






## A New Dynamic S-Box-Based Microfluidic Technique

Ali M. Jasim<sup>1</sup> , Isam H. Halil<sup>1</sup> , Nadia M. G. Al-Saidi<sup>2\*</sup> 

<sup>1</sup> Department of Mathematics, College of Science, University of Kirkuk, Kirkuk 36001, Iraq

<sup>2</sup> Department of Applied Sciences, University of Technology, Baghdad 10066, Iraq

Corresponding Author Email: [nadia.m.ghanim@uotechnology.edu.iq](mailto:nadia.m.ghanim@uotechnology.edu.iq)

Copyright: ©2024 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/isi.290536>

### ABSTRACT

**Received:** 24 April 2024

**Revised:** 4 September 2024

**Accepted:** 14 September 2024

**Available online:** 24 October 2024

#### Keywords:

*chaotic system, Dynamic S-Box, microfluidic technology, image encryption, differential attack, cryptography*

This research paper presents a method for encrypting images by combining a 7D-chaotic system with microfluidic techniques to generate dynamic S boxes. The core of this approach involves using a generalized form of the Lorenz system called a 7D system, which offers complex behavior, wide distribution, and good ergodicity. These unique features are used to improve image encryption strategies. This study explores the field of microfluidics with a focus on its control of fluid dynamics at a small scale where surface forces dominate. This interdisciplinary technique combines engineering, physics, and mathematics to create a kind of S-box. A new encryption method is proposed by addressing the limitations of image encryption in large-size images. It demonstrates how integrating behavior and microfluidic technology can create an efficient encryption mechanism suitable for highly secure data applications. The paper provides insights into the algorithm implementation, encryption process, and potential use cases. The effectiveness of this approach is highlighted after implementing some statistical analysis that plays a significant role in demonstrating the encryption method's security. The elevated value of the NPCR and UACI determines its robustness to differential attack analysis.

## 1. INTRODUCTION

In the current era, the field of image encryption has seen notable advancements. However traditional methods still face challenges that limit their effectiveness. While these methods work well for some types of data, they fail to address the complexities presented by image data, such as high redundancy and large file sizes [1-4]. These limitations can be observed in many aspects, for example, Insufficient Complexity; where the simplicity and predictability of their maps make them vulnerable to some types of attacks. Lack of Scalability; where traditional methods face difficulties when dealing with high-resolution images due to the load and time required for encrypting sized files. Security for Sensitive Applications; where the existing encryption methods often fail to meet the security requirements in critical fields, like military and medical imaging where security is paramount. Ensuring the security of the system is one of the most crucial aspects of any cryptographic primitive. Therefore, the challenge is to find an approach that could share unique characteristics with them and overcome such limitations. Due to their properties that meet the cryptography requirements, such as complex behavior, a large distribution, and higher ergodicity, the chaos theory is extremely significant in cryptographic systems [5-8].

Since this discovery, several investigators have presented many encryption systems based on highly performed chaotic systems. For example, Hua et al. [9] proposed a new 2D Logistic-Sine-Coupling Map-based encryption image technique (2D-LSCM). Al-Saidi et al. [10] adopted new 2D

maps to be used for the construction of a new image encryption technique (2D-LICM). Roy et al. [11] investigated colour image encryption algorithm processes which employed the polarization dynamics synchronization in a free-running vertical-cavity surface-emitting laser (VCSEL). Farhan et al. [12] presented a novel chaotic system possessing the peculiar ability to move both inside and outside of a cylinder periodically. In the same year, Ali et al. [6] proposed an encryption method based on a new chaotic map called 2D-HLCM. Additionally, in 2021 Alwan et al. [13] introduced a new chaotic encryption method called 2D-LCHM. In their 2023 study, Ndassi et al. [7] developed an innovative cryptosystem based on compression. This system uniquely blends a high-dimensional chaotic system with a Dynamic S-Box, and utilizes 2D compress sensing for image encryption. The encryption process incorporates a secret key generated through the SHA-256 function. Additionally, the system features a key-dependent Mordell elliptic curve-based Dynamic S-Box for substituting the compressed image. Moreover, Alwan et al. [8] introduced a new encryption algorithm based on an nD-hyperchaotic system derived from the Lorenz system.

This paper introduces a new image encryption method that overcomes limitations in the aforementioned previous methods. This is achieved by combining a 7D chaotic system with a dynamic mechanism to generate new S-boxes utilizing microfluidic technology. This integration does not increase the complexity and unpredictability of the encryption process, it also ensures the scalability and robustness of the algorithm,

which makes it suitable for high-resolution images. The dynamic nature of the S-boxes generated using these new techniques provides a higher level of security that meets the requirements of sensitive applications and represents an advancement in the field of image encryption.

## 2. THE NEW 7D-CHAOTIC SYSTEM

In this section, we introduced a new 7D hyperchaotic system for some reason such as; increased complexity compared to lower-dimensional systems, highly unpredictable and robust cryptographic keys [14, 15]. Furthermore, the additional dimensions in a 7D system refer to more complex interactions between variables, enhancing the system's ergodicity and distribution in the state space. This case ensures achieving the diffusion and confusion principles of Shannon [16]. Based on the Lorenz system [17], the 7D chaotic system is defined by

$$\begin{aligned} \frac{dX_1}{dt} &= (a * x(2) - x(6))x(7) \\ \frac{dX_2}{dt} &= (b * x(3) - x(7))x(1) \\ \frac{dX_3}{dt} &= (c * x(4) - x(1))x(2) \\ \frac{dX_4}{dt} &= (d * x(5) - x(2))x(3) \\ \frac{dX_5}{dt} &= (e * x(6) - x(3))x(4) \\ \frac{dX_6}{dt} &= (f * x(7) - x(4))x(5) \\ \frac{dX_7}{dt} &= (g * x(1) - x(5))x(6) \end{aligned} \tag{1}$$

where,  $x(i) \ i = 1, \dots, 7$  is the initial state and  $(a, b, c, d, e, f, g)$  is the parameters of the system (1).

### 2.1 The equilibria and its stability

Equilibrium points of system (1) are obtained by setting its right-hand side to zero, that is,

$$\begin{aligned} (a * x(2) - x(6)) * x(7) &= 0 \\ (b * x(3) - x(7)) * x(1) &= 0 \\ (c * x(4) - x(1)) * x(2) &= 0 \\ (d * x(5) - x(2)) * x(3) &= 0 \\ (e * x(6) - x(3)) * x(4) &= 0 \\ (f * x(7) - x(4)) * x(5) &= 0 \\ (g * x(1) - x(5)) * x(6) &= 0 \end{aligned} \tag{2}$$

Therefore, we have some equations come from Eq. (2)

$$\begin{aligned} aX_2 &= X_6X_7 \\ bX_3 &= X_7X_1 \\ cX_4 &= X_1X_2 \\ dX_5 &= X_2X_3 \\ eX_6 &= X_3X_4 \\ fX_7 &= X_4X_5 \\ gX_1 &= X_5X_6 \end{aligned}$$

By solving these equations, we have a trivial equilibrium point occurs when all variables are set to zero:

$$X_1 = X_2 = X_3 = X_4 = X_5 = X_6 = X_7 = 0$$

This is because setting all variables to zero satisfies all the

equations simultaneously.

If we assume  $X_1, X_2, \dots, X_7$  are not all zero, we will have:

$$\begin{aligned} X_2 &= \frac{X_6X_7}{a}, X_3 = \frac{X_7X_1}{b}, X_4 = \frac{X_1X_2}{c}, X_5 \\ &= \frac{X_2X_3}{d}, X_6 = \frac{X_3X_4}{e}, X_7 \\ &= \frac{X_4X_5}{f}, X_1 = \frac{X_5X_6}{g} \end{aligned} \tag{3}$$

With simple substitution we have,

$$\begin{aligned} X_2 = \frac{X_6X_7}{a} \text{ and } X_3 = \frac{X_7X_1}{b}, \text{ substitute into } X_4 = \frac{X_1X_2}{c}, \\ \text{we get } X_4 = \frac{X_1 \cdot \frac{X_6X_7}{a}}{c} = \frac{X_1X_6X_7}{ac} \end{aligned} \tag{4}$$

Since

$$\begin{aligned} X_5 &= \frac{X_2X_3}{d}, \\ X_5 &= \frac{\frac{X_6X_7}{a} \cdot \frac{X_7X_1}{b}}{d} = \frac{X_1X_6X_7^2}{abd} \end{aligned} \tag{5}$$

Substitute  $X_4$  and  $X_5$  back into  $X_6 = \frac{X_3X_4}{e}$ , we have:

$$X_6 = \frac{\frac{X_7X_1}{b} \cdot \frac{X_1X_6X_7}{ac}}{e} = \frac{X_1^2X_6X_7^2}{abce} \tag{6}$$

and for  $X_7 = \frac{X_4X_5}{f}$ , we have

$$X_7 = \frac{\frac{X_1X_6X_7}{ac} \cdot \frac{X_1X_6X_7^2}{abd}}{f} = \frac{X_1^2X_6^2X_7^3}{abcdf} \tag{7}$$

Substituting the above values into the other equations leads to a complex set of nonlinear equations. Therefore, the non-trivial equilibrium points depend on the specific values of the parameters and require complex numerical methods to solve due to the nonlinear nature of the system.

