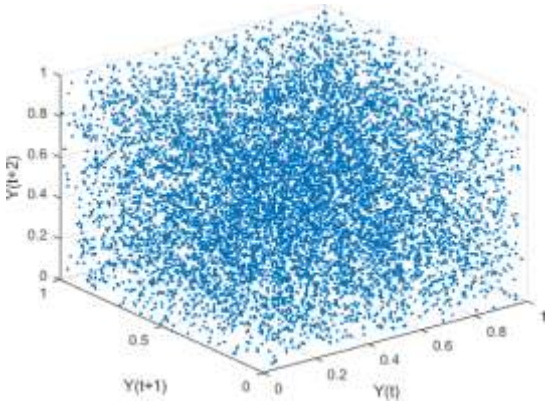
(a) x for 2D Logistic Map



(b) y for 2D Logistic Map



(c) x for FMCS



(d) y for FMCS

Figure 9. 2D Logistic map, 3D phase space analysis (a)-(b), FMCS (c)-(d)

3.3.5 Information Entropy

Lower entropy presence of structure or regularity, which can pose a security risk in cryptographic applications [29]. The formula of entropy is give below:

Entropy $= -\sum_{i=1}^{n} p_i \, log_2 \left(\frac{1}{p_i}\right)$

Here, p_i represents the probability of a specific outcome i, and the logarithm is taken to base 2.

Table 2. Information entropy results for the 2D Logistic Map

Interval	**Frequency**	**Probabilities** (P_i)	$\mathbf{Log_{10}}(P_i)$	$P_i \times \boldsymbol{Log}_{10}(P_i)$
[0.0, 0.1]	70	0.070	-1.1549	-0.08084
[0.1, 0.2]	131	0.131	-0.88273	-0.11564
[0.2, 0.3]	116	0.116	-0.93554	-0.10852
[0.3, 0.4]	108	0.108	-0.96658	-0.10439
[0.4, 0.5]	135	0.135	-0.86967	-0.11740
[0.5, 0.6]	126	0.126	-0.89963	-0.11335
[0.6, 0.7]	114	0.114	-0.94310	-0.10751
[0.7, 0.8]	84	0.084	-1.0757	-0.09036
[0.8, 0.9]	70	0.070	-1.1549	-0.08084
[0.9, 1.0]	46	0.046	-1.3372	-0.06151

Table 3. Proposed FMCS with results of Information entropy

Interval	Frequency	Probabilities (P_i)	$\mathbf{Log_{10}}(P_i)$	$P_i \times \mathbf{Log_{10}}(P_i)$
[0.0, 0.1]	93	0.10333	-0.98576	-0.10186
[0.1, 0.2]	83	0.09222	-1.03520	-0.09547
[0.2, 0.3]	81	0.09000	-1.04580	-0.09412
[0.3, 0.4]	88	0.09778	-1.00980	-0.09873
[0.4, 0.5]	80	0.08889	-1.05120	-0.09344
[0.5, 0.6]	100	0.11111	-0.95424	-0.10603
[0.6, 0.7]	103	0.11444	-0.94141	-0.10774
[0.7, 0.8]	89	0.09889	-1.00490	-0.09937
[0.8, 0.9]	86	0.09556	-1.01970	-0.09744
[0.9, 1.0]	97	0.10778	-0.96747	-0.10427

Table 4. The comparison of proposed FMCS and information entropy for 2D logistic map.

Chaotic Map	2D Logistic Map	Proposed FMCS
Information Entropy	0.98787	0.99236

3.3.6 Sample Entropy

Sample Entropy (SampEn) is a statistical measure used to determine the level of complexity and irregularity in a time series. It estimates how often similar patterns of data points continue to remain similar when one more point is added. In simpler words, it helps evaluate how predictable or random a system is. Lower SampEn values indicate regular and predictable behavior, while higher values indicate more randomness and chaotic variation [30]. Unlike Approximate Entropy, SampEn is less sensitive to data length and avoids self-matching, which makes it suitable for analyzing short or noisy datasets. These advantages make it useful for studying chaotic systems and evaluating pseudo- random number generators (PRNGs) [31].

In this work, SampEn is applied to compare the complexity of the classical 2D Logistic Map and the proposed 2D chaotic system (FMCS). The 2D logistic map is defined as,

$$\begin{cases} x_{n+1} = \mu_1 x_n(1 - x_n) + \varepsilon_1 y_n, \\ y_{n+1} = \mu_2 y_n(1 - y_n) + \varepsilon_1 x_n, \end{cases} \tag{3.4}$$

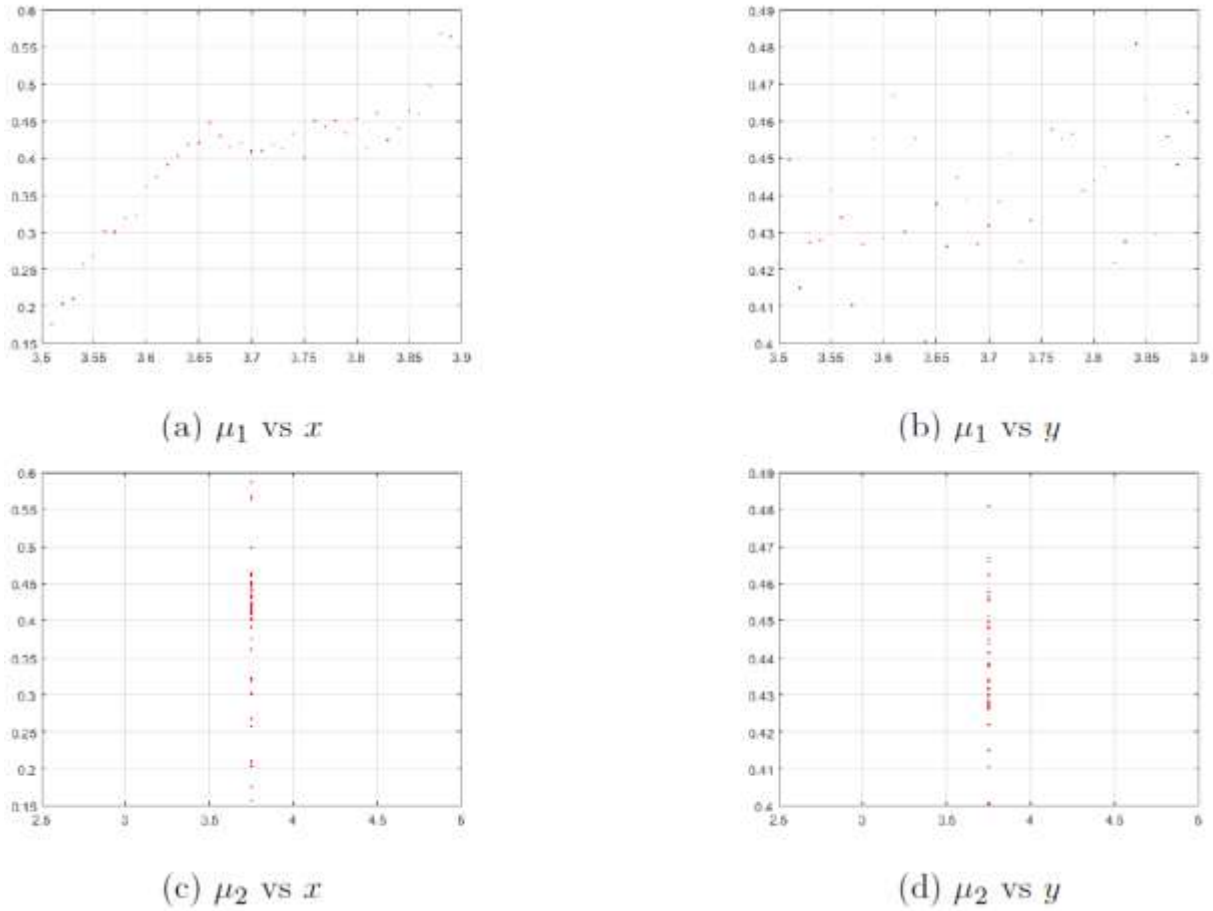
where μ_1, μ_2 are control parameters and ε_1 , ε_2 are compiling parameters.

For comparison, the proposed FMCS system uses prime numbers P and q as control parameters. Both systems were iterated over 10,000 steps to obtain sufficient data for reliable entropy estimation.

Mathematically, Sample Entropy is defined as,

$$sampEn(m, r, N) = -\ln\left(\frac{A}{B}\right), \tag{3.5}$$

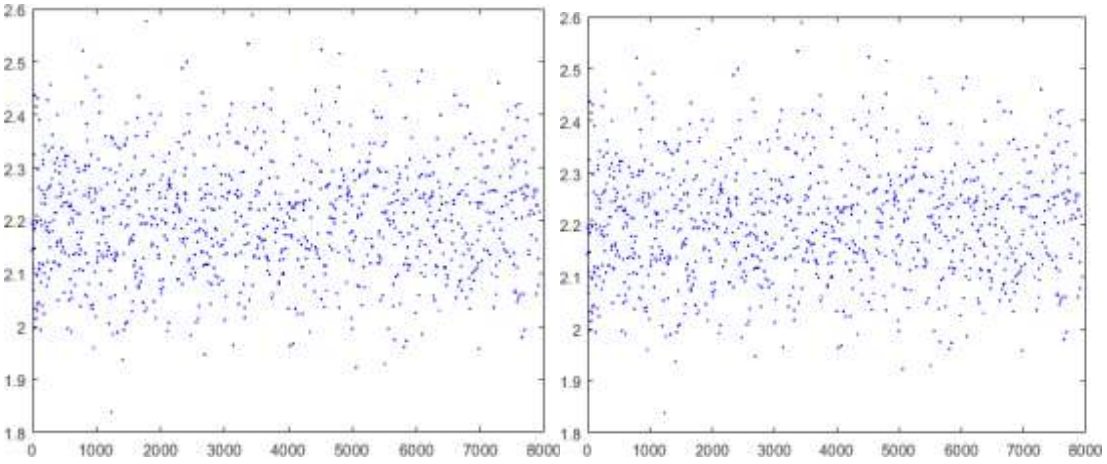
where A represents the number of matching vector pairs of length $(m + 1)$ and B the number of matching pairs of length m. The final value shows the probability that two similar sequences remain similar as their length increases. The results, as illustrated in Figure 10, for the 2D Logistic Map and Figure 11, for the proposed FMCS, show that the FMCS exhibits significantly higher Sample Entropy (SampEn) values than the classical 2D Logistic Map.



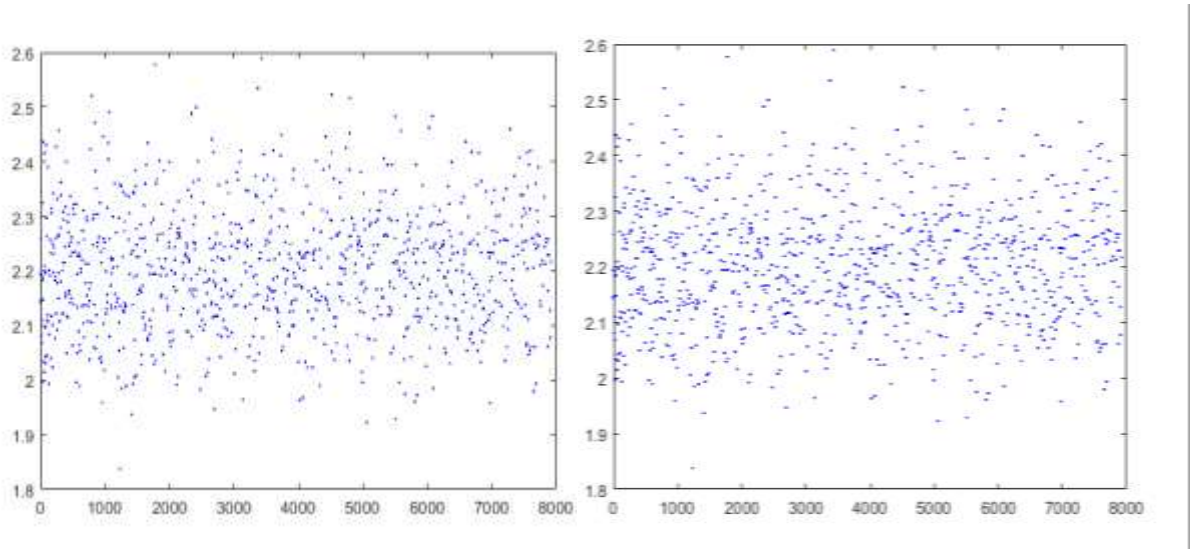
(a) μ_1 vs x (b) μ_1 vs y

(c) μ_2 vs x (d) μ_2 vs y

Figure 10, entropy sample analysis of the classical 2D logistic map across different control parameters.



(a) P vs y (b) q vs x



(c) x vs P (d) y vs q

Figure 11. Entropy analysis sample of the proposed 2D chaotic system (FMCS) under prime number control parameters.

3.3.7 Correlation Dimension Analysis

In this study, the correlation dimension is applied to compare the classical two-dimensional (2D) Logistic Map and the proposed Fermionic Matter Chaotic System (FMCS) [32]. The coupled 2D Logistic Map is used as given in equation (3.4), with the, μ_1 and μ_2 are control parameters, while ε_1 and ε_2 are coupling strengths. For the simulation, parameters were set to $\mu_1 = 3.8$, $\mu_2 = 3.75$, and $\varepsilon_1 = \varepsilon_2 = 0.02$, with initial conditions $x_0 = y_0 = 0.1$. The system was iterated for 1000 steps, and the first 100 were discarded to remove transient effects.

The proposed FMCS system, defined with prime number control parameters P and q is expressed as,

$$\begin{cases} X_{n+2} = mod\left(P\left((2X_n+3)\sqrt{Y_n + X_{n+1}^2} + 3\,ln\left(\sqrt{Y_n} + \sqrt{1+Y_{n+1}}\right)\right), 1\right), \\ Y_{n+2} = mod\left(q\left((6Y_n+3)\sqrt{X_n + Y_{n+1}^2} + 3\,ln\left(\sqrt{XX_n} + \sqrt{1+X_{n+1}}\right)\right), 1\right), \end{cases} \tag{3.6}$$

where P and q are selected from the first 1000 prime numbers to analyze the system's response to parameter variation. Initial conditions were set to $X_0 = Y_0 = 0.10000000001$, and the system was simulated for 1000 iterations. The reconstructed phase space is formed using embedding dimension $m = 2$ and time delay $\tau = 1$,

$$X_i = [x(i), x(i+\mathcal{T})] \tag{3.7}$$

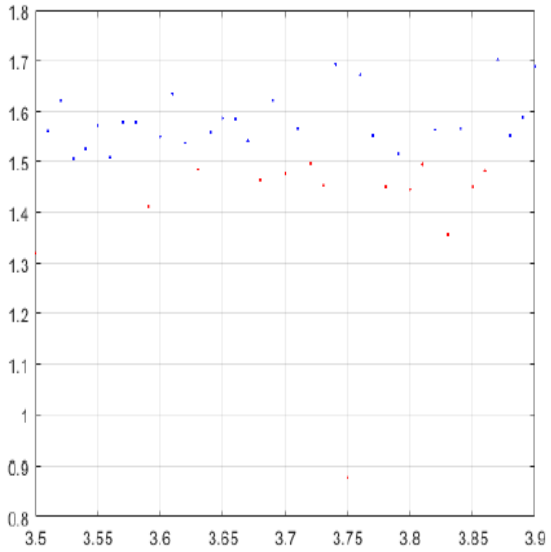
The correlation sum $C(r)$ is defined as,

$$C(r) = \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} H\left(r - \left\|X_i - X_j\right\|\right), \tag{3.8}$$

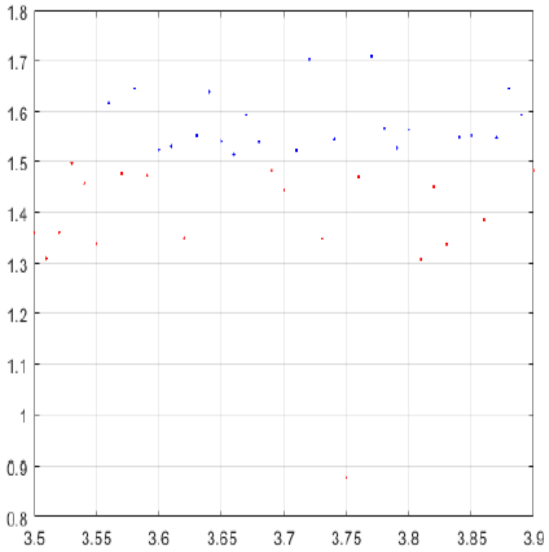
where $H(\cdot)$ is the Heaviside step function and r is the neighborhood radius. The slope of the linear region of the $log(C(r))$ versus $log(r)$ plot estimates the correla- tion dimension:

$$D_2 = \lim_{r \to 0} \frac{d\ log\ C(r)}{d\ logr}. \tag{3.9}$$

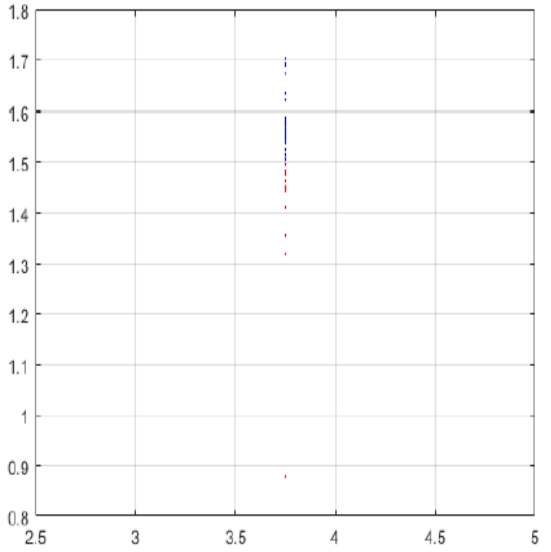
The results in Figure 12, show that FMCS consistently achieves a higher correlation dimension (D_2) than the 2D Logistic Map.