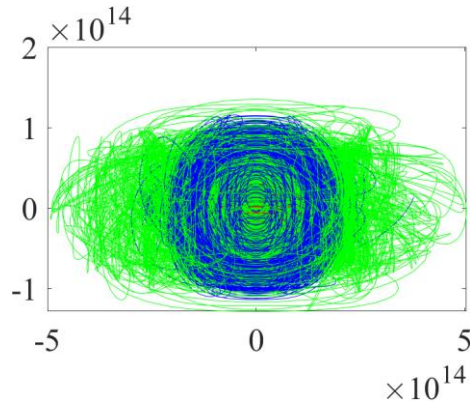
The selection of parameters plays a role in achieving the desired behavior and ensuring the robustness of the encryption process. These parameters were chosen based on the following factors [9, 11, 18, 19]:

- Complexity and Sensitivity; We carefully selected parameters that exhibit a level of complexity and sensitivity to conditions. This is vital for generating sequences that enhance the security of the encryption.
- Distribution and Ergodicity; Our chosen parameters ensure a uniform distribution of the map across the state space contributing to the system ergodicity. This uniform distribution is essential for creating an unpredictable S-box.
- Real-world Applicability; Practical applicability in real-world scenarios was also taken into consideration when selecting these parameters. We aimed to ensure that the system can be efficiently implemented without requiring resources.
- Robustness to Cryptanalytic Attacks; The selection process also focused on resistance against attacks. Parameters that contribute to a level of security against

linear cryptanalysis were given priority.

To understand how the 7D hyperchaotic system behaves under certain conditions and ascertain its suitability, for encryption purposes it is crucial to conduct a stability analysis.

### 2.2 Attractor



**Figure 1.** Attractors of 7D-hyperchaotic system

It visualizes and analyzes how the system evolves through iterations. These plots provide insights into how slight changes in conditions affect the system response. A dynamical system's attractor represents a set of points in state space that depict the output of the system. Figure 1. shows the attractor of the 7D-

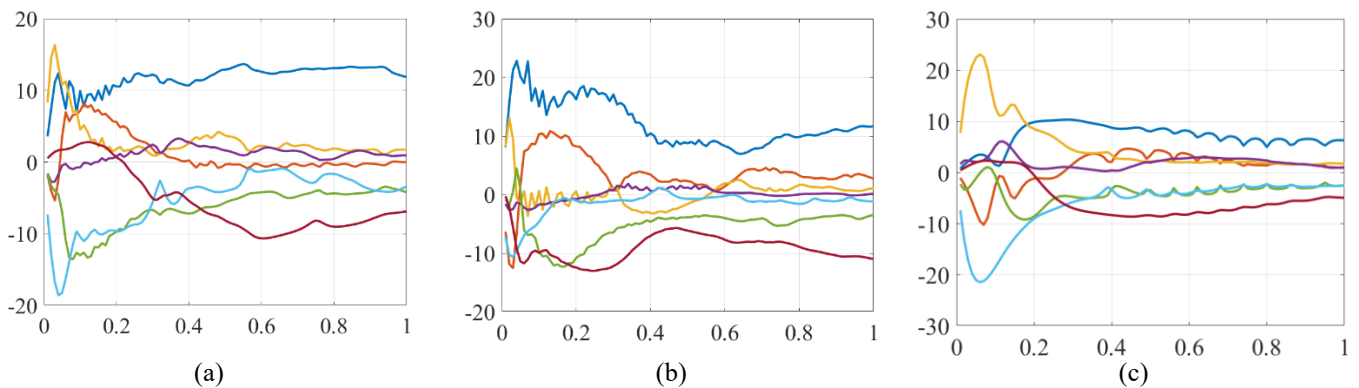
hyperchaotic system utilizing different parameters values, in this case,  $a=40; b=60; c=8; d=20; e=0.1; f=77; g=10$ . Figure 1(a) shows attractor diagrams with different values of the initial states  $(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = (1, 1, 2, 2, 1, 1, 1)$ ,  $(1, 1, 2, 2, 1, 1, 0.1)$ , and  $(1, 1, 2, 2, 1, 0.1, 0.1)$  blue, red and green, respectively.

### 2.3 Lyapunov exponents

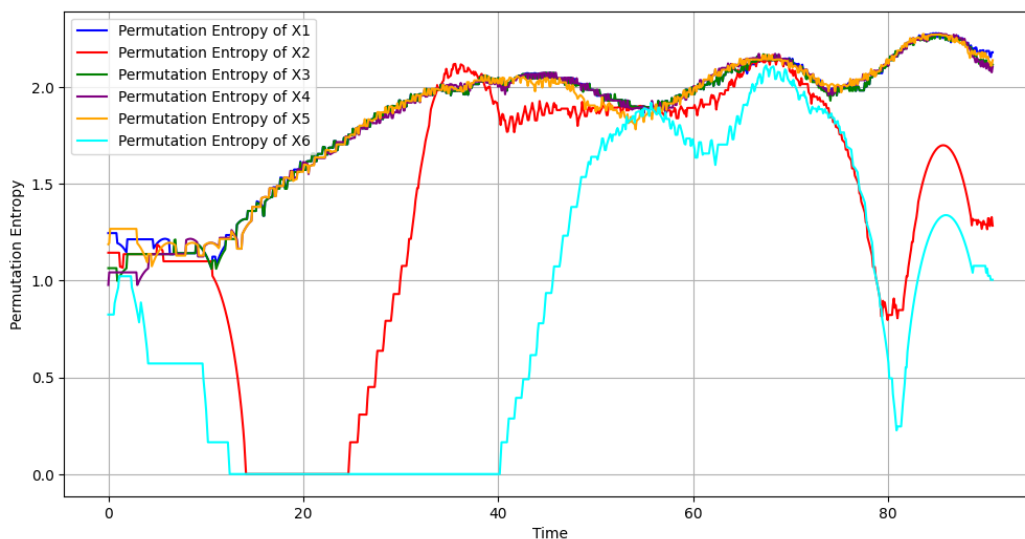
Lyapunov exponents (LE) are computed to evaluate how sensitive the system is to its conditions. A positive Lyapunov exponent indicates behavior, which is desirable for encryption. The calculation of these exponents for the 7D system is complex. Requires methods and a thorough understanding of the system's dynamics. The rate of divergence or convergence of surrounding trajectories is measured by LE. It can be mathematically defined as [5]:

$$\lambda \cong \frac{1}{t} \ln \frac{\|\delta x(t)\|}{\|\delta x(0)\|} \quad (8)$$

where,  $\frac{\|\delta x(t)\|}{\|\delta x(0)\|}$  refers to the distance between two different trajectories as illustrated in Figure 2. In this Figure 2(a), (b) and (c), show hyperchaotic behavior of the proposed system, especially, for the parameters  $(a=40; b=60; c=8; d=20; e=0.1; f=77; g=10)$ ,  $(a=40; b=60; c=8; d=20; e=0.1; f=77; g=10)$ , and  $(a=40; b=60; c=8; d=20; e=0.1; f=77; g=10)$ , respectively.