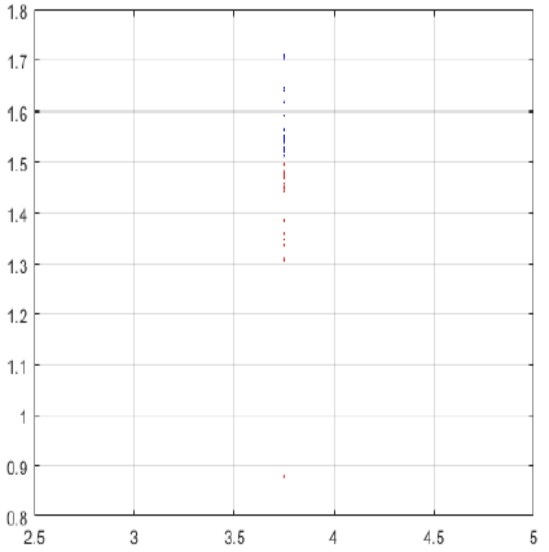
(a) μ_1 vs x



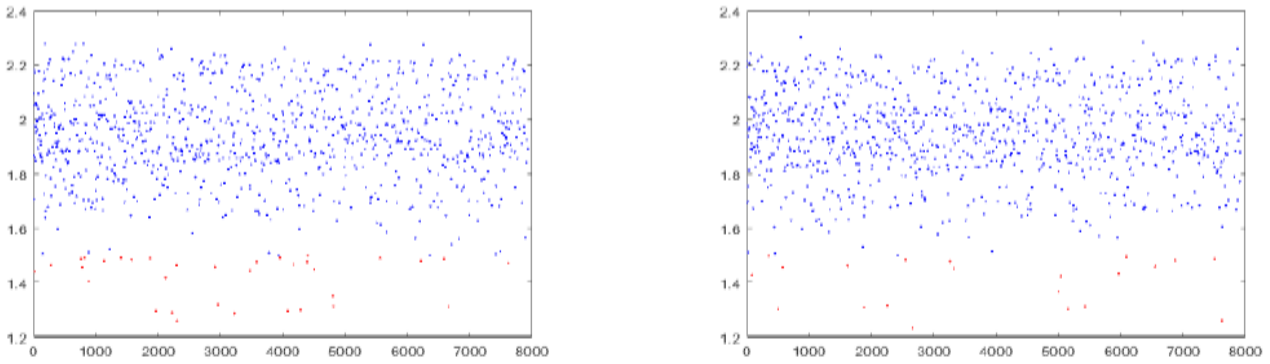
(b) μ_1 vs y



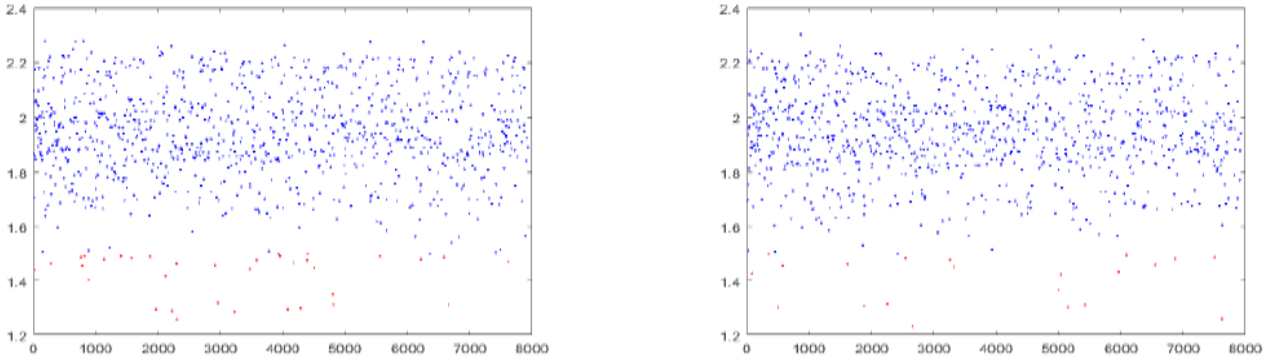
(c) μ_2 vs x



(d) μ_2 vs y



(e) P vs x (f) P vs y

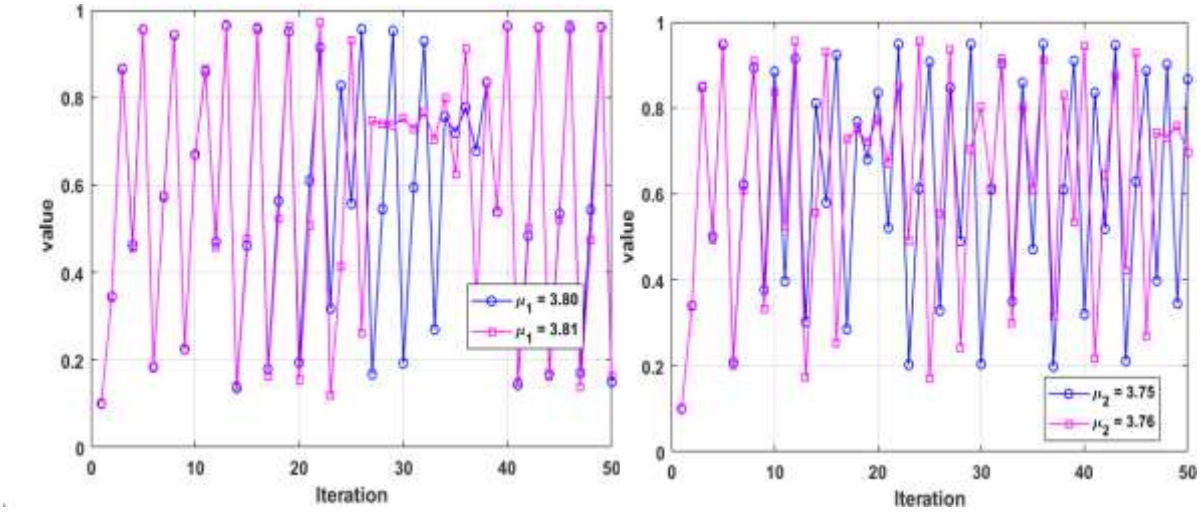


(g) q vs x (h) q vs y

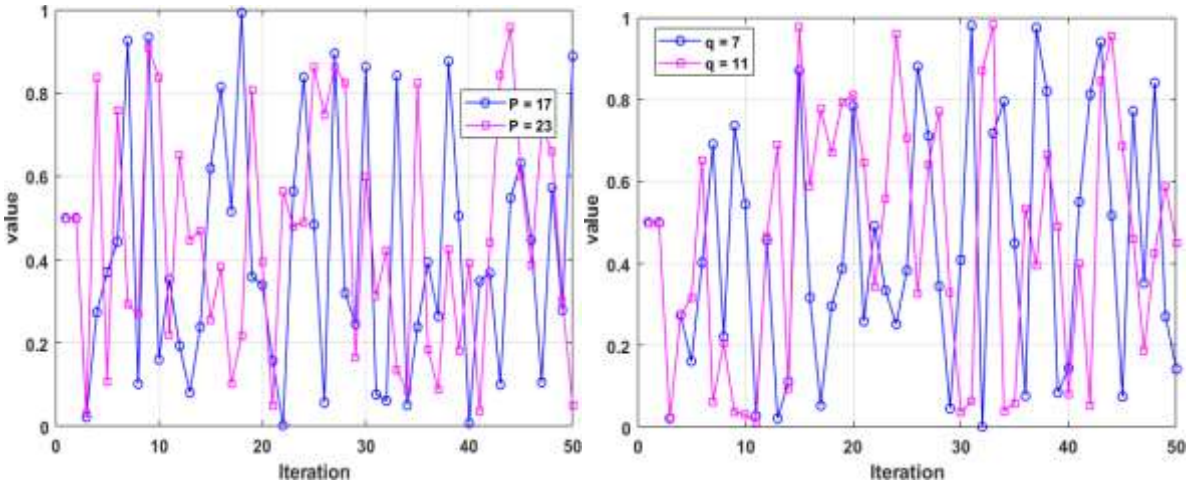
Figure 12. Analysis correlation dimension (e-h) FMCS showing higher chaotic complexity and nonlinear dynamics (a-d) logistic map.

3.3.8 Key Sensitivity Analysis

Indicating, how a slight variation in control parameters or secret keys can lead to a completely different chaotic sequence and key sensitivity is a critical property of secure cryptosystems.



(a) (b)



(c) (d)

Figure 13. key sensitivity analysis: (a-b) for 2D Logistic Map and (c-d) for FMCS.

3.4 NIST SP800-22 Statistical Test Suite

The National Institute of Standards and Technology is an organization in the Department of Commerce in the U.S. In advance, the measurement standards and encouragement are recognized for framework related to cybersecurity. The National Institute of Standards and Technology (NIST SP800-22) test suite is a set of fifteen statistical tests that are used to validate if the series of numbers RNG is random or not [33]. The tests are categorized in the NIST SP800-22 suite into four main clusters: Runs, block, frequency and miscellaneous tests. To evaluate the result of randomness assessments mostly tests use a p- value. For the testing a bit of sequences provided, the p-value is considered statistical metrics derived from that series. A general beginning for the p-value is set at 0.01 as if p 0.01, so the is considered to have failed. If all the tests pass the sequence within the suite, then it is appropriate to consider the cryptographic application. Some specific tests comprising the NIST SP800-22 suite are discussed below as,

(1) Frequency Test (2) Frequency Test within a Block (3) Runs Test (4) Tests for the Longest-Run-of-Ones in a Block (5) Binary Matrix Rank Test (6) Discrete Fourier Transform Test (7) Non-overlapping Template Matching Test (8) Overlapping Template Matching Test (9) Maurer's Universal Statistical Test (10) Linear Complexity Test (11) Serial Test (12) Approximate Entropy Test (13) Cumulative Sums Test (14) Random Excursions Test (15) Random Excursions Variant Test.

3.4.1 Frequency test

The test is correlated to the number of zeros and the ones in the sequence are known as Monbiot. Its purpose is to evaluate whether the sequence contains approximately equal numbers of ones and zeros, as would be expected from a truly random sequence [33]. The smooth equation (6) is utilized to calculate test statistics,

$$x^2 = \frac{(n_0 - n_1)^2}{n} \tag{3.10}$$

where the x^2 is the test statistic, n is the total number of bits in the sequence, n_0 is the count of zeros in the sequence and n_1 is the count of ones in the series.

3.4.2 Frequency test within a block

The Frequency Test within a Block is a statistical test used to measure the random- ness of a bit sequence. To qualify for the test, the percentage of ones and zeroes in each block must be determined and compared to the expected percentage [33]. The equation (7) is as follows:

$$x^2 = \frac{(b_0 - b_1)^2}{b} \tag{3.11}$$

where x^2 signifies test statistic. Total number of bits is b in the sequence, in a block b_0 is number of zeros and b_1 represents number of ones in a block.

3.4.3 Runs test

The run test is a statistical test used to determine whether a series of data exhibits randomness or a pattern [33]. The mathematical equation (8) for the run test is as follows,

$$v = \frac{2n_1n_2}{n_1 + n_2} + 1 \tag{3.12}$$

Test statistic represents v, in the sequence n is the total number of bits, n_1 is a number of runs of ones and n_2 is a number of runs of zeros. To determine the randomness this test is very useful and used for ensuring that RNG does not produced extended runs of bits (either zeros or ones) that reason of randomness.

3.4.4 Tests for the Longest-Run-of-Ones in a Block

To check the longest runs in a series of blocks, this test is used. The test statistic for the longest run of ones in a block is determined by using the equation (9). Where v signifies the test statistic for the longest run of ones in a block, $l_1, l_2, l_3, \dots, l_m$ are the lengths of the longest runs of ones in each block, and m is the total number of blocks [33].

$$v = max(l_1, l_2, \dots, l_m) \tag{3.13}$$

3.4.5 Binary Matrix Rank Test

To check the randomness of the bits by adapting them to a binary matrix and then the rank is considered. In the statistics test, w (10) is calculated on the foundation of the ranks and metrics. To observe test statistics laterally with the predicted rank and the standard deviation of the rank, facilitate the calculation of the p-value [33]. This p-value is evaluated against to assess whether a bit of series can be categorized as random.

$$w = r_1 + r_2 + \cdots + r_k \tag{3.14}$$

3.4.6 Discrete Fourier Transform Test

The level of randomness in a bit of a series is evaluated by the Discrete Fourier Transform (DFT). To recognize any non-random patterns within the input sequence, the discrete Fourier transform is first applied. Once the frequency spectrum is generated, it must be assessed. Generally, the test statistics can be defined based on the characteristics being analyzed. Let y denote the values of the binary sequence and y' represent the corresponding Discrete Fourier Transform values [33]. The test statistic T can be defined by the equation (11).

$$t = f(y_0, y_1, .., y_{n-1}) \tag{3.15}$$

3.4.7 Non-overlapping Template Matching

Test In the series, to check the instances of predetermined patterns, this test is used to across non-overlapping portions [33]. The equation (12) is well-defined for the statistics test, such as,

$$t = \sum_{i=0}^{12} \left(n_i - \frac{n}{2}\right)^2 \tag{3.16}$$

The statics test is t. In the sequence, m is the number of blocks that do not overlap, n is the number of times the template appears in the i^{th} block, and n is the whole number of times the template appears during the complete sequence.

3.4.8 Overlapping Template Matching Test

The Overlapping Template Matching Test is similar to nonoverlapping, though it too acknowledges overlapping patterns in the sequence [33]. This equation is used for this test.

$$\chi^2 = \sum \frac{(o_i - e_i)^2}{e_i} \tag{3.17}$$

The statistical chi squared test is χ^2, the observed count of occurrences for the i^{th} pattern is o_i and e_i is the predictable number of amounts for the ith pattern supposing randomness. The sum is taken from all over the templates.

3.4.9 Maurer's Universal Statistical Test

This test identifies the number of bits among the matching templates of the series to treasure whether the data and information are lost or not if the sequence can be compressed [33]. Length of the compressed sequence

$$c = \frac{Length\ of\ the\ compressed\ sequence}{Length\ of\ the\ original\ sequence} \tag{3.18}$$

3.4.10 The Linear Complexity Test

This test the length of the Linear Shift Back Register to determine the difficulty of the sequence. The shortest length is essential to duplicate a binary sequence to calculate the liner complexity of the series that can be used [33]. Mathematically, can be expressed as,

$$l(s) = min\{m|\ s\ is\ a\ generated\ by\ an\ m\} \tag{3.19}$$

3.4.11 Serial Test

To check the frequency of all the expected overlapping M-bit patterns in the series, the Serial test is used. In calculating the frequencies, the test involves all the possible M-bit patterns and applying statistical analysis [33]. To compute the following equation,

$$\chi^2 = \frac{(o_i - e_i)^2}{e_i} \tag{3.20}$$

where o_i signifies the experiential frequency of pattern i while e_i is the possible frequency of pattern i.