**Figure 2.** LE of the 7D-hyperchaotic system



**Figure 3.** PE of all variables

## 2.4 Permutation entropy

Entropy is a concept in information theory, thermodynamics, and dynamical systems, that measures the unpredictability, or the amount of information in a system [8]. In this paper, we introduced a special type of entropy called Permutation Entropy (PE). The measure is defined as;

$$H_{PE} = - \sum_{i=1}^N p_i \log_2 p_i$$

where,  $p_i$  is the probability of the  $i$ -th permutation pattern occurring in the time series, and  $N$  is the total number of possible patterns (which depends on the window length). The average PE across multiple windows is also computed to provide a single value that represents the complexity of the variable's time series, where the parameter set  $(a, b, c, d, e, f)$  maximizes the system's overall complexity. A parameter sweep across specified ranges is performed. For each combination of parameters, the system of differential equations was numerically integrated to produce time series data, and the average PE across all dimensions was calculated. The parameter set that resulted in the highest average PE was selected as the optimal configuration, indicating the most chaotic system behavior. Figure 3 shows the PE of all variables where the best parameters are;  $a=55.0, b=40.0, c=10.0, d=20.0, e=0.075, f=75.0$  and the initial values are:

$$x(1) = 0.1, x(2) = 0.1, x(3) = 0.1, x(4) = 0.1, x(5) = 0.1, x(6) = 0.1 \text{ and } x(7) = 0.1$$

## 3. NEW IMAGE ENCRYPTION METHOD

The algorithms of image encryption are divided into two categories. One treats a digital image as a bitstream and encrypts the image using traditional techniques. The other category uses current techniques like chaos, wavelet transforms, and so on. The performance of the algorithms that are based on chaotic maps is good [5].

S-boxes play a role, in image encryption because of the connection between neighbouring pixels, in images. If not properly encrypted this correlation makes images available to attacks. By using S-boxes we can minimize this correlation and enhance the security of the encrypted image. This section introduces a new S-box and strings letters and mixture them via the microfluidic technique. However, this S-box is adopted to generate keys for the image encryption algorithm.

### 3.1 Design of a new S-box

S-boxes play a crucial role in providing the strength, reliability, and security of image encryption methods. This is achieved due to its nonlinearity improves confusion and diffusion and offers protection against different types of cryptographic attacks. Microfluidic technology (MF) offers unique advantages in controlling and manipulating fluids at a micro-scale, where surface forces such as capillary action and electrokinetic effects dominate [20]. These principles enable the precise generation of complex cryptographic elements, making microfluidics a valuable tool in enhancing the security and efficiency of encryption systems. By understanding and leveraging these forces, microfluidic devices can produce

highly secure, dynamic cryptographic components that are difficult to predict [21]. MF can be used to mix fluids that represent different chaotic sequences such as system (1) in a highly controlled environment. The chaotic nature of the system, combined with the micro-scale precision of fluid manipulation, results in the generation of dynamic and highly unpredictable cryptographic elements such as S-box [21, 22].

In this work, a new Dynamic S-Box is unique due to its integration with advanced MF and the use of a 7D hyperchaotic system is designed. This combination allows for the real-time generation of highly adaptive and unpredictable S-boxes, significantly enhancing security by introducing dynamic variability with each encryption process. The MF ensures precise control and mixing of chaotic sequences, which is difficult to achieve with conventional methods, thereby providing a superior defense against cryptographic attacks. Figure 4 shows the flowchart of the S-box generation steps.

This section introduced a procedure to generate a new S-box based on system (1) and MF. Figure 3 shows this procedure.

#### 3.1.1 Generate random sequence based on system (1)

To generate random sequence based on system (1). First step is set initial state and parameters of system (1) with set iterations and time, after that, we implement the system generates seven randoms chaotic sequences, these steps illustrated in Algorithm 1.

---

#### Algorithm 1: Generate SC chaotic system sequence

---

**Input:** initial parameters  $(a, b, c, d, e, f, g)$ ;

**Output:** Random sequence;

**Begin**

**for**  $i = 1$  **to** 7

    Apply (1) to generate random numbers with 7 dimensions

$X = [x_1, x_2, x_3, x_4, x_5, x_6, x_7], x_i$  with size  $n, i = 1, 2, \dots, 7$ ;

$SC(i) = \text{sum}(x_i)$ ;

**End**

$SC = \text{random permeation}(SC(i))$ , by MF;

Random sequence  $SC$ ;

**End**

---

#### 3.1.2 Microfluidic mixer to generate S-box

In this phase, we generated an S-box based on system (1), and a string of letters, then mixed them via the microfluidic mixer; as shown in Algorithm 2 and Table 1.

---

#### Algorithm 2: Generate an S-box via microfluidic

---

1. **Input:**  $SC$ , a string of letters and numbers and some symbols  $Sl$ ;

2. **Output:** S- box

**Begin**

3. Convert  $SC$  into binary numbers;

4. Generate sequences of letters and numbers and some symbols  $Sl$ ;

5.  $Sl = \text{random permeation}(Sl)$ , by Microfluidic technique;

6. Convert  $SC$  into binary numbers;

7.  $S = \text{random permeation}(Sl, SC)$ , by Microfluidic technique;

8.  $S\text{-box} = \text{reshape}(S)$  into  $n \times n$ ;

9. **end**

---

Table 1 shows a sample of  $10 \times 10$  S-box generated via microfluidic technique with the hexadecimal number. The size of the S-box can be changed according to the size of the image to be encrypted.

The utilization of the microfluidic technique offers numerous benefits compared to traditional methods in the

generation of S-boxes for encryption purposes. The utilization of the microfluidic technique provides a dynamic generation and precise control, integration with chaotic maps. It is an excellent choice for enhancing the encryption algorithm's security due to its efficiency and scalability.

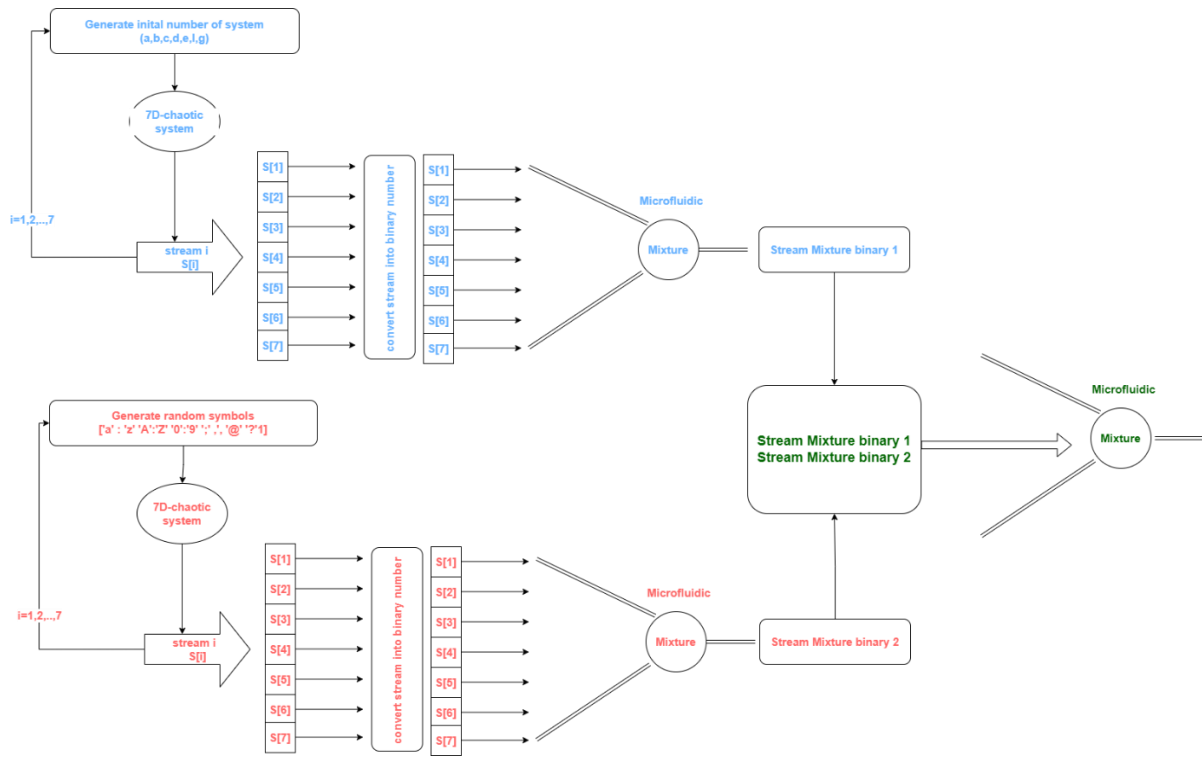


Figure 4. Flowchart of S-box generation steps

Table 1. Sample of S-box generated via a microfluidic technique of size  $10 \times 10$

S-Box via Microfluidic									
Col 1	Col 2	Col 3	Col 4	Col 5	Col 6	Col 7	Col 8	Col 9	Col 10
12D0	8316	B162	060C	42D1	0CB8	C483	52B0	8B06	3162
60C1	0C58	C18C	1831	8B16	3062	060C	64C1	2C18	C58B
83C6	3162	060C	62B2	0B58	C183	5831	8B16	B160	060C
2B58	B18B	12C0	836B	3C62	062C	60C5	0C1B	C5CB	38C0
B160	162C	621C	0B18	C185	18B1	8326	B060	160C	62C5
C13B	18C1	8306	3060	142C	62C5	1C58	C28B	18B0	8B16
18C0	8B16	3060	060C	62C5	2C18	C583	38B0	8B16	B062
62C1	2C38	328B	58C1	8B16	3160	16C2	62B1	2C58	C1B3
8B06	30B2	150C	60B5	2C85	C583	5831	8313	B160	062B
2C5B	C15B	18B2	8B06	3162	1B2C	62C5	BC18	C182	1830

Also, the dynamic nature of the S-boxes generated by microfluidics makes it challenging for cryptanalysis resistance to attacks. Attackers cannot rely on precalculated tables or traditional methods of analyzing cryptographic systems, as the properties of the S-boxes change in an unpredictable manner with each encryption instance. Microfluidic techniques have a wide range of applications for data encryption, not just limited to images. This adaptability makes this technique valuable in various fields that require strong encryption, such as telecommunications, data storage and secure transmissions.

### 3.2 S-box performance metrics

At this stage of our work, we need to test our proposed algorithm using well-known analysis frameworks in the scientific community to ensure the effectiveness of our S-box.

For example, we will evaluate its performance based on nonlinearity, the Avalanche Effect (AE), and the Output Bit Independence Criterion (BIC) to demonstrate the effectiveness of the proposed S-box design.

#### 1. Nonlinearity

One of the tests to measure the efficiency of an S-box is nonlinearity, which is defined as the minimum Hamming distance between the S-box function and the set of all affine functions [22], it can be mathematically defined as:

$$S_{(f)}(\omega) = \sum_{\omega \in GF(2^n)} (-1)^{f(x) \oplus x \cdot \omega} \quad (9)$$

where, the dot product between  $x$  and  $\omega$  is defined by:

$$x \cdot \omega = x_1 \cdot \omega_1 \oplus x_2 \cdot \omega_2 \oplus \dots \oplus x_n \cdot \omega_n$$

Thus, the nonlinearity is calculated by

$$N_f = 2^{n-1} \left( 1 - 2^{-n} \max_{\omega \in GF(2^n)} |S_{(f)}(\omega)| \right) \quad (10)$$

The nonlinearity of the proposed S-box and some previous methods are shown in Table 2.

**Table 2.** The nonlinearity of some S-boxes

S-Box Methods	Value
Proposed	106
[12]	106
[23]	105
[24]	100
[25]	106

## 2. Avalanche effect (AE)

The AE measures how much the output changes when a single input bit is flipped. The general idea, where a flipping one bit of the input sequence has the effect of changing about half of the bits in the output [23]. Table 3 shows the AE of the proposed S-box and some previous methods.

**Table 3.** AE values of some S-boxes

S-Box Methods	AE
Proposed	0.7555625
Ref. [12]	0.5783
Ref. [24]	0.4971
Ref. [25]	0.4825
Ref. [26]	0.4916

## 3. Bit Independence Criterion (BIC)

The BIC measures how independently the output bits behave when specific input bits are flipped. It checks the independence between pairs of output bits for different input bit changes. The general idea of BIC lies in the flipping of one input bit will cause an independent and unpredictable changes in the output bits. The BIC values should show minimal correlation between the effects on different output bits [23]. Table 4 shows the BIC of our proposed S-box and some other S-boxes from the literature.

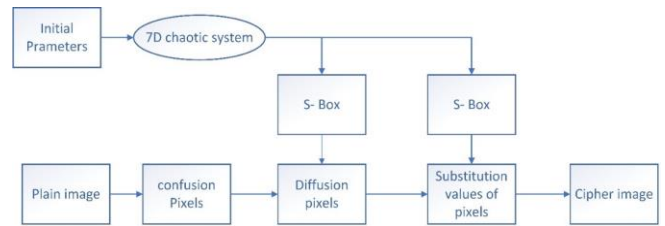
**Table 4.** BIC of the proposed S-box and some other S-boxes

S-Box Methods	BIC-SAC	BIC-Nonlinearity
Proposed	0.6064	104.2
Ref. [12]	0.5064	103.92
Ref. [24]	0.5044	102.96
Ref. [25]	0.4962	101.9
Ref. [26]	0.5058	104.14

## 3.3 Encryption part

In this part, three steps (Confusion Pixels, Pixels-permutation, Substitution) are implemented to produce the encrypted image. In our proposed encryption algorithm, we choose some parameters to generate S-boxes using a system (1) with parameters:  $a = 40, b = 60, c = 8, d = 20, e = 0.1, f = 77,$  and  $g = 10,$  and initial state  $x(i) = 0.1$ . These parameters were chosen to ensure that the chaotic system exhibits strong sensitivity to initial conditions and generates highly unpredictable sequences, essential for secure S-box

construction based on its evaluation tests. The system (1) produced by iterating this system is mapped to an 8x8 S-box, with each sequence value normalized and converted into an S-box entry. To add an additional layer of randomness, microfluidic technology is employed to mix and refine the chaotic sequences, resulting in a unique S-box for each encryption session. This S-box is then integrated into the image encryption process at the substitution stage, where it replaces pixel values in the image with their corresponding S-box values. The continuous adaptation of the S-box for each block of data significantly enhances the security, making the encryption resistant to a variety of cryptographic attacks. They are described as a block diagram in Figure 5.



**Figure 5.** The encryption process structure

### 3.3.1 Confusion process

The process of rearranging the positions of pixels is referred to as confusion. Let us say we have a plain image called  $I$  with dimensions  $m \times n$  where each pixel's value falls within the range of  $[0, 255]$ . To change the locations, we create a random matrix that matches the size of the original image. Then we proceed to modify the locations based on this random matrix. The confused pixels of the image can be generated as shown in Algorithm 3. In the decryption part, the confusing image should pass through Algorithm 4 which represents the inverse operations.

#### Algorithm 3: Confusion process

- Input:** plain image  $I$ .
  - Output:** Image  $P$  after the confusion process
- Begin**
- $R$  = row's size of  $I$ ;  
 $C$  = column's size of  $I$ ;
  - $R\_new$  = a new random  $R$ ;  
 $C\_new$  = a new random  $C$ ;
  - for**  $i = 1$  **to**  $R$
  - for**  $j = 1$  **to**  $C$
  - $P = I(R\_new(i), C\_new(j))$ ;
  - end**
  - end**
  - end**

#### Algorithm 4: The inverse of the Confusion process

- Input:** Confused image  $P$ ;  $R\_new$ ;  $C\_new$ ;
  - Output:** the original image  $I$ ;
- Begin**
- $R$  = row's size of  $I$ ;  
 $C$  = column's size of  $I$ ;
  - for**  $i = 1$  **to**  $R$
  - for**  $j = 1$  **to**  $C$
  - $I(i,j) = P(R\_new(i), C\_new(j))$ ;
  - end**
  - end**
  - end**



### 3.3.2 Diffusion process

The process of altering the image's values is known as diffusion. Let us say we have an image called  $I$  with dimensions  $m \times n$ , where each pixel has a value ranging in  $[0, 255]$ . To modify these values, we apply the XOR operation using an S-box. The diffusion pixels of the image can be generated as shown in Algorithm 5. In the decryption part, we used Algorithm 6 which performed the inverse operations on the diffused image.

#### Algorithm 5: Diffusion pixels

```

1. Input: Confused image  $P$ ; S-box;
2. Output: Diffusion image  $D$ ;
   Begin
3.  $R = \text{size of the row of } P$ ;
    $C = \text{size of the column of } P$ ;
4. for  $i = 1$  to  $R$ 
5.   for  $j = 1$  to  $C$ 
6.      $D(i, j) = P(i, j) \text{ XOR } S\_box(i, j)$ ;
7.   end
8. end
   end

```

#### Algorithm 6: Diffusion inverse

```

1. Input: Diffusion image  $D$ ; S-box;
2. Output: Confused image  $P$ ;
   Begin
3.  $R = \text{size of the row of } D$ ;
    $C = \text{size of the column of } D$ ;
4. for  $i = 1$  to  $R$ 
5.   for  $j = 1$  to  $C$ 
6.      $P(i, j) = D(i, j) \text{ XOR } S\_box(i, j)$ ;
7.   end
8. end
   End

```

### 3.3.3 Substitution of pixel values

This part of the encryption operation used the S-box to change the values of pixels via substitution values of the image with values of the S-box. The substitution values of image pixels are generated by Algorithm 7. Algorithm 8 is used to perform the inverse operations on the substitution values of the image in the decryption part.