3.4.12 Approximate Entropy Test

To check the frequency of intersecting sub-strings of successive lengths in a random series, the Approximate Entropy Test is applied. The test is grounded on the assessed entropy, which measures to check the randomness in a sequence. In finding the non-random template in the data this test is very helpful [33]. To calculate this, the following equation,

$$ae(m,n) = \varphi_m(n) - \varphi_{m+1}(n) \tag{3.21}$$

3.4.13 Cumulative Sums Test

The Cumulative Sums Test chooses whether the cumulative number of partial sequences happening in the tested series is too large or small, with significance to the cumulative behavior of random series [33]. The test statistics for the Cumulative Sums Test are determined using the formulae,

Forward cumulative sums,

$$s_i^+ = max(0, s_{i-1}^+ + x_i - 0.5) \tag{3.22}$$

Reverse cumulative sums,

$$s_i^- = max(0, s_{i-1}^- + x_i + 0.5) \tag{3.23}$$

The element of i in the series is represented by s_i, s_i^+ signifies the cumulative sum of the onward sequence at the position s_i, s_i^- and stands for the cumulative sum of the reverse sequence at position i. This test statistic is determined by taking the maximum total value between these cumulative sums [33].

$$s = max(max(s_i^+), max(s_i^-)) \tag{3.24}$$

3.4.14 Random Excursion Test

The Random Excursion test sums the total number of excursions or regressions to the mean in the cumulative total of a binary sequence to recognize any deviations from randomness. This test is useful for discovering trends or patterns that may point to non-random behavior [33].

3.4.15 Random Excursions Variant Test

The Random Excursions Variant Test finds how many times a state happens in the series in a cumulative overall. It provides further understandings of the behavior of the sequence to check the randomness [33].

$$x = \max_i(|S_i|) \tag{3.25}$$

where S_i signifies cumulative sum of the series up to the i^{th} position. The maxi mum total value of s_i of the all positions is determined. The correlation coefficient is calculated as follows,

$$r = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum(X_i - \bar{X})^2 \sum(Y_i - \bar{Y})^2}} \tag{3.26}$$

Table 5. NIST Test Results

Test	2D Logistic Map		Proposed FMCS	
	P -Value	Proportion	P -Value	Proportion
Frequency	0.1238	5.493	0.6519	9.99
Block Frequency	0.2478	9.719	0.4885	9.91
Cumulative Sums	0.1316	5.617	0.6293	9.92
Runs	0.1029	5.127	0.5025	9.91
Longest Run	0.2468	8.305	0.4931	9.92
Rank	0.4094	9.425	0.4547	9.9
FFT	0.3463	8.699	0.4752	9.86
Non-Overlapping Template	0.4597	9.868	0.9826	10
Overlapping Template	0.4399	9.672	0.2865	10
Approximate Entropy	0.0096	5.176	0.6487	9.48
Serial	0.0109	4.221	0.4792	9.98
Linear Complexity	0.4925	9.755	0.4792	9.77
Random-excursions	0.0020	9.0376	0.1369	10
Random-excursions Variant	9.9900	9.755	0.0951	10
Universal	0.0254	0.00	0.0254	9.9

4 IMAGE ENCRYPTION ALGORITHM BASED ON FMCS

In this section we present image encryption algorithm based on FMCS.

4.1 Image Encryption Algorithm

The proposed image encryption algorithm known as the Pendulum-Wave Intelligent Mapping with Deterministic Probing (PWIM- DP). The method employs the newly developed Fermionic Matter Chaotic System (FMCS) as the source of pseudo-random sequences to achieve strong pixel permutation and diffusion. The algorithm ensures high sensitivity, deterministic reversibility, and complete key dependence between encryption and decryption.

The PWIM-DP algorithm introduces a deterministic, collision-free image encryption method that integrates nonlinear chaotic sequences, pendulum-like pixel motion, and XOR-based diffusion. Unlike frequency-domain schemes, PWIM-DP operates purely in the spatial domain while maintaining perfect reversibility and uniform randomness.

The core of the algorithm relies on

- Chaotic sequences generated from the FMCS
- Pendulum-inspired continuous mapping for pixel position prediction
- Deterministic open-address probing to resolve pixel collisions
- XOR-based diffusion with feedback chaining to enhance confusion.

4.2 Chaotic Sequence Generation (FMCS Model)

The FMCS is a two-dimensional chaotic system governed by the following nonlinear equations,

$$X_{n+2} = p\left(2x_n\sqrt{y_n - x_{n+1}^2} - \frac{(2x_n+3)x_{n+1}^2}{\sqrt{y_n - x^2 - x_{n+1}}}\right) mod\ 1 \tag{4.1}$$

$$Y_{n+2} = q\left(6y_n\sqrt{x_n - y_{n+1}^2} + \frac{(6y_n+3)y_{n+1}^2}{\sqrt{x_n + y_{n+1}^2}}\right) s\ mod\ 1 \tag{4.2}$$

where p and q are prime control parameters, and (x_0, x_1, y_0, y_1) are secret initial keys. These sequences (X, Y) exhibit ergodicity and high sensitivity, forming the backbone of the encryption process.

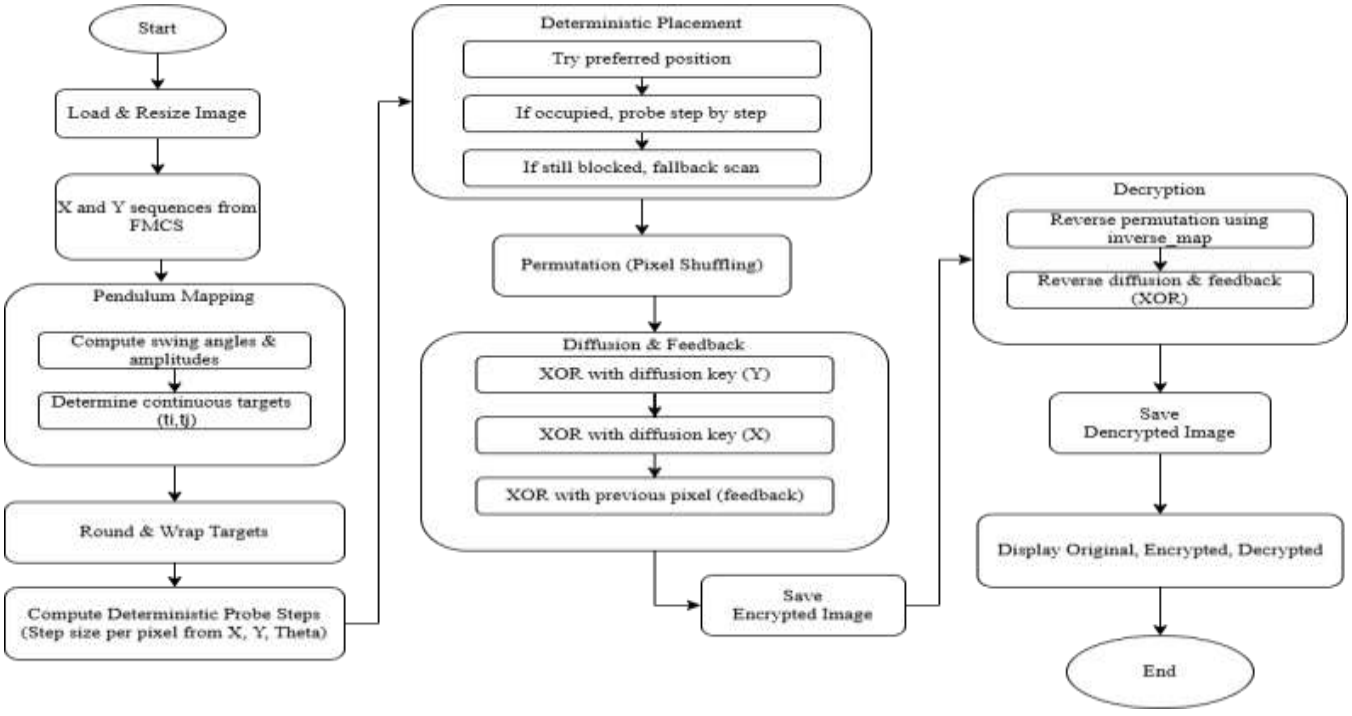


Figure 14. Flowchart of the Proposed PWIM-DP Encryption and Decryption Process

4.3 Encryption process

The encryption process in PWIM-DP is divided into distinct logical phases from pixel mapping to diffusion.

4.3.1 Step-by-Step Encryption

Step 1. Load Image The input color image $I(x, y)$ is read and resized to a fixed dimension $M \times N$.

Step 2. Generate FMCS chaotic sequences chaotic sequences X and Y are generated using the FMCS model, each of length $M \times N$.

Step 3. Pendulum mapping (continuous targets) Each chaotic pair (x_n, y_n) determines a swing angle $\Theta = sin(2\pi X_n)$ and amplitude $a = base\_amp + 10y_n$. These are used to compute continuous pixel targets (t_i, t_j).

Step 4. Rounding and Wrapping The continuous coordinates (t_i, t_j) are rounded and clamped within valid pixel bounds $[1, M\ ]$ and $[1, N\ ]$.

Step 5. Deterministic probing and placement each pixel's preferred position is assigned using a deterministic open-addressing method, if a collision occurs (i.e., two pixels want the same slot), the algorithm probes step-by-step using a pendulum-derived offset until an empty slot is found.

This produces two mappings.

- forward map: old → new positions
- inverse map: new → old positions

Step 6. Pixel permutation

The image pixels are rearranged according to the forward mapping, producing the permuted image I_p.

Step 7. Diffusion and Feedback (XOR Mixing) To enhance security, two key matrices K and C are derived from Y and X, respectively. Each permuted pixel is diffused using XOR operations.

$$E(i,j) = I_p(i,j) \oplus K(i,j) \oplus C(i,j) \oplus E(i,j-1)$$

where E is the encrypted (cipher) image.

4.4 Decryption Process

The decryption process is the exact inverse of encryption and fully deterministic.

Step 1. Regenerate FMCS Sequences

The same initial keys (x_0, x_1, y_0, y_1) and parameters (p, q) are used to recreate X and Y.

Step 2. Reverse Diffusion and Feedback

Using the stored or recomputed inverse mapping, pixel is returned to their original spatial positions.

$$I_{dec}(old) = I_p(new)$$

The result is the perfectly recovered decrypted image I_{dec} identical to the original.

4.5 Key Characteristics of PWIM-DP

High key sensitivity tiny variations in initial conditions (x_0, y_0) lead to entirely different chaotic sequences, ensuring strong key sensitivity. Deterministic and collision-free mapping each pixel achieves a unique destination due to deterministic open addressing and fallback probing, guaranteeing one-to-one mapping. Strong confusion and diffusion, the combination of pendulum mapping (permutation) and XOR feedback (diffusion) achieves high randomness and pixel decorrelation. Perfect reversibility, all encryption operations are mathematically invertible, allowing lossless decryption when correct keys are used.