#### Algorithm 7: Substitution of pixel values

```

1. Input: Diffusion image  $D$ ; S-box;
2. Output: Cipher image  $C$ ;
   Begin
3.  $R = \text{size of the row of } D$ ;
    $C = \text{size of the column of } D$ ;
4. for  $i = 1$  to  $R$ 
5.   for  $j = 1$  to  $C$ 
6.      $C(i, j) = S\_box(i, j)$ ;
7.   end
8. end
   end

```

#### Algorithm 8: Inverse of substitution process

```

1. Input: Cipher image  $C$ ; S-box;
2. Output: Diffusion image  $D$ ;
   Begin
3.  $R = \text{size of the row of } C$ ;
    $C = \text{size of the column of } C$ ;

```

```

4. for  $i = 1$  to  $R$ 
5.   for  $j = 1$  to  $C$ 
6.      $D(i, j) = S\_box(i, j)$ ;
7.   end
8. end
   end

```

After these three operations (confusion pixels, diffusion Pixels and substitution of Pixel's values), the cipher image is obtained. The main steps are abstracted in Algorithm 9.

#### Algorithm 9: Encryption process

```

1. Input: Plain image  $P$ ; S-box;
2. Output: Cipher image  $C$ ;
   Begin
3.  $R = \text{size of the row of } I$ ;
    $C = \text{size of the column of } I$ ;
4. Apply confusion pixels algorithm % Algorithm 3
5. Apply diffusion pixels algorithm % Algorithm 5
6. Apply substitution values of pixels algorithm %
   Algorithm 7
7. Cipher image  $C$ ;
   end

```

### 3.4 Decryption process

In general, the decryption operation is the inverse operation of encryption algorithm, which means the last stage of encryption become the first stage in the encryption algorithm. The steps, for decryption are outlined in Figure 6.

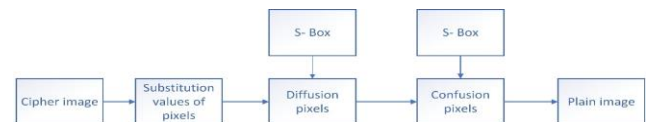


Figure 6. The decryption processes

## 4. SIMULATION RESULTS

To assess the quality of encrypted images, we can employ various statistical tests. At the beginning of our experiment, we chosen carefully the parameters of system (1), as these choices significantly impact the randomness and unpredictability of the encrypted images. The parameters  $a = 40$ ,  $b = 60$ ,  $c = 8$ ,  $d = 20$ ,  $e = 0.1$ ,  $f = 77$ , and  $g = 10$  and the initial state of each variable was set to 0.1 to maximize the system's sensitivity to initial conditions, thereby producing sequences that are highly unpredictable. One such test is the histogram test, which helps analyze the uniform distribution of pixel values in the encrypted images. A high level of statistical significance in these tests indicates that your encryption technique effectively randomizes pixel values, crucial for thwarting statistical attacks. Another analysis method involves evaluating the correlation coefficients between adjacent pixels in the encrypted images. When the correlation coefficient is lower and closer to zero, it produces a higher level of security as it indicates that the encryption process efficiently breaks down any correlation present in the original image. We can also calculate the information entropy of encrypted images to gain insights into their randomness. Ideally, an entropy value close to 8 (for 256 grey levels) signifies a high level of randomness present in the encrypted image.

### 4.1 The histogram

One of the most effective tests to show the efficiency of a powerful encryption algorithm is histogram analysis. A histogram is a graphical representation showing the frequency distribution of pixel intensity values in the image. In our simulation we applied our algorithm to several images and Figure 7 illustrates a comparative analysis of image histograms before and after encryption. Subfigure (a) shows the plain images in their original form, while subfigure (b) presents the corresponding histograms, highlighting the pixel intensity distributions, also, subfigure (c) depicts the encrypted versions of the same images from (a), and subfigure (d) shows the histograms of these encrypted images.

### 4.2 Correlation measure

The correlation coefficient  $CC$  values from -1 to 1, if  $CC =$

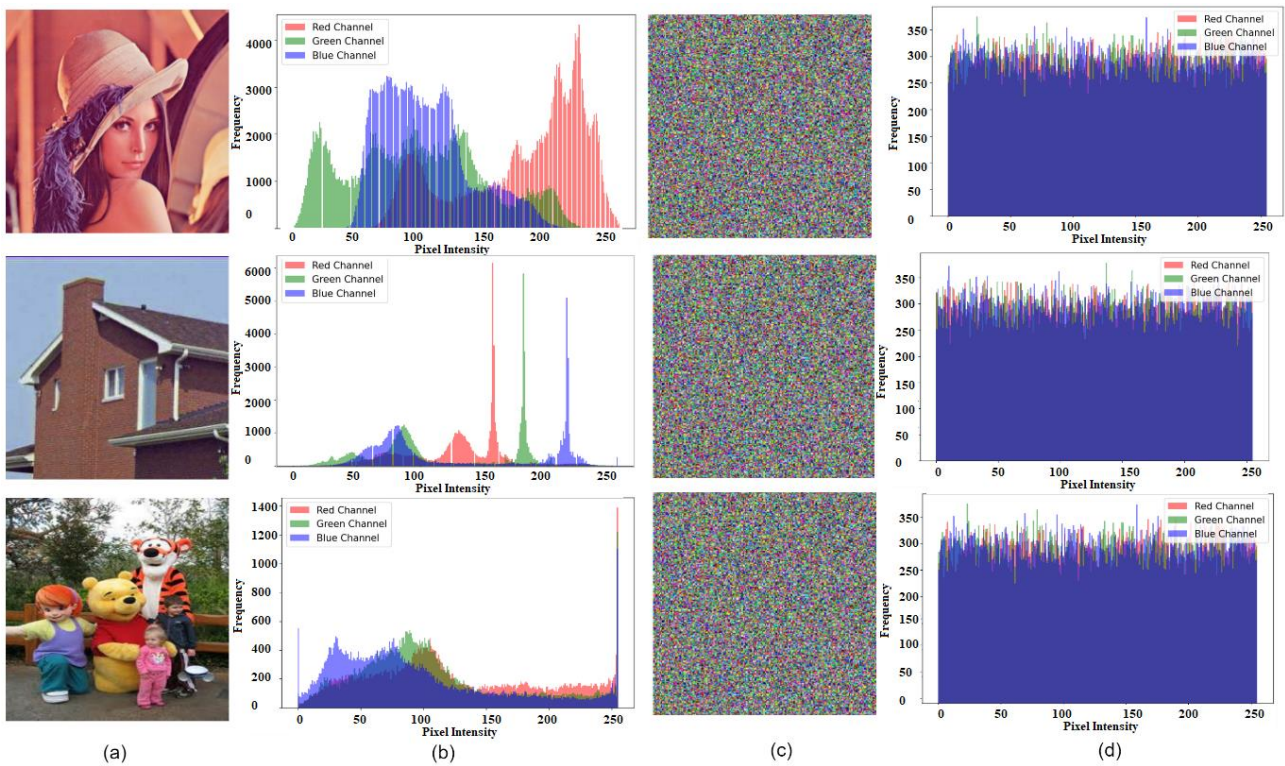
1 that means a perfect positive linear relationship,  $CC = -1$  mean perfect negative linear relationship, and  $CC = 0$  mean no linear relationship. The correlation measure can be defined mathematically as:

$$CC = \frac{E(x - E(x))E(y - E(y))}{\sqrt{D(x)}\sqrt{D(y)}}$$

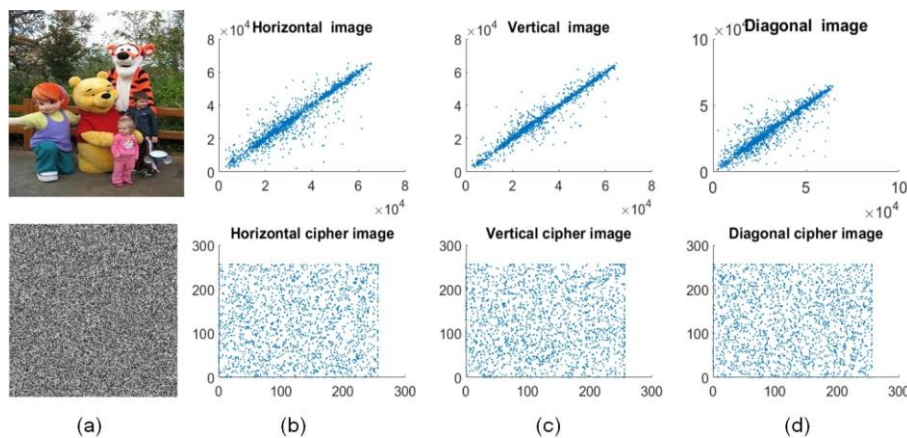
where,

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

The results of some colour images for the correlation of pixels are shown in Table 5. Figure 8 shows the correlations of the images in Winne bear directions.

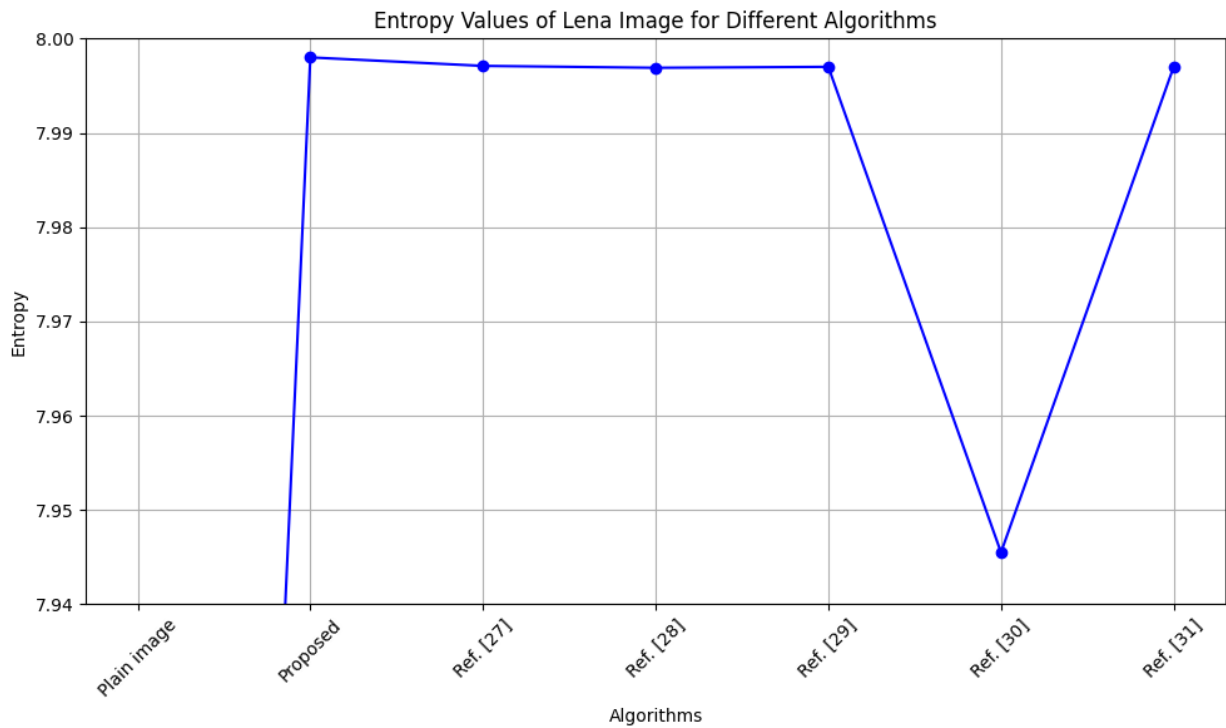


**Figure 7.** The histogram of some images: (a) plain images; (b) histogram of (a); (c) the encrypted images of (a); (d) histogram of (c)



**Figure 8.** Correlation of Winne bear: (a) original image and its cipher image, where (b), (c) and (d) shows three directions, vertical, horizontal, and diagonal, respectively





**Figure 9.** Entropy values of Lena’s image for different algorithms

**Table 5.** Correlation coefficients of some plain and cipher images

Name	Plain Color Image			Cipher Image		
	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Lena	0.932142	0.94836	0.91287	-0.000285	-0.00578	-0.00316
Tree	0.963143	0.96724	0.92598	-0.005129	-0.00512	-0.00372
Jelly Beans	0.9618	0.97963	0.94991	-0.000878	-0.00406	-0.00144
Moon	0.96243	0.9639	0.91748	-0.000113	0.002852	0.001445

**Table 6.** Entropy of some images

Image	Plain-Image	Cipher Image
<b>Lena</b>	7.4530	7.998
<b>Tree</b>	7.652	7.998
<b>Peppers</b>	7.6840	7.9979

**Table 7.** Values of entropy test for Lena image

Methods	Values
Plain Image	7.5984
Proposed	7.998
Ref. [27]	7.9971
Ref. [28]	7.9969
Ref. [29]	7.9970
Ref. [30]	7.9455
Ref. [31]	7.9970

### 4.3 The entropy measures

Shannon [16] proposed the first haphazardness measure in 1948, which determined the expected information value in communication in bits units. The entropy of Shannon can be defined as:

$$H(e) = \sum_{i=1}^E p(e_i) \log \frac{1}{p(e_i)}$$

where,  $E$  is the total number of symbols  $e_i$ , and  $p(e_i)$  refers to

the probability of  $e_i$ . If the information source transmits 256 symbols, then  $H(e) = 8$ . The entropy results of some images are illustrated in Table 6. Table 7 show the comparison of entropy values of Lena’s image between our proposed encryption algorithm and some previous algorithms. Figure 9 shows the entropy values of Lena’s image for different algorithms.

## 5. THE SECURITY PERFORMANCE MEASURE

For evaluating the security of the encryption technique to withstand different attacks for example differential attacks, where these attacks involve making small changes to the original image and observing the resulting differences in the encrypted output. Some performance measures are implemented to show the effectiveness of the proposed technique. The dynamic nature of the generated S-box, which changes with each encryption instance significantly enhance security against both differential and linear cryptanalysis.

### 5.1 Analysis of key sensitivity performance

The Mean Absolute Error (MAE) serves as a technique for measuring the disparity between two continuous variables. In the context of image encryption, it functions to ascertain the dissimilarity between two images: the original (plain) image and the encrypted (ciphered) image. The MAE is computed

using the following formula:

$$MAE = \frac{1}{row \times col} \sum_{i=0}^{row-1} \sum_{j=0}^{col-1} |P_{i,j} - C_{i,j}|$$

where the *row* and *col* represent the dimensions of the images, referring to the number of rows and columns, respectively of the image. While the  $P_{i,j}$  refer to the value of the pixel at the location *i*-th row and *j*-th column of the plain image, while  $C_{i,j}$  refer to the value of the pixel at the location *i*-th row and *j*-th column of the ciphered image. A higher MAE value indicates a significant difference between the plain and cipher images, signifying a favourable outcome for encryption as it suggests that the encrypted image bears little resemblance to the original, making decryption more challenging. Mean Squared Error (MSE) provides a measure of dissimilarity between two images by calculating the average of the squared differences between corresponding pixels:

$$MSE = \frac{1}{row \times col} \sum_{i=0}^{row-1} \sum_{j=0}^{col-1} (P_{i,j} - C_{i,j})^2$$

The analysis of pixel differences between the plain and cipher images is presented in Table 8.

### 5.2 Number of Pixels Change Rate (NPCR)

The NPCR, or the Number of Pixel Changes Rate, quantifies the percentage of pixel positions where two encrypted images exhibit differences. In simpler terms, it gauges the extent of pixel alterations when a single pixel in the original image changes. If  $C_1(i, j)$  and  $C_2(i, j)$  represent two encrypted images, the NPCR is given by the formula:

$$NPCR = \frac{1}{row \times col} \sum_{i,j} x(i, j)$$

where, rows and cols are the dimensions (height and width) of the images. The function  $x(i, j)$  equals 0 if the pixel values at position  $(i, j)$  in both encrypted images are identical, and 1 if

they differ, specifically defined as 0 if  $C_1(i, j) = C_2(i, j)$  and 1 if  $C_1(i, j) \neq C_2(i, j)$ . A higher NPCR value means that a significant number of pixels have changed between the two encrypted images, indicating that the encryption algorithm is highly sensitive to minor modifications in the input image. Essentially, NPCR evaluates the robustness of the encryption scheme against differential attacks by analyzing how the encrypted image responds to small changes in the original image.