5 SECURITY ANALYSIS AND EXPERIMENTAL RESULTS

In this section, we present experimental results of the proposed FMCS-based image encryption system introduced in the previous chapter. A secure encryption system must resist various attacks, including differential, statistical, brute-force, chosen plain-image, and chosen cipher-image attacks. To evaluate the performance and reliability of the proposed method, several stochastic and deterministic tests have been performed.

5.3 Simulation Result

The standard grayscale images Goat, Cat, and Rabbit were resized to 256×256 pixels. The proposed FMCS (Fermionic Matter Chaotic System) image encryption algorithm was implemented and evaluated using MATLAB R2021. The chaotic matrix is derived from a user-defined secret key, which is hashed using the SHA- 1 algorithm. The encryption process is key-dependent, highly sensitive to small changes in initial seeds and control parameters, ensuring strong security and complexity. The FMCS chaotic sequences are generated based on the initial seeds and control parameters, x_0, x_1, y_0, y_1, p, q, with a precision of 10^9 for X and Y , and prime control parameters p and q. Figures 15, 16, 17 and 18 decryption and encryption are successful with the correct secret key.

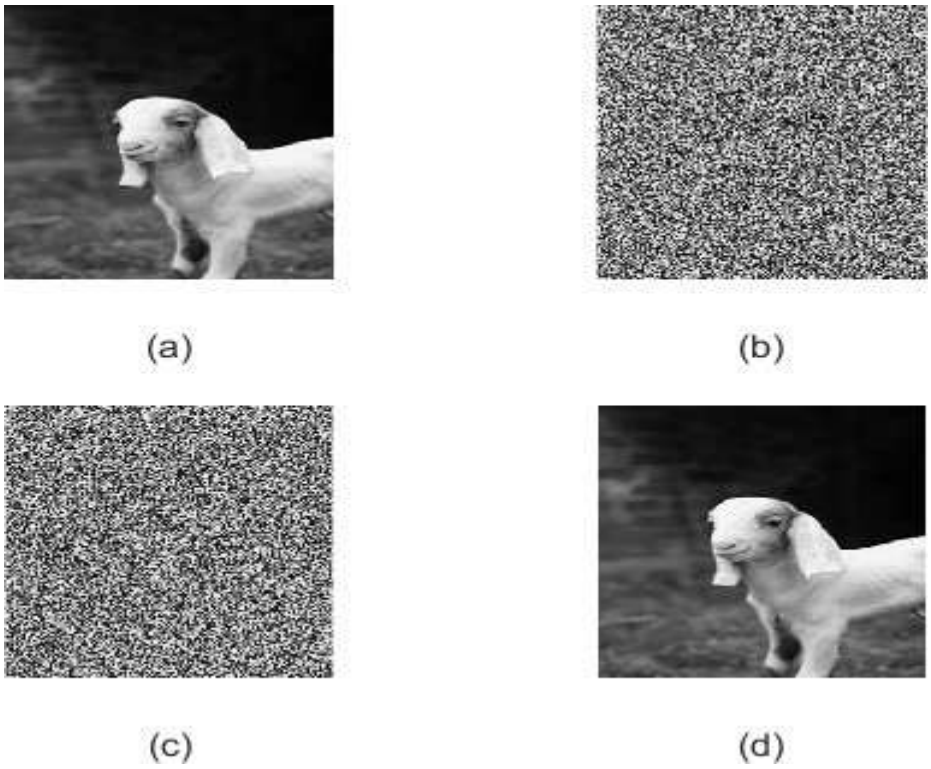


Figure 15. Encryption and decryption results for Goat image: (a) Original, (b) Encrypted, (c) Encrypted (same), (d) Decrypted using correct key.

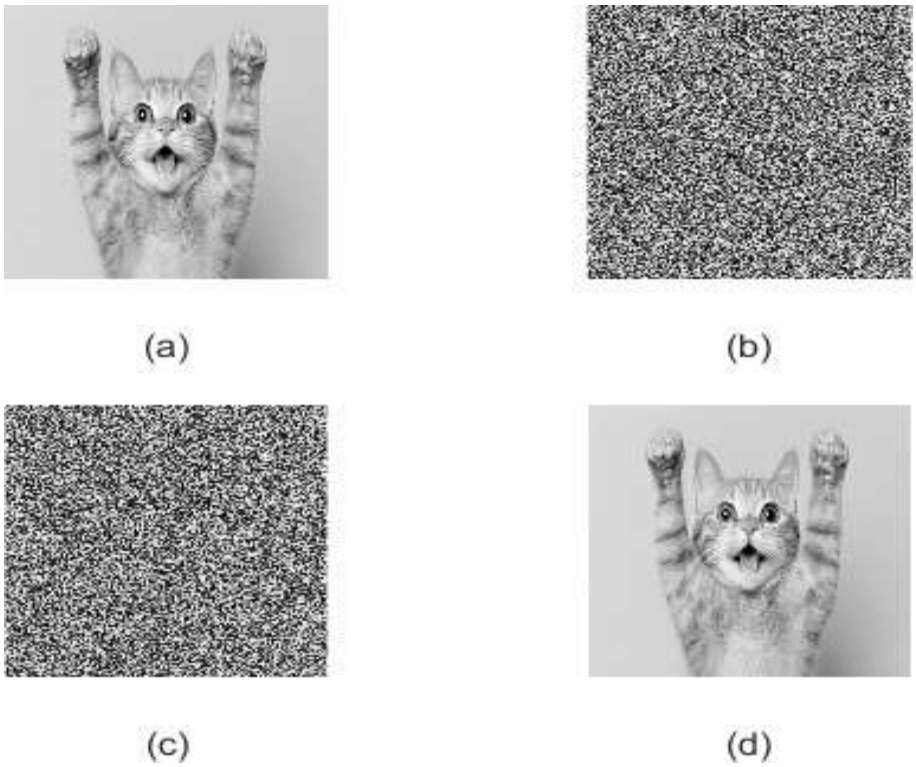


Figure 16. Encryption and decryption results for Cat image: (e) Original, (f) Encrypted, (g) Encrypted (same), (h) Decrypted.

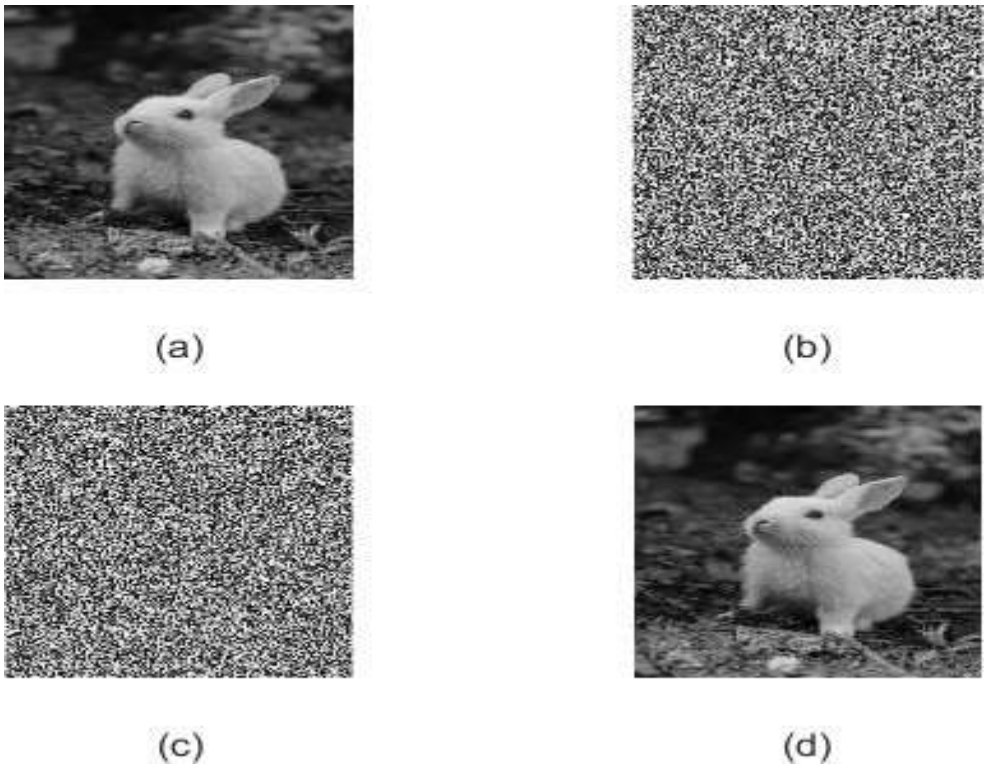


Figure 17. Encryption and decryption results for Rabbit image: (a) Original, (b) Encrypted, (c) Encrypted (same), (d) Decrypted.

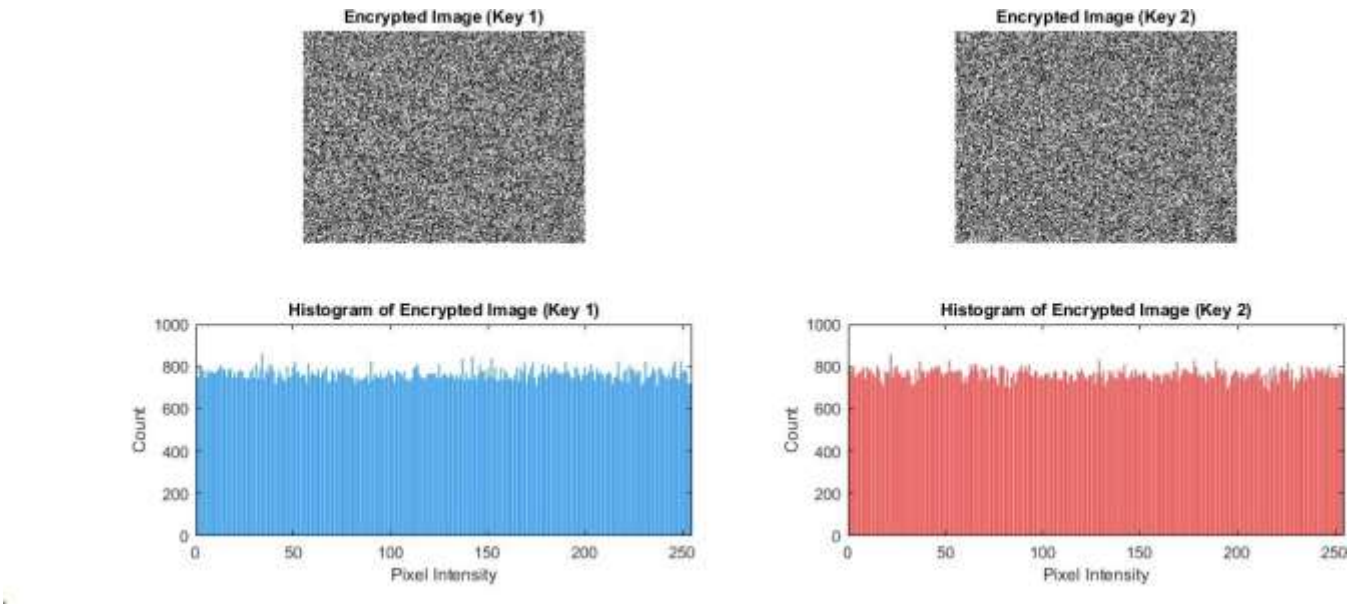


Figure 18. Encrypted image comparison using two slightly different FMCS keys for Rabbit image, (a) Key 1 with seeds $x_0 = 0.7231, x_1 = 0.2345, y_0 = 0.5432, y_1 = 0.9876, p = 3, q = 5$; (b) Key 2 with seeds $x_0 = 0.7231000001, x_1 = 0.2345000001, y_0 = 0.5432000001, y_1 = 0.9876000001, p = 3, q = 5$ (c) Histogram of Key 1 (d) Histogram of Key 2.