**Table 8.** Difference analysis of pixels between plain image and cipher image

Image Name	Our Proposed		Ref. [27]	
	MAE	MSE	MAE	MSE
Pepper	85.52	9159.72	85.48	8992.82
Lena	81.34	9737.35	79.88	8765.76
Baboon	82.11	9578.2	81.53	8619.66
Jellybeans	73.24	9489.11	66.83	8566.12
Tree	84.96	941.10	78.99	7436.10

### 5.3 Unified Average Intensity Change Intensity (UACI)

The UACI is a metric evaluating encryption algorithm sensitivity by measuring the average intensity difference between two encrypted images. It is defined as;

$$UACI = \frac{1}{row \times col} \sum_{i=0}^{row-1} \sum_{j=0}^{col-1} \left| \frac{C_1(i, j) - C_2(i, j)}{255} \right|$$

The biggest values of UACI refer to significant intensity differences, which means the encryption scheme introduces substantial changes even with minor alterations to the input image. To assess the encryption scheme's sensitivity through NPCR or UACI, begin by encrypting the original image ( $P$ ) to create  $C_1$ . Introduce a random alteration to a single pixel in  $P$ , and subsequently encrypt this modified image to generate  $C_2$ . Calculate NPCR or UACI using the specified formulas, and then compare the resulting cipher images,  $C_1$  and  $C_2$ . The experimental findings of these evaluations are presented in Table 9 and Table 10.





**Table 9.** Analysis of NPCR test between plain and cipher images in comparison to Ref. [27]

Image Name	NPCR				Ref. [27]			
	Gray	R	G	B	Gray	R	G	B
Pepper	99.84	99.89	99.89	99.99	99.92	99.72	99.82	99.61
Lena	99.83	99.87	99.86	99.83	99.86	99.86	99.81	99.89
Baboon	99.70	99.89	99.89	99.89	99.88	99.85	99.72	99.87
Jellybeans	99.83	99.81	99.82	99.71	99.81	99.82	99.83	99.77
Tree	99.88	99.89	99.98	99.89	99.82	99.76	99.67	99.87

**Table 10.** Analysis of UACI test between plain and cipher images in comparison to Ref. [27]

Image Name	UACI				Ref. [27]			
	Gray	R	G	B	Gray	R	G	B
Pepper	33.51	33.84	33.89	33.81	33.58	36.39	33.14	35.26
Lena	33.86	34.93	34.81	34.80	33.68	34.97	33.06	33.81
Baboon	33.47	34.81	34.82	34.86	33.64	35.48	33.06	34.81
Jellybeans	33.89	34.89	34.84	34.89	33.21	32.94	31.85	33.18
Tree	33.87	33.45	33.86	33.72	33.47	32.56	34.11	32.25

**Table 11.** GLCM analysis

Plain Image	ID	E	Cots	Max <sub>D</sub>	Hog
	85302	0.025671	4851.43	$6.4510 \times 10^5$	0.261361
	83745	0.025171	4836.14	$6.427 \times 10^5$	0.260281
	83256	0.025035	4843.54	$6.4206 \times 10^5$	0.261711
	83254	0.026032	4842.54	$6.9420 \times 10^4$	0.261651

#### 5.4 Analysis of Gray-Level Co-Occurrence Matrix

In this test, we will highlight the performance of the proposed encryption algorithm based on the Gray Level Co-occurrence Matrix (GLCM) test, which is a description of some characteristics such as Irregular deviation (*ID*) contrast (*Cots*), energy (*E*), and uniformity [9]. Each one of them can be defined as the following:

##### 5.4.1 Irregular deviation

The Irregular Deviation (*ID*) quantifies the precise difference between each pixel in the plain image (*I*) and its corresponding pixel in the encrypted image (*C*). *ID* is then calculated by taking the average of the absolute differences between the histogram  $M_H$  and its mean value  $m_{DH}$  [27].

$$\begin{cases} HG(i) = |I(i) - C(i)| \\ ID = \frac{\sum_1^{256} |M_H(i) - m_{DH}|}{M \times N} \end{cases} \quad (11)$$

where, *I* refer to plain image, while *C* refer to cipher image. Also,  $M_H$  refer to the frequency distribution of pixel intensity values that is mean histogram of *HG*, and  $m_{DH}$  refer to the mean value of the *ID*.

##### 5.4.2 Maximum deviation

The Maximum Deviation ( $Max_D$ ) metric measures the highest possible difference within the histogram when comparing the *I* and *C* images. It takes into account the amplitude values of the histogram at the extreme ends (1 and 256) and sums up the values from 2 to 255, can be defined by the  $Max_D$  [27].

$$Max_D = \frac{AM_v(i) + AM_v(256)}{2} + \sum_{i=2}^{255} H(i) \quad (12)$$

where,  $AM_v(i)$  is the amplitude value of the *HG* mentioned in the above Section.

##### 5.4.3 Contrast

Contrast (*Cots*) is used to measure the difference in brightness or colour that allows an object within an image to be distinguished. In the context of GLCM, it evaluates the degree of variation between pairs of pixels in the image. *Cots* is then calculated by [27]:

$$Cots = \sum_{i,j} |i - j|^2 G(i,j) \quad (13)$$

##### 5.4.4 Energy

Energy (*E*) in general is a measure of the uniformity or orderliness of an image, where the low value of *E* refers to the *C*, which means a high randomness. The formula for calculating energy is [27]:

$$E = \sum_{i,j} G(i,j)^2 \quad (14)$$

##### 5.4.5 Homogeneity

Homogeneity (*Hog*) is used to measure the similarity between pixel pairs in an image, typically yielding a result between 0 and 1, where the low value of this measurement refers to the *C*, which exhibits significant variation between pixel pairs, and this tends to robust encryption and effective concealment of the *P*. The *Hog* can be calculated as [27]:

$$Hog = \sum_{i,j} \frac{G(i,j)}{1 + |i - j|} \quad (15)$$

Table 11 presents some results of the GLCM analysis for several enciphered images.

#### 5.5 Analysis of resistance to classical attacks

To maintain basic security requirements for any encryption algorithms, four classical attacks should be discussed. In the chosen plaintext attack, an unauthorized individual introduces

forged text without knowing the key but somehow obtains the corresponding ciphertext. Another technique is the chosen-ciphertext attack, in which the attacker creates fake ciphertext and plaintext in order to obtain the relevant data illegally. The difference between ciphertext-only and known-plaintext attacks is that the former only includes obtaining the ciphertext, while the latter only involves obtaining the plaintext. Nonetheless, our cryptosystem is firmly grounded in a very sensitive and stochastic chaotic system, in which even minute modifications to the initial conditions yield completely distinct chaotic sequences. Consequently, we use the pixel average values from the original image to disturb the initial values while setting them. This guarantees that our approach is robust to traditional attacks and has good plaintext sensitivity. Since the selected plaintext assault is thought to be the most aggressive, the other three forms of attacks usually pose little harm to an image encryption system that can withstand it. In this study, the high dependency on the plaintext means that any attempt by the attackers to reconstruct the original messages, after they have obtained the ciphertext through phoney messages, will be unsuccessful. Additionally, the NPCR and UACI values obtained in Section 4.8 further demonstrate that our algorithm is fully resistant to chosen-plaintext attacks. Also, we can see our system efficiency against the entropy-based cryptanalysis because the values of our experiment are near 8 (see section 4.3). Therefore, based on our experiment and cryptosystem analysis we can conclude that our scheme of cryptosystem is capable of resisting all four classical attacks.

## 6. CONCLUSIONS

In this research, we have introduced a novel method for image encryption that combines a 7-dimensional hyperchaotic system with advanced microfluidic technology for generating Dynamic S-Boxes. Our approach addresses the limitations of conventional encryption methods by offering enhanced complexity, unpredictability, and resilience against various cryptographic attacks. The dynamic nature of microfluidic technology in generating robust S-boxes, combined with the complexity of the 7D hyperchaotic system, ensures a high level of security by making the encryption process resistant to both differential and linear cryptanalysis. Additionally, the permutation entropy is introduced to further enhance the chaotic analysis, providing deeper insights into the system's unpredictability. The statistical analyses, including correlation coefficient analysis, information entropy, NPCR, UACI tests, and the Gray Level Co-occurrence Matrix (GLCM) analysis—assessing metrics like irregular deviation, contrast, energy, and uniformity—clearly demonstrate the superiority of our technique in terms of security and reliability. Furthermore, our encryption method has shown strong resistance to classical attacks, as evidenced by the comprehensive security evaluations conducted. When compared to existing encryption techniques, our method exhibits notable advantages in terms of scalability, efficiency, and security. The integration of microfluidic technology with chaotic systems in encryption represents a significant advancement in cryptographic research, offering a new paradigm for secure communication systems. This approach opens up several paths for future research. To have a cryptographic system with higher complexity by optimization of microfluidic technology is through integrating AI-driven fluid dynamics. Moreover, the

proposed Dynamic S-Box can be applied in other cryptographic systems, for example; quantum-resistant algorithms, and digital communication systems.

## ACKNOWLEDGEMENT

This work was accomplished at the University of Technology, and the University of Kirkuk, Kirkuk, Iraq.