5.4 Key Space Analysis

The key space of a cryptographic system refers to the total number of possible keys that can be used. It directly measures the system's resistance to brute-force attacks, where an attacker attempts every possible key until the correct one is found. In the proposed FMCS-based encryption system, the secret keys consist of six parameters, $x_0, x_1, y_0, y_1, p, and\ q$. With a precision of 10^{-14} for X and Y , and prime numbers for p and q, the theoretical key space without any hash function is approximately 1084. This means there are 10^{84} unique possible combinations of the secret key parameters specifically, the initial seeds x_0, x_1, y_0, y_1 and control parameters p, q. When a SHA-1 hash is applied as a key-derivation function, the key space expands dramatically to approximately 10^{132}. This corresponds to about 438.98 bits of entropy, providing extremely high resistance to brute-force attacks. To put these numbers into perspective.

Magnitude. 1084 is a 1 followed by 84 zeros, while 10^{132} is a 1 followed by 132 zeros. Both numbers are astronomically large; the estimated number of atoms in the observable universe is only about 10^{80}.

Brute-force infeasibility. Even with extremely fast modern computers capable of testing billions of keys per second, it would take longer than the age of the universe to exhaustively search these key spaces, making brute-force attacks practically impossible.

Contribution of Precision. The key space depends on the parameter precisions. High precision ensures a very large number of distinguishable keys, which increases exponentially.

Chaotic Sensitivity. Due to the chaotic nature of FMCS, even tiny changes in the keys result in completely different encryption outputs. This guarantees that attackers cannot approximate the key or predict the sequence. Overall, the FMCS key space, especially when enhanced with SHA-1, provides a very strong foundation for secure image encryption.

5.5 Key Sensitivity Analysis

Determining the impact of small changes to the encryption key still requires examining the robustness of the algorithm's core sensitivity features. During this phase of the analysis, technical parameters like unified average changing intensity (UACI) and number of pixels change rate (NPCR) prove that even minor changes to the secret key yield disparate images post-encryption.

5.5.7 Differential Analysis

The described comparative assessment locks in as the most relevant one out of all possible methods in regards to studying the security and robustness of the image encryption techniques. Using this approach, the authors opt for two relatively popular NPCR (Number of Pixels Change Rate) and UACI (Unified Average Changing Intensity) [34]. NPCR is defined as the ratio of the number of pixels that change value between two encrypted images to the total number of pixels in the image [35]. Its formula is,

$$NPCR(E_1, E_2) = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} D(i,j)}{N \times M} \times 100\% \tag{5.1}$$

where

$$D(i,j) = \begin{cases} 1, if\ E_1(i,j) \neq E_2(i,j) \\ 0, if\ E_1(i,j) = E_2(i,j) \end{cases}$$

Here, N and M represent the height and width of the image, and (i,j) are the pixel coordinated of E_1 and E_2.

The $UACI$ metric quantifies the average intensity difference between two images,

$$UACI = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} |C_1(i,j) - C_2(i,j)|}{N \times M} \times 100\% \tag{5.2}$$

Table 6. Original image sensitivity analysis using NPCR test

Image	Proposed FMCS
Goat	99.6002
Cat	99.6175
Rabbit	99.6028

Table 7. Show NPCR P-values, score and test result for Goat, Cat and Rabbit images.

Image	NPCR P-Value	NPCR Score	Result
Goat	0.97071	99.6002	Pass
Cat	0.85246	99.5982	Pass
Rabbit	0.78690	99.5931	Pass

Table 8. UACI test results computed for Goat, Cat, and Rabbit images with test result

Image	UACI P-Value	UACI Score	Result
Goat	0.20978	35.9718	Pass
Cat	0.66153	33.1783	Pass

Rabbit	0.92277	33.4004	Pass

Table 9. UACI test results computed for same images with different keys

Image	Proposed FMCS
Goat	33.504004
Cat	33.520185
Rabbit	33.499931

Table 10. NPCR and UACI results for encrypted and decrypted images using different keys

Image	NPCR Score	UACI Score
Goat	99.6104	36.0243
Cat	99.6033	33.2038
Rabbit	99.6104	33.2966

5.6 Statistical Analysis

The statistical features of the algorithm are studied with the help of the histogram analysis as well as the correlation coefficient.

5.6.7 Histogram Analysis

Histogram analysis provides an intuitive method to evaluate how effectively an encryption algorithm conceals the statistical characteristics of an image. By comparing the histograms of the original and encrypted images, as illustrated in Figure 19, it is evident that the proposed FMCS-based encryption produces a nearly uniform intensity distribution. The histograms of the original Cat, Goat and Rabbit images exhibit noticeable peaks and the natural pixel correlations present in unencrypted images.

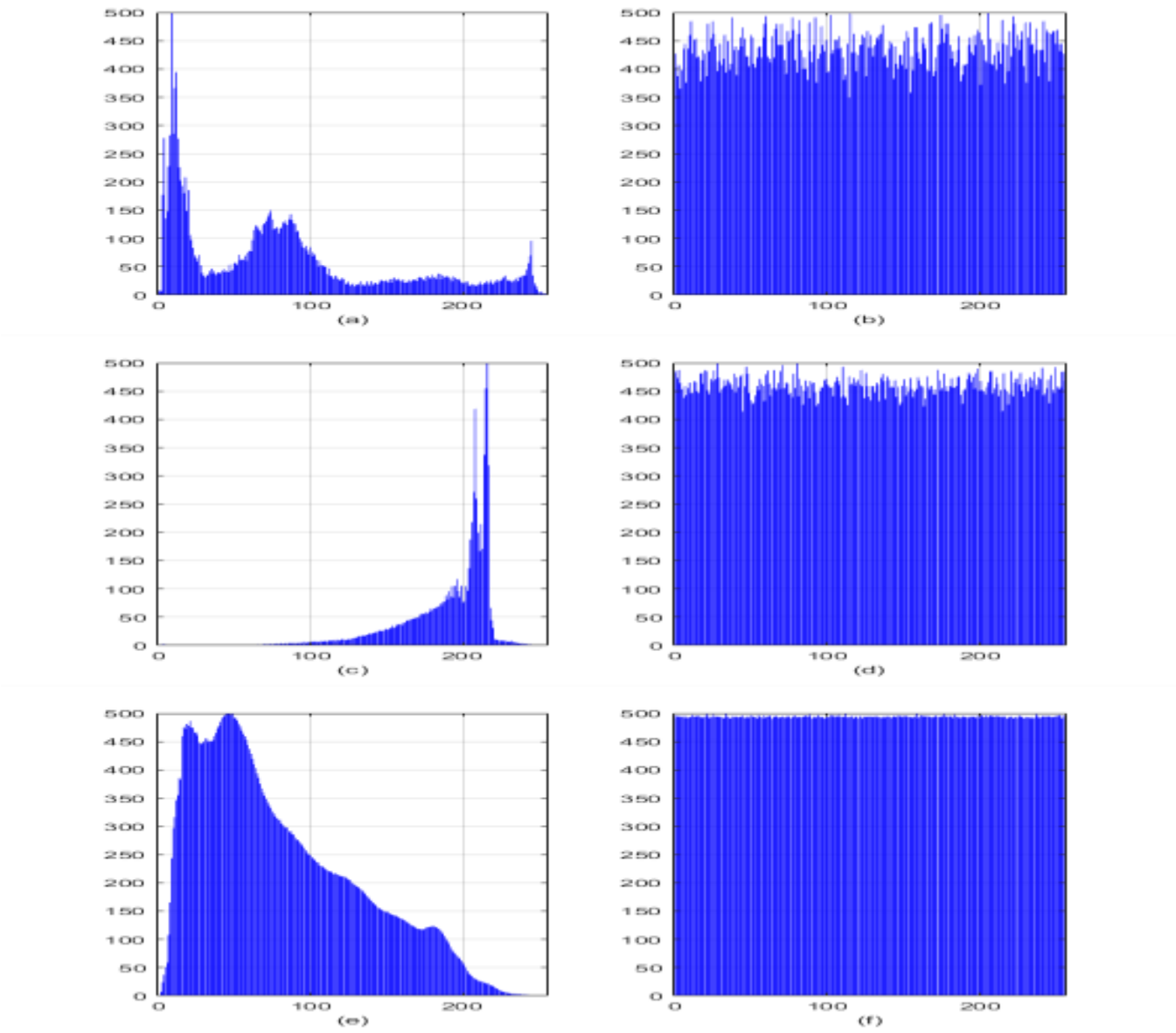


Figure 19. (a) Original Goat Histogram (b) Encrypted Goat Histogram (c) Original Goat Histogram (d) Encrypted Goat Histogram (e) Original Histogram of Rabbit, (f) Rabbit Encrypted Histogram.

5.6.8 Correlation Analysis

Correlation analysis plays a vital role in evaluating the performance of image encryption algorithms. It measures how strongly the pixel values of an image are related to their neighboring pixels. In plain (unencrypted) images, adjacent pixels whether in the horizontal, vertical, or diagonal direction usually have high correlation because natural images contain smooth regions and gradual intensity changes. However, an effective encryption algorithm should break this relation- ship, resulting in encrypted images with near-zero or negative correlation. This ensures higher resistance to statistical and differential attacks [36]. To evaluate the proposed FMCS-based encryption scheme, correlation coefficients were calculated using 3,000 randomly selected pixel pairs from the standard grayscale test images: Goat, Cat, and Rabbit (each of size 256 × 256). The analysis was carried out in three main directions: horizontal, vertical, and diagonal.

The correlation coefficient γ is defined as,

where,

$$D(x) = \frac{1}{M}\sum_{m=1}^{M}(x_m - \bar{x})^2 \tag{5.4}$$

$$Cov(x,y) = \frac{1}{M}\sum_{m=1}^{M}(x_m - \bar{x})(y_m - \bar{y}) \tag{5.5}$$

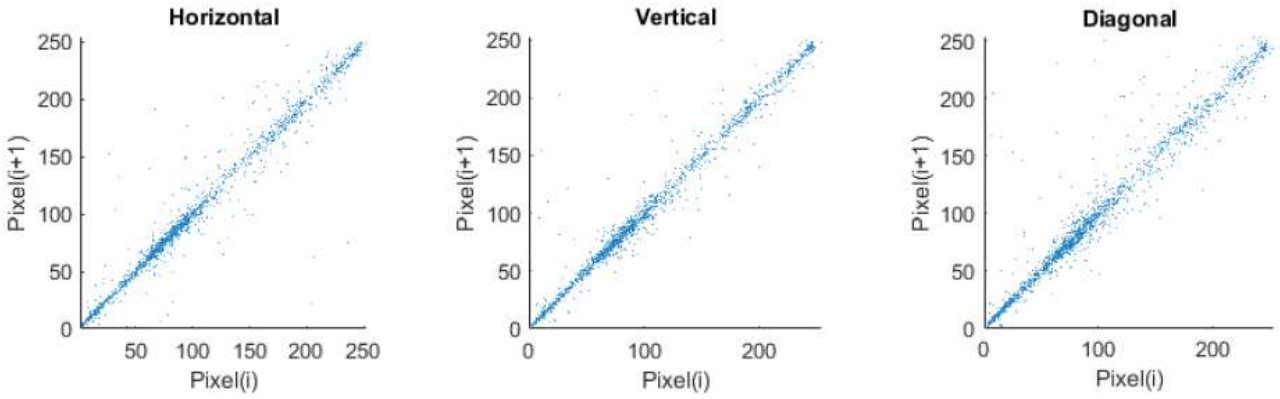
Here, $\bar{x}$ and $\bar{y}$ shoe the mean values of pixel sets y and x and M consider the total number of pixel pairs.

While the encrypted images exhibit values close to zero in all directions, Table 11, show that he results summarized and the plain images have high correlation coefficients

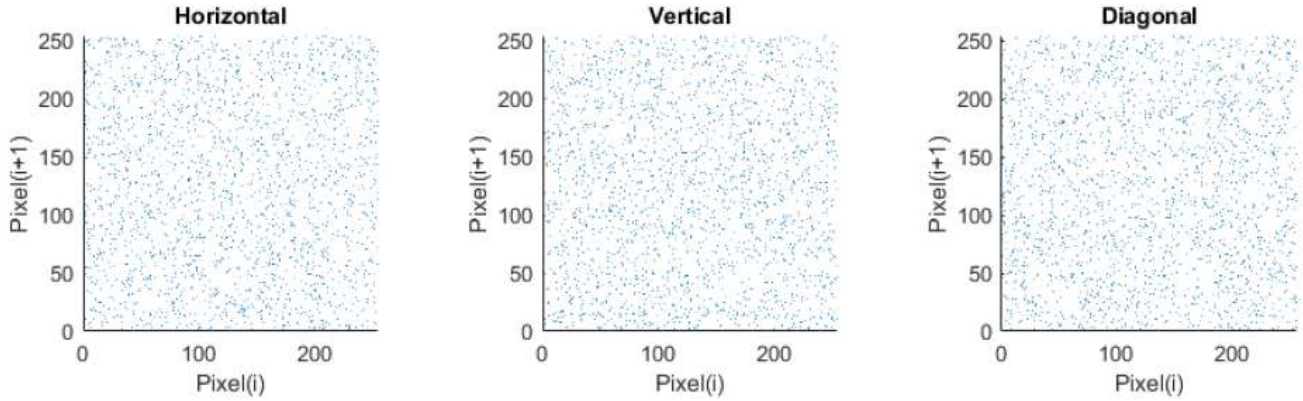
Table 11. Show the correlation for Cat, Goat and Rabbit images.

Image	Direction	Plain Image	Encrypted Image (FMCS)
Goat	Horizontal	0.9877	-0.0038
	Vertical	0.9915	0.0230
	Diagonal	0.9853	-0.0213
Cat	Horizontal	0.9467	0.0386
	Vertical	0.9430	0.0056
	Diagonal	0.9292	0.0239
Rabbit	Horizontal	0.9822	-0.0089
	Vertical	0.9812	-0.0075
	Diagonal	0.9726	0.0228

Figures 20, 21, and 22 show the graphical results in further illustrate this difference.

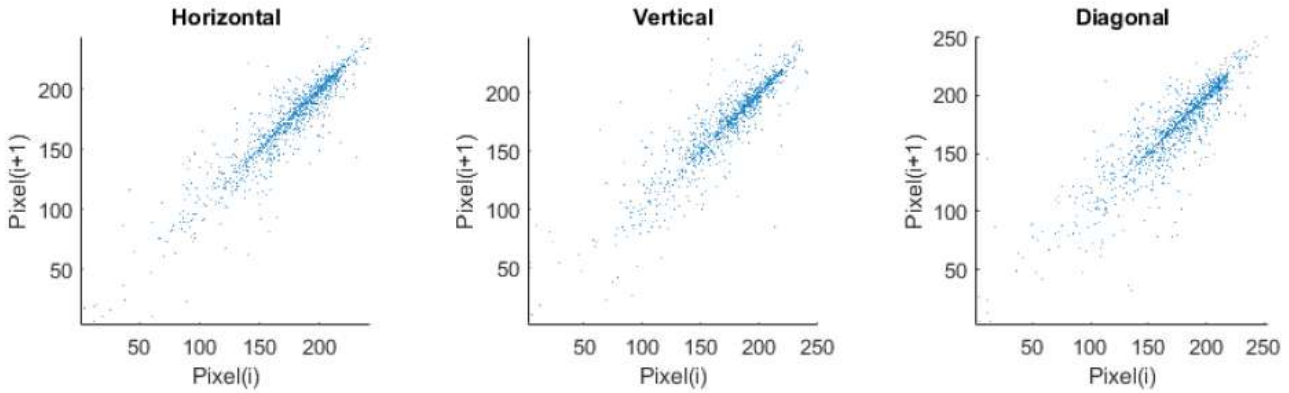


(a) Original Image

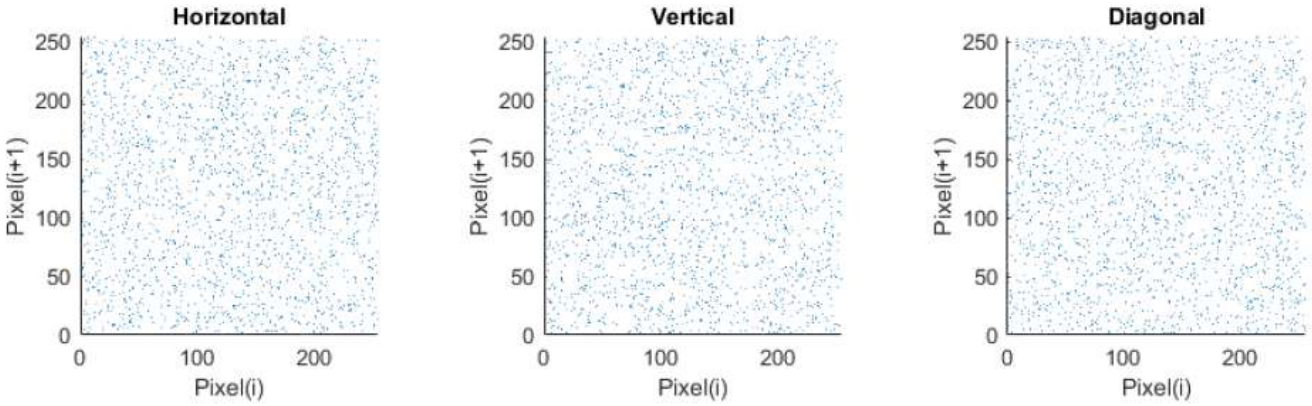


(b) Encrypted Image

Figure 20. Correlation of Goat image in vertical, horizontal and diagonal directions.

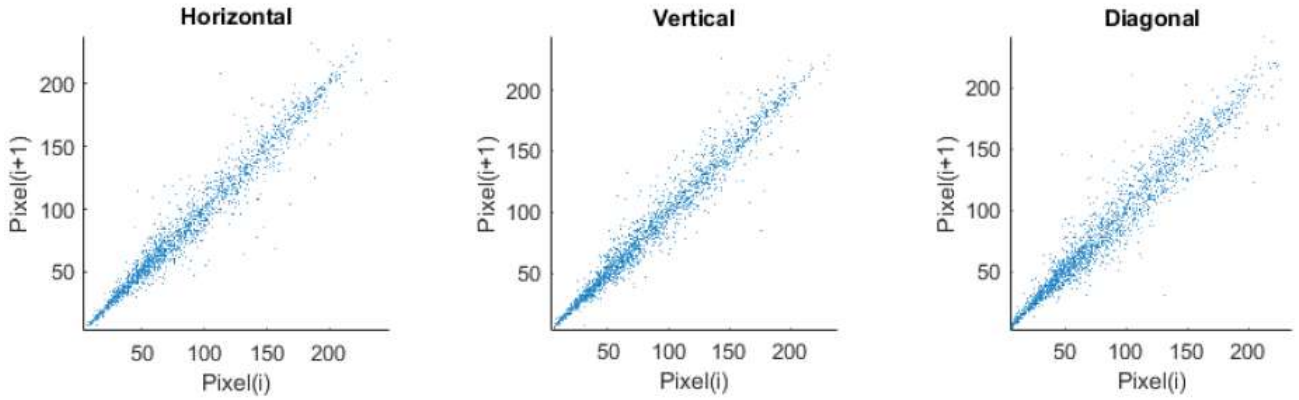


(a) Original Image

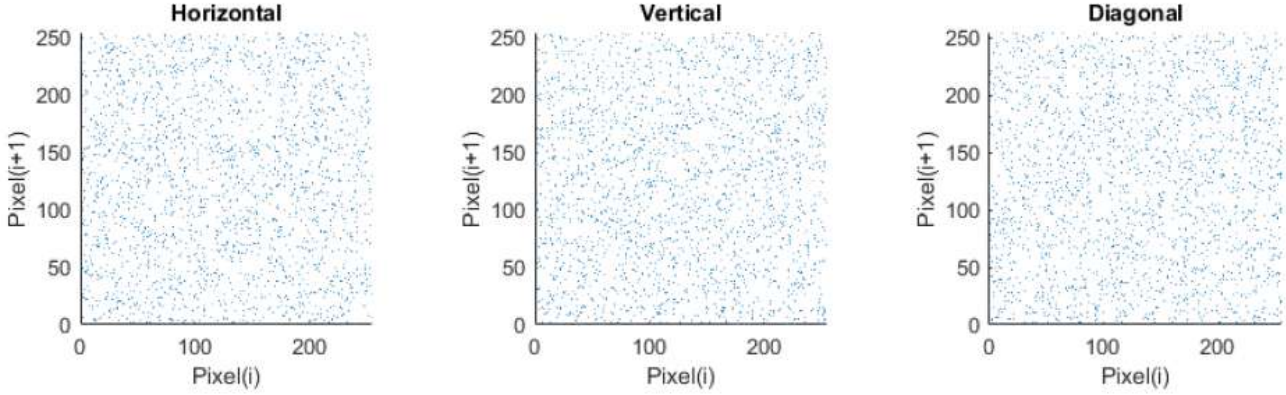


(b) Encrypted Image

Figure 21. Correlation of Cat image in vertical, horizontal and diagonal directions.