## REFERENCES

- [1] Yang, D., Liao, X., Wang, Y., Yang, H., Wei, P. (2009). A novel chaotic block cryptosystem based on iterating map with output-feedback. *Chaos, Solitons & Fractals*, 41(1): 505-510. <https://doi.org/10.1016/j.chaos.2008.02.017>
- [2] Yang, H., Wong, K.W., Liao, X., Zhang, W., Wei, P. (2010). A fast image encryption and authentication scheme based on chaotic maps. *Communications in Nonlinear Science and Numerical Simulation*, 15(11): 3507-3517. <https://doi.org/10.1016/j.cnsns.2010.01.004>
- [3] Minas, N.A., MohammedSediq, F.H., Salih, A.I. (2018). Color image encryption using hybrid method of fractal-based key and private XOR key. *Kirkuk University Journal/Scientific Studies*, 13(1): 104-117.
- [4] Aaref, A.M. (2016). A developed discrete fourier transform based cryptosystem. *Kirkuk Journal of Science*, 11(1): 20-28.
- [5] Wazi, M.T., Ali, D.S., Al-Saidi, N.M., Alawn, N.A. (2022). A secure image cryptosystem via multiple chaotic maps. *Discrete Mathematics, Algorithms and Applications*, 14(4): 2150141. <https://doi.org/10.1142/S179383092150141X>
- [6] Ali, D. S., Alwan, N. A., & Al-Saidi, N. M. (2019, December). Image encryption based on highly sensitive chaotic system. In *AIP Conference Proceedings*, Istanbul, Turkey, p. 080007. <https://doi.org/10.1063/1.5136200>
- [7] Ndassi, H.L., Kengne, R., Tegue, A.G.G., Motchongom, M.T., Tchitnga, R., Tchoffo, M. (2023). A robust image encryption scheme based on compressed sensing and novel 7D oscillato with complex dynamics. *Heliyon*, 9(6): e16514.
- [8] Alwan, N.A., Obaiys, S.J., Noor, N.F.B.M., Al-Saidi, N.M., Karaca, Y. (2024). Color image encryption through multi-S-box generated by hyperchaotic system and mixture of pixel bits. *Fractals*, 2024: 2440039. <https://doi.org/10.1142/S0218348X24400395>
- [9] Hua, Z., Jin, F., Xu, B., Huang, H. (2018). 2D Logistic-Sine-coupling map for image encryption. *Signal Processing*, 149: 148-161. <https://doi.org/10.1016/j.sigpro.2018.03.010>
- [10] Al-Saidi, N.M., Younus, D., Natiq, H., Ariffin, M.R.K., Asbullah, M.A., Mahad, Z. (2020). A new hyperchaotic map for a secure communication scheme with an experimental realization. *Symmetry*, 12(11): 1881. <https://doi.org/10.3390/sym12111881>
- [11] Roy, A., Misra, A.P., Banerjee, S. (2019). Chaos-based image encryption using vertical-cavity surface-emitting lasers. *Optik*, 176: 119-131. <https://doi.org/10.1016/j.ijleo.2018.09.062>
- [12] Farhan, A.K., Ali, R.S., Natiq, H., Al-Saidi, N.M. (2019). A new S-box generation algorithm based on

- multistability behavior of a plasma perturbation model. *IEEE Access*, 7: 124914-124924. <https://doi.org/10.1109/ACCESS.2019.2938513>
- [13] Alwan, N.A., Yousif, A.Y., Al-Saidi, N.M. (2021). Performance-enhancing of RSA public key via three-dimensional hyperchaotic system. In *AIP Conference Proceedings*, Mersin, Turkey, p. 020027. <https://doi.org/10.1063/5.0040397>
- [14] Thivagar, M.L., Hamad, A.A., Tamilarasan, B., Antony, G.K. (2022). A novel seven-dimensional hyperchaotic. In *Proceedings of Second Doctoral Symposium on Computational Intelligence: DoSCI 2021*, Lucknow, India, pp. 329-340. [https://doi.org/10.1007/978-981-16-3346-1\\_27](https://doi.org/10.1007/978-981-16-3346-1_27)
- [15] Varan, M., Akgul, A. (2018). Control and synchronisation of a novel seven-dimensional hyperchaotic system with active control. *Pramana*, 90: 1-8. <https://doi.org/10.1007/s12043-018-1546-9>
- [16] Shannon, C.E. (1949). Communication theory of secrecy systems. *The Bell System Technical Journal*, 28(4): 656-715. <https://doi.org/10.1002/j.1538-7305.1949.tb00928.x>
- [17] Lorenz, E.N. (1996). Predictability: A problem partly solved. In *Proc. Seminar on Predictability*, Cambridge University Press, pp. 40-58. <https://doi.org/10.1017/CBO9780511617652.004>
- [18] Talatahari, S., Kaveh, A., Sheikholeslami, R. (2012). Chaotic imperialist competitive algorithm for optimum design of truss structures. *Structural and Multidisciplinary Optimization*, 46: 355-367. <https://doi.org/10.1007/s00158-011-0754-4>
- [19] Cao, C., Sun, K., Liu, W. (2018). A novel bit-level image encryption algorithm based on 2D-LICM hyperchaotic map. *Signal Processing*, 143: 122-133. <https://doi.org/10.1016/j.sigpro.2017.08.020>
- [20] Whitesides, G.M. (2006). The origins and the future of microfluidics. *Nature*, 442(7101): 368-373. <https://doi.org/10.1038/nature05058>
- [21] Alwan, N.A., Obaiys, S.J., Al-Saidi, N.M., Noor, N.F.B.M., Karaca, Y. (2024). A pseudo random number generator based on 4D hyperchaotic systems, riddled basins of attraction and advanced microfluidic technology. In *International Conference on Computational Science and Its Applications*, Hanoi, Vietnam, pp. 91-109. [https://doi.org/10.1007/978-3-031-65154-0\\_6](https://doi.org/10.1007/978-3-031-65154-0_6)
- [22] Fang, P., Liu, H., Wu, C., Liu, M. (2023). A survey of image encryption algorithms based on chaotic system. *The Visual Computer*, 39(5): 1975-2003. <https://doi.org/10.1007/s00371-022-02459-5>
- [23] Adams, C., Tavares, S. (1990). The structured design of cryptographically good S-boxes. *Journal of Cryptology*, 3(1): 27-41. <https://doi.org/10.1007/BF00203967>
- [24] Özkaynak, F., Özer, A.B. (2010). A method for designing strong S-Boxes based on chaotic Lorenz system. *Physics Letters A*, 374(36): 3733-3738. <https://doi.org/10.1016/j.physleta.2010.07.019>
- [25] Khan, M., Shah, T., Batool, S.I. (2016). Construction of S-box based on chaotic Boolean functions and its application in image encryption. *Neural Computing and Applications*, 27: 677-685. <https://doi.org/10.1007/s00521-015-1887-y>
- [26] Wang, X., Çavuşoğlu, Ü., Kacar, S., Akgul, A., Pham, V.T., Jafari, S., Nguyen, X.Q. (2019). S-box based image encryption application using a chaotic system without equilibrium. *Applied Sciences*, 9(4): 781. <https://doi.org/10.3390/app9040781>
- [27] Liu, L., Wang, J. (2023). A cluster of 1D quadratic chaotic map and its applications in image encryption. *Mathematics and Computers in Simulation*, 204: 89-114. <https://doi.org/10.1016/j.matcom.2022.07.030>
- [28] Gong, L., Qiu, K., Deng, C., Zhou, N. (2019). An image compression and encryption algorithm based on chaotic system and compressive sensing. *Optics & Laser Technology*, 115: 257-267. <https://doi.org/10.1016/j.optlastec.2019.01.039>
- [29] Ye, G., Pan, C., Huang, X., Mei, Q. (2018). An efficient pixel-level chaotic image encryption algorithm. *Nonlinear Dynamics*, 94: 745-756. <https://doi.org/10.1007/s11071-018-4391-y>
- [30] Ahmad, J., Khan, M.A., Ahmed, F., Khan, J.S. (2018). A novel image encryption scheme based on orthogonal matrix, skew tent map, and XOR operation. *Neural Computing and Applications*, 30: 3847-3857. <https://doi.org/10.1007/s00521-017-2970-3>
- [31] Song, C.Y., Qiao, Y.L., Zhang, X.Z. (2013). An image encryption scheme based on new spatiotemporal chaos. *Optik-International Journal for Light and Electron Optics*, 124(18): 3329-3334. <https://doi.org/10.1016/j.ijleo.2012.11.002>