(a) Original Image



(b) Encrypted Image

Figure 22. Correlation analyze of Rabbit image in vertical, horizontal and diagonal directions.

5.7 Information Entropy

In image encryption, to check how well the algorithm hides the details and patterns of the image [37]. Entropy tells how evenly the pixel values are spread [38].

Mathematically, entropy is given by,

$$H = -\sum_{i=0}^{N-1} p_i \, log_2(p_i), \tag{5.6}$$

where, N = total number of gray levels (256 for 8-bit images), H = information entropy (in bits), p_i = probability of each pixel value. This happens when all pixel values (0–255) occur equally often, meaning the image is completely random. In image encryption, a value close to 8 means the encryption is very strong because the pixels are well distributed and un predictable [39]. A lower entropy value means the image still has some visible or statistical structure, which can make it less secure.

Table 12. Information entropy analysis for test images

Image	Plain Image	Cipher Image
Goat	7.4112	7.9972

Cat	6.3021	7.9973
Rabbit	7.4677	7.9972

6 CONCLUSION

In this research, an image encryption method was developed using Fermionic matter chaotic system. The system showed strong sensitivity to initial conditions. Even small change in the initial values completely change the whole output can cause. This kind of behavior is very good for security because it makes the output hard to predict for the hacker. SHA-1 hashing was used for key generation. The concept was simple even if the starting values change a little bit, the resulting key should become totally different. And that's exactly the FMCS reacts if small variations occur in the initial valves. During testing the results were good. The histogram of the ciphertext pictures was nearly flattened that is, the distribution of the valve of the pixels was even. The correlation of pixel was near zero and this implies that pixels were randomized and not according to any pattern. The entropy remained near to 8 that indicates a high level of randomness. Outputs of NPCR and UACI revealed that a minor fluctuation in the input could lead to massive fluctuation of the encrypted output. To conclude, the FMCS is secure and quick. It may be applied to secure sharing of images, Cloud Storage and a number of multimedia applications.

Data Availability Statement: No data set is used in this study.

Declaration Statement: The authors declare no conflicts of interest.

REFERENCES

[1] A. A. Yazdeen, S. R. M. Zeebaree, M. M. Sadeeq, S. F. Kak, O. M. Ahmed, and R. R. Zebari, FPGA implementations for data encryption and decryption via concurrent and parallel computation: A review, Qubahan Academic Journal, vol. 1, no. 2, pp. 8–16, 2021.

[2] A. M. Abdullah et al., Advanced encryption standard (AES) algorithm to encrypt and decrypt data, Cryptography and Network Security, vol. 16, no. 1, p. 11, 2017.

[3] J. Kapoor and D. Thakur, Analysis of symmetric and asymmetric key algorithms, in ICT Analysis and Applications, pp. 133–143, 2022.

[4] S. A. Wadho, A. F. Meghji, A. Yichiet, R. Kumar, and F. B. Shaikh, Encryption Techniques and Algorithms to Combat Cybersecurity Attacks: A Review, VAWKUM Transactions on Computer Sciences, vol. 11, no. 1, pp. 295–305, 2023.

[5] M. N. Alenezi, H. Alabdulrazzaq, and N. Q. Mohammad, Symmetric encryption algorithms: Review and evaluation study, International Journal of Communication Networks and Information Security, vol. 12, no. 2, pp. 256–272, 2020.

[6] M. S. A. Mohamad, R. Din, and J. I. Ahmad, Research trends review on RSA scheme of asymmetric cryptography techniques, Bulletin of Electrical Engineering and Informatics, vol. 10, no. 1, pp. 487–492, 2021. [7] Y. Yan, The overview of elliptic curve cryptography (ECC), Journal of Physics: Conference Series, vol. 2386, no. 1, p. 012019, 2022.

[8] M. S. A. Mohamad, R. Din, and J. I. Ahmad, Research trends review on RSA scheme of asymmetric cryptography techniques, Bulletin of Electrical Engineering and Informatics, vol. 10, no. 1, pp. 487–492, 2021. [9] G. J. Simmons, Symmetric and asymmetric encryption, ACM Computing Surveys, vol. 11, no. 4, pp. 305–330, 1979.

[10] H. Arora, G. K. Soni, R. K. Kushwaha, and P. Prasoon, Digital image security based on the hybrid model of image hiding and encryption, in Proc. 2021 6th Int. Conf. Communication and Electronics Systems (ICCES), pp. 1153–1157, 2021.

[11] D. Blackman and S. Vigna, Scrambled linear pseudorandom number generators, ACM Transactions on Mathematical Software (TOMS), vol. 47, no. 4, pp. 1–32, 2021.

[12] M. Zhao and H. Liu, Construction of a nondegenerate 2D chaotic map with application to irreversible parallel key expansion algorithm, Inter- national Journal of Bifurcation and Chaos, vol. 32, no. 06, p. 2250081, 2022.

[13] A. H. Alrubaie, M. A. A. Khodher, and A. T. Abdulameer, Image encryption based on 2DNA encoding and chaotic 2D logistic map, Journal of Engineering and Applied Science, vol. 70, no. 1, p. 60, 2023.

[14] M. Zhao, L. Li, and Z. Yuan, An image encryption approach based on a novel two-dimensional chaotic system, Nonlinear Dynamics, vol. 112, no. 22, pp. 20483–20509, 2024.

[15] Y. Zheng, Q. Huang, S. Cai, X. Xiong, and L. Huang, Image encryption based on novel Hill Cipher variant and 2D-IGSCM hyper-chaotic map, Nonlinear Dynamics, vol. 113, no. 3, pp. 2811–2829, 2025.

[16] Z. Zhang, J. Tang, H. Ni, and T. Huang, Image adaptive encryption algorithm using a novel 2D chaotic system, Nonlinear Dynamics, vol. 111, no. 11, pp. 10629–10652, 2023.

[17] U. Erkan, A. Toktas, and Q. Lai, 2D hyperchaotic system based on Schaffer function for image encryption, Expert Systems with Applications, vol. 213, p. 119076, 2023.

[18] Q. Lai and Y. Liu, A meaningful image encryption method based on dynamic update pixel diffusion and 2D hyperchaotic map, Nonlinear Dynamics, vol. 112, no. 16, pp. 14527–14546, 2024. [19] L. Huang, Y. Ye, and Y. Liu, Image encryption based on 2D-SAHM chaotic system and a novel DNA operation rule, The European Physical Journal Special Topics, vol. 233, no. 6, pp. 1311–1330, 2024.
[20] Y. Wu, S. Chu, H. Bao, D. Wang, and J. Zhou, Efficient Image Encryption via 2D Logistic Chaos Mapping: Strengthening Security with Pixel-Level Dynamics, International Arab Journal of Information Technology, vol. 21, no. 5, pp. 915–924, 2024.
[21] A. Elghandour, A. Salah, and A. Karawia, A new cryptographic algorithm via a two-dimensional chaotic map, Ain Shams Engineering Journal, vol. 13, no. 1, p. 101489, 2022.
[22] V. Kumar and A. Girdhar, A 2D logistic map and Lorenz-Rossler chaotic system based RGB image encryption approach, Multimedia Tools and Applications, vol. 80, no. 3, pp. 3749–3773, 2021.
[23] A. Chan, A. De Luca, and J. T. Chalker, Spectral Lyapunov exponents in chaotic and localized many-body quantum systems, Physical Review Research, vol. 3, no. 2, p. 023118, 2021.
[24] T. Bonny, S. Vaidyanathan, A. Sambas, K. Benkouider, W. Al Nassan, and O. Naqaweh, Multistability and bifurcation analysis of a novel 3D jerk system: Electronic circuit design, FPGA implementation, and image cryptography scheme, IEEE Access, vol. 11, pp. 78584–78600, 2023.
[25] Y. Jeon, S. Baek, H. Kim, G. Kim, and J. Kim, Differential uniformity and linearity of S-boxes by multiplicative complexity, Cryptography and Communications, pp. 1–26, 2022.
[26] A. Chan, M. Khalil, K. A. Shahriar, D. V. Plant, and L. R. Chen, Encryption in phase space for classical coherent optical communications, Scientific Reports, vol. 13, no. 1, p. 12965, 2023.
[27] D. A. Rusakov, A misadventure of the correlation coefficient, Trends in Neurosciences, vol. 46, no. 2, pp. 94–96, 2023.
[28] S. Chatterjee, A new coefficient of correlation, Journal of the American Statistical Association, vol. 116, no. 536, pp. 2009–2022, 2021.
[29] F. Varghese and P. Sasikala, A detailed review based on secure data transmission using cryptography and steganography, Wireless Personal Communications, vol. 129, no. 4, pp. 2291–2318, 2023.
[30] S. Zhu, L. Wu, J. Wang, and Y. Zhang, Secure image encryption scheme based on a new robust chaotic map and strong S-box, Mathematics and Computers in Simulation, vol. 207, pp. 322–346, 2023.
[31] Z. R. Shu, H. C. Deng, P. W. Chan, and X. H. He. Evaluating the intrinsic predictability of wind speed time series via entropy-based approaches. Journal of Wind Engineering and Industrial Aerodynamics, 257:105972, 2025.
[32] Z. Hong, T. Lv, D. Zhao, L. Dong, S. Liu, and S. Zhao. Improvement of pipeline leak detection method: Integration of spectral entropy and sample entropy for better description of complexity features. Applied Acoustics, 231:110458, 2025.
[33] D. Chen, H. Chen, L. Fan, and K. Luo, Error analysis of NIST SP 800- 22 test suite, IEEE Transactions on Information Forensics and Security, vol. 18, pp. 3745–3759, 2023.
[34] M. S. Khan, A. Al-Dubai, J. Ahmad, N. Pitropakis, and B. Ghaleb, A novel feature-aware chaotic image encryption scheme for data security and privacy in IoT and edge networks, arXiv preprint arXiv:2505.00593, 2025.
[35] P. Fedoseev, D. Pesterev, V. Rozhkov, V. Rybin, and D. Butusov, Chaotic encryption algorithm based on Gingerbreadman map with adaptive symmetry, Chaos Theory and Applications, vol. 7, no. 1, pp. 31–41, 2025. [36] M. Ali, J. Ahmad, M. A. H. Khan, S. Ullah, M. U. Rehman, S. A. Shah, and M. S. Khan, A chaotic image encryption scheme using novel geometric block permutation and dynamic substitution, arXiv preprint arXiv:2503.09939, 2025.
[37] H. Al-Asady, an image encryption method based on logistical chaotic maps to encrypt communication data, Kufa Journal of Engineering, vol. 15, no. 4, pp. 55–64, 2024.
[38] J. Sun, Construction of hyperchaotic maps based on 3D-CCC and its ap- plications in image encryption, arXiv preprint arXiv:2503.23655, 2025.
[39] P. K. Pattnaik, S. Swain, and A. K. Rath, Hybrid chaotic-based cryptographic approach for securing IoT-enabled applications in cloud and blockchain environment, IEEE Access, vol. 9, pp. 3105847, 2021